

SIEMENS

SIMATIC

STEP 7
S7-1200 Motion Control V11 SP2

Function Manual

Preface

Using S7-1200
Motion Control

1

S7-1200 Motion Control

2

12/2011

A5E03790551-01

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.
⚠ WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.
⚠ CAUTION
with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.
CAUTION
without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.
NOTICE
indicates that an unintended result or situation can occur if the relevant information is not taken into account.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Preface

Preface

Purpose of this manual

This document provides you with detailed information on S7-1200 Motion Control. The contents of this document correspond to the contents of the STEP 7 V11 SP2 online help with respect to contents and structure. Interaction with STEP 7 is required to understand many parts of this document.

This document is aimed at programmers of STEP 7 programs and at people who work in the areas of configuring, commissioning and servicing automation systems with motion control applications.

Required basic knowledge

General knowledge in the fields of automation engineering and motion control is required to understand this document.

It is also essential to be familiar with the use of computers or programming devices under the Windows operating system.

Because S7-1200 Motion Control is based on STEP 7, you need knowledge of working with STEP 7.

Scope of this manual

This manual is valid for STEP 7 V11 SP2.

Further support

If you have any questions relating to the products described in this manual, and do not find the answers in this documentation, please contact your Siemens partner at our local offices.

You will find information on who to contact at:

(<http://www.siemens.com/automation/partner>)

A signpost to the documentation of the various SIMATIC products and systems is available at:

(<http://www.siemens.com/simatic-tech-doku-portal>)

You will find the online catalog and order system at:

(<http://mall.automation.siemens.com>)

Training center

We offer a range of courses to help get you started with the S7 programmable controller. Please contact your regional training center or the central Training center (<http://support.automation.siemens.com/WW/view/en/24486113>).

Table of contents

	Preface	3
1	Using S7-1200 Motion Control	9
1.1	Introduction	9
1.1.1	Motion functionality of the CPU S7-1200.....	9
1.1.2	Hardware components for motion control.....	10
1.2	Basics for working with S7-1200 Motion Control	13
1.2.1	CPU outputs relevant for motion control.....	13
1.2.2	How the pulse interface works.....	15
1.2.3	Relationship between the travel direction and voltage level at the direction output.....	16
1.2.4	Hardware and software limit switches	18
1.2.5	Jerk limit.....	19
1.2.6	Homing.....	20
1.3	Guidelines on use of motion control	21
1.4	Overview of versions.....	22
1.5	Techology object axis	24
1.5.1	Integration of the axis technology object	24
1.5.2	Tools of the axis technology object.....	27
1.5.3	Add technological object Axis	29
1.5.4	Configuring the axis technology object.....	30
1.5.4.1	Working with the configuration dialog	30
1.5.4.2	Basic parameters	31
1.5.4.3	Extended parameters.....	33
1.6	Technology object command table	53
1.6.1	Use of the command table technology object.....	53
1.6.2	Command table technology object tools.....	53
1.6.3	Adding the technological object command table	54
1.6.4	Configuring the command table technology object.....	55
1.6.4.1	Working with the configuration dialog	55
1.6.4.2	Basic parameters	56
1.6.4.3	Extended parameters.....	71
1.7	Download to CPU	74
1.8	Commissioning the axis - Axis control panel	76
1.9	Programming.....	79
1.9.1	Overview of the Motion Control statements.....	79
1.9.2	Creating a user program	80
1.9.3	Programming notes.....	83
1.9.4	Behavior of the Motion Control commands after POWER OFF and restart.....	85
1.9.5	Monitoring active commands	86
1.9.5.1	Monitoring active commands	86
1.9.5.2	Motion control instructions with "Done" output parameter	86
1.9.5.3	Motion control instruction MC_MoveVelocity	90

1.9.5.4	Motion control instruction MC_MoveJog	94
1.9.6	Error displays of the Motion Control statements	98
1.10	Axis - Diagnostics	100
1.10.1	Status and error bits	100
1.10.2	Motion status	103
1.10.3	Dynamics settings	103
1.11	Working with watch tables	104
1.12	Appendix	105
1.12.1	Using multiple axes with the same PTO	105
1.12.2	Using multiple drives with the same PTO	108
1.12.3	Tracking jobs from higher priority classes (execution levels)	109
1.12.4	Special cases for use of software limit switches	112
1.12.4.1	Software limit switches in conjunction with a homing operation	112
1.12.4.2	Software limit switches and software limit switch position changes.	116
1.12.4.3	Software limit switches in conjunction with dynamic changes	117
1.12.5	Reducing velocity for a short positioning duration	119
1.12.6	Dynamic adjustment of start/stop velocity	119
1.12.7	List of ErrorIDs and ErrorInfos (technology objects as of V2.0)	120
1.12.8	Tag of the Axis technology object	130
1.12.8.1	Config. tag	130
1.12.8.2	MotionStatus. tag	142
1.12.8.3	StatusBits. tag	143
1.12.8.4	ErrorBits. tag	146
1.12.8.5	Internal. tag	148
1.12.8.6	ControlPanel tag	148
1.12.8.7	Update of the technology object tags	148
1.12.9	Command table technology object tag	149
1.12.9.1	Config.Command.Command[1 ... 32] tag	149
1.12.10	Documentation for functions from previous versions	151
1.12.10.1	Configuration - Homing (technology object "Axis" V1.0)	151
1.12.10.2	List of ErrorIDs and ErrorInfos (technology objects V1.0)	154
2	S7-1200 Motion Control	161
2.1	MC_Power	161
2.1.1	MC_Power: Enable, disable axis	161
2.1.2	MC_Power: Function chart	165
2.2	MC_Reset	166
2.2.1	MC_Reset: Acknowledge error	166
2.3	MC_Home	168
2.3.1	MC_Home: Home axes, set home position	168
2.4	MC_Halt	172
2.4.1	MC_Halt: Halt axis	172
2.4.2	MC_Halt: Function chart	174
2.5	MC_MoveAbsolute	176
2.5.1	MC_MoveAbsolute: Absolute positioning of axes	176
2.5.2	MC_MoveAbsolute: Function chart	178
2.6	MC_MoveRelative	180
2.6.1	MC_MoveRelative: Relative positioning of axes	180

2.6.2	MC_MoveRelative: Function chart.....	182
2.7	MC_MoveVelocity	184
2.7.1	MC_MoveVelocity: Move axes at preset rotational speed	184
2.7.2	MC_MoveVelocity: Function chart	187
2.8	MC_MoveJog	189
2.8.1	MC_MoveJog: Move axes in jogging mode	189
2.8.2	MC_MoveJog: Function chart	191
2.9	MC_CommandTable	192
2.9.1	MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0).....	192
2.10	MC_ChangeDynamic	194
2.10.1	MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0).....	194
	Index.....	197

Using S7-1200 Motion Control

1.1 Introduction

1.1.1 Motion functionality of the CPU S7-1200

The TIA Portal, together with the "Motion Control" functionality of the CPU S7-1200, supports you in controlling stepper motors and servo motors with pulse interface:

- In the TIA Portal, you configure the "Axis" and "Command table" technology objects. The CPU S7-1200 controls the pulse and direction outputs for control of the drives using these technology objects.
- In the user program you control the axis by means of motion control instructions and initiate motion jobs of your drive.

See also

Hardware components for motion control (Page 10)

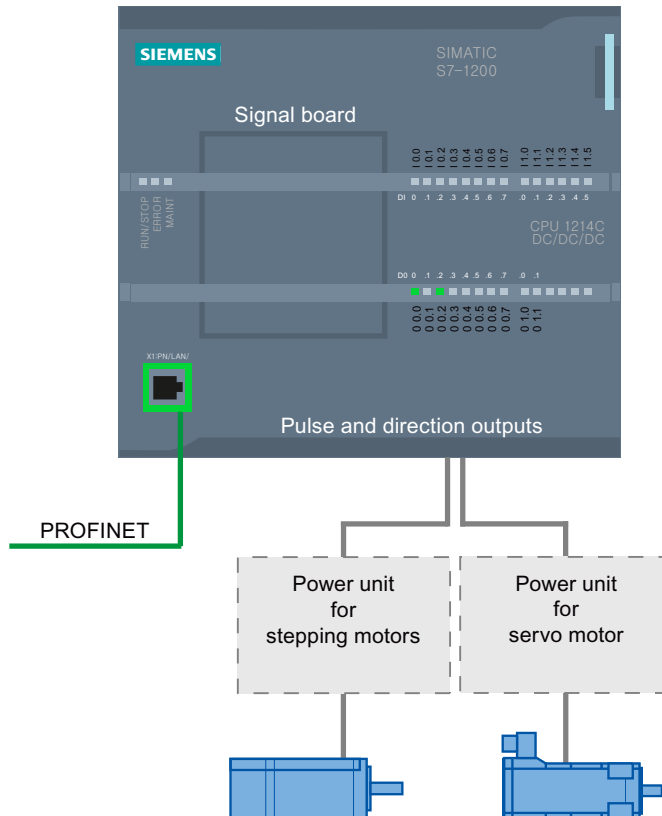
Integration of the axis technology object (Page 24)

Use of the command table technology object (Page 53)

Command table technology object tools (Page 53)

1.1.2 Hardware components for motion control

The representation below shows the basic hardware configuration for a motion control application with the CPU S7-1200.



CPU S7-1200:

CPU S7-1200 combines the functionality of a programmable logic controller with motion control functionality for operation of stepper motors and servo motors with pulse interface. The motion control functionality takes over the control and monitoring of the drives.

The DC/DC/DC variants of the CPU S7-1200 have onboard outputs for direct control of drives. The relay variants of the CPU require one of the signal boards described below to control a drive.

Signal board

You add further inputs and outputs to the CPU with the signal boards. The digital outputs can be used as pulse and direction outputs for controlling drives as required.

In CPUs with relay outputs, the pulse signal cannot be output on the on-board outputs because the relays do not support the necessary switching frequencies. A signal board with digital outputs must be used to enable you to work with the PTO (Pulse Train Output) on these CPUs.

When a DC/DC/DC variant of the CPU S7-1200 is used together with a signal board, the maximum number of controllable drives is limited to "2".

PROFINET

Use the PROFINET interface to establish the online connection between the CPU S7-1200 and the programming device. In addition to the online functions of the CPU, additional commissioning and diagnostic functions are available for motion control.

Maximum number of controllable drives

The maximum number of controllable drives for the various CPU versions is provided in the following table:

CPU		Signal board					
		without	DI2/DO2 x DC24V 20kHz	DI2/DO2 x DC24V 200kHz	DO4 x DC24V 200kHz	DI2/DO2 x DC5V 200kHz	DO4 x DC5V 200kHz
CPU 1211C, CPU 1212C, CPU 1214C	DC/DC/DC	2	2	2	2	2	2
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2

Limit frequencies of pulse outputs

The following limit frequencies apply to the pulse outputs:

Pulse output	Limit frequencies for technology object "Axis" V1.0	Limit frequencies for technology object "Axis" V2.0 and higher
Onboard	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$
Signal board DI2/DO2 x DC24V 20kHz	$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 20 \text{ kHz}$
Signal board DI2/DO2 x DC24V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$
Signal board DO4 x DC24V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$
Signal board DI2/DO2 x DC5V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$
Signal board DO4 x DC5V 200kHz	$2 \text{ Hz} \leq f \leq 100 \text{ kHz}$	$2 \text{ Hz} \leq f \leq 200 \text{ kHz}$

Ordering information

The order information listed below applies to the currently installed product phase (without any installed Hardware Support Packages) of the TIA Portal.

Name	MLFB (order no.)
CPU 1211C DC/DC/DC	6ES7211-1AD30-0XB0
CPU 1211C AC/DC/RLY	6ES7211-1BD30-0XB0
CPU 1211C DC/DC/RLY	6ES7211-1HD30-0XB0
CPU 1212C DC/DC/DC	6ES7212-1AD30-0XB0
CPU 1212C AC/DC/RLY	6ES7212-1BD30-0XB0
CPU 1212C DC/DC/RLY	6ES7212-1HD30-0XB0
CPU 1214C DC/DC/DC	6ES7214-1AE30-0XB0
CPU 1214C AC/DC/RLY	6ES7214-1BE30-0XB0
CPU 1214C DC/DC/RLY	6ES7214-1HE30-0XB0
Signal board DI2/DO2 x DC24V 20kHz	6ES7223-0BD30-0XB0
Signal board DI2/DO2 x DC24V 200kHz	6ES7223-3BD30-0XB0
Signal board DO4 x DC24V 200kHz	6ES7222-1BD30-0XB0
Signal board DI2/DO2 x DC5V 200kHz	6ES7223-3AD30-0XB0
Signal board DO4 x DC5V 200kHz	6ES7222-1AD30-0XB0

Use a Hardware Support Package (HSP) to install new hardware components. The hardware component will then be available in the hardware catalog.

See also

Motion functionality of the CPU S7-1200 (Page 9)

CPU outputs relevant for motion control (Page 13)

1.2 Basics for working with S7-1200 Motion Control

1.2.1 CPU outputs relevant for motion control

Pulse and direction output

The CPU provides one pulse output and one direction output for controlling a stepper motor drive or a servo motor drive with pulse interface. The pulse output provides the drive with the pulses required for motor motion. The direction output controls the travel direction of the drive.

Pulse and direction outputs are permanently assigned to one another. Onboard CPU outputs or outputs of a signal board can be used as pulse and direction outputs. You select between onboard CPU outputs and outputs of the signal board during device configuration under Pulse generators (PTO/PWM) on the "Properties" tab.

The following table shows the address assignment of the pulse and direction outputs:

CPU S7-1200:	Without signal board				Signal boards DI2/DO2 *)				Signal boards DO4 **)			
	Outputs PTO1		Outputs PTO2		Outputs PTO1		Outputs PTO2		Outputs PTO1		Outputs PTO2	
	Pls.	Dir.	Pls.	Dir.	Pls.	Dir.	Pls.	Dir.	Pls.	Dir.	Pls.	Dir.
CPU 1211C, CPU 1212C, CPU 1214C (DC/DC/DC)	Ax.0	Ax.1	Ax.2	Ax.3	Ax.0	Ax.1	Ax.2	Ax.3	Ax.0	Ax.1	Ax.2	Ax.3
					Ay.0	Ay.1			Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211C, CPU 1212C, CPU 1214C (AC/DC/RLY)	-	-	-	-	Ay.0	Ay.1	-	-	Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211C, CPU 1212C, CPU 1214C (DC/DC/RLY)	-	-	-	-	Ay.0	Ay.1	-	-	Ay.0	Ay.1	Ay.2	Ay.3

x = Initial byte address of onboard CPU outputs (default value = 0)

y = Initial byte address of signal board outputs (default value = 4)

* If a DC/DC/DC CPU variant is used together with a DI2/DO2 signal board, the signals of the PTO1 can be generated via the onboard CPU outputs or via the signal board.

** If a DC/DC/DC CPU variant is used together with a DO4 signal board, the signals for PTO1 and PTO2 can be generated via the onboard CPU outputs or via the signal board.

Drive interface

For motion control, you can optionally parameterize a drive interface for "Drive enabled" and "Drive ready". When using the drive interface the digital output for the drive enable and the digital input for "drive ready" can be freely selected.

Note

The firmware will take control via the corresponding pulse and direction outputs if the PTO (Pulse Train Output) has been selected and assigned to an axis.

With this takeover of the control function, the connection between the process image and I/O output is also disconnected. While the user has the possibility of writing the process image of pulse and direction outputs via the user program or watch table, this is not transferred to the I/O output. Accordingly, it is also not possible to monitor the I/O output via the user program or watch table. The information read reflects the value of the process image and does not match the real status of the I/O output.

For all other CPU outputs that are not used permanently by the CPU firmware, the status of the I/O output can be controlled or monitored via the process image, as usual.

See also

How the pulse interface works (Page 15)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Hardware and software limit switches (Page 18)

Jerk limit (Page 19)

Homing (Page 20)

Hardware components for motion control (Page 10)

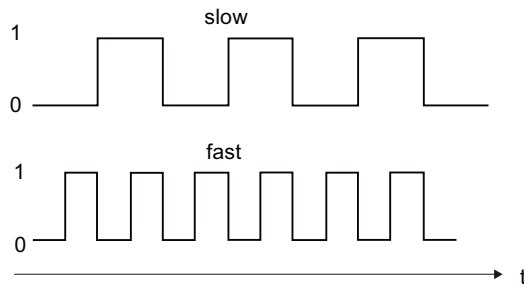
Integration of the axis technology object (Page 24)

Tools of the axis technology object (Page 27)

1.2.2 How the pulse interface works

Depending on the settings of the stepper motor, each pulse affects the movement of the stepper motor by a specific angle. If the stepper motor is set to 1000 pulses per revolution, for example, it moves 0.36° per pulse.

The speed of the stepper motor is determined by the number of pulses per time unit.



(The statements made here also apply to servo motors with pulse interface.)

See also

CPU outputs relevant for motion control (Page 13)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Hardware and software limit switches (Page 18)

Jerk limit (Page 19)

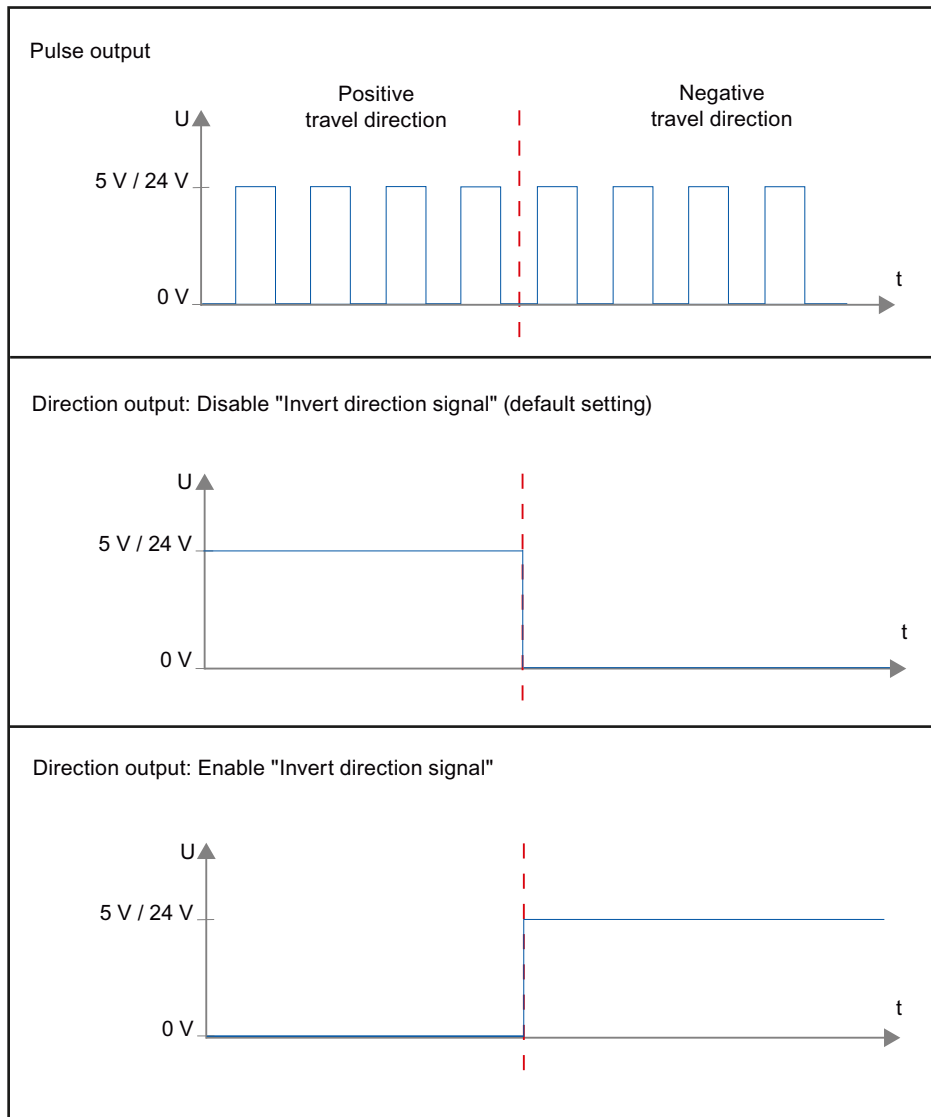
Homing (Page 20)

Integration of the axis technology object (Page 24)

Tools of the axis technology object (Page 27)

1.2.3 Relationship between the travel direction and voltage level at the direction output

The direction output of the CPU specifies the travel direction of the drive. You configure the direction signal under "Mechanics" in the axis configuration. The relationships between configuration, direction output, and travel direction are presented in the following diagram:



If "Invert direction signal" is deactivated in the configuration, a level of 5 V / 24 V will be output at the direction output for a positive travel direction (the voltage depends on the hardware used). A 0 V level will be output at the direction output for positive travel direction if "Invert direction signal" is activated in the configuration.

See also

CPU outputs relevant for motion control (Page 13)

How the pulse interface works (Page 15)

Hardware and software limit switches (Page 18)

Jerk limit (Page 19)

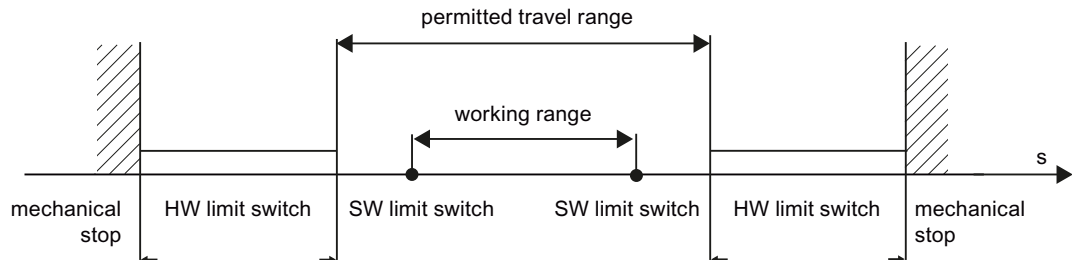
Homing (Page 20)

Integration of the axis technology object (Page 24)

Tools of the axis technology object (Page 27)

1.2.4 Hardware and software limit switches

Use the hardware and software limit switches to limit the "permitted traversing range" and the "working range" of your axis technology object. The relationships are shown in the following diagram:



Hardware limit switches are limit switches that limit the maximum "permitted traversing range" of the axis. Hardware limit switches are physical switching elements that must be connected to interrupt-capable inputs of the CPU.

Software limit switches limit the "working range" of the axis. They should fall inside the hardware limit switches relative to the traversing range. Since the positions of the software limit switches can be flexibly set, the working range of the axis can be adapted on an individual basis, depending on the current traversing profile. In contrast to hardware limit switches, software limit switches are implemented exclusively via the software and do not require their own switching elements.

Hardware and software limit switches must be activated prior to use in the configuration or in the user program.. Software limit switches are only active after homing the axis.

See also

CPU outputs relevant for motion control (Page 13)

How the pulse interface works (Page 15)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Jerk limit (Page 19)

Homing (Page 20)

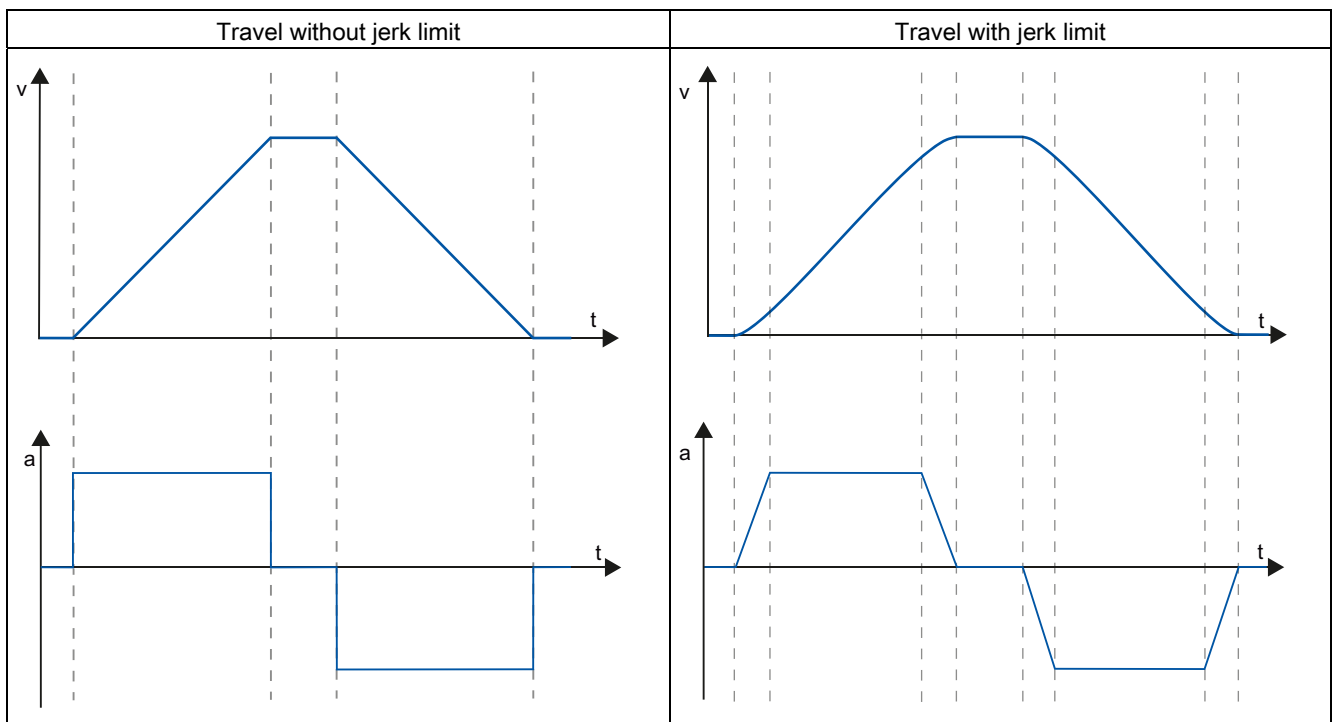
Integration of the axis technology object (Page 24)

Tools of the axis technology object (Page 27)

Position limits (Page 35)

1.2.5 Jerk limit

With the jerk limit you can reduce the stresses on your mechanics during an acceleration and deceleration ramp. The acceleration and deceleration value is not changed abruptly when the jerk limiter is active; it is gradually increased and decreased. The figure below shows the velocity and acceleration curve without and with jerk limit.



The jerk limit gives a "smoothed" velocity profile of the axis motion. This ensures soft starting and braking of a conveyor belt for example.

See also

Behavior of the axis when using the jerk limit (Page 44)

CPU outputs relevant for motion control (Page 13)

How the pulse interface works (Page 15)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Hardware and software limit switches (Page 18)

Homing (Page 20)

Integration of the axis technology object (Page 24)

Tools of the axis technology object (Page 27)

1.2.6 Homing

Homing means matching the axis coordinates of the technology object to the real, physical location of the drive. For position-controlled axes the entries and displays for the position refer exactly to these axis coordinates. Therefore, agreement between the axis coordinates and the real situation is extremely important. This step is necessary to ensure that the absolute target position of the axis is also achieved exactly with the drive.

In the S7-1200 CPU, axis homing is implemented with the motion control instruction, "MC_Home". The following homing modes exist:

Homing modes

- **Active homing**

In active homing mode, the motion control instruction "MC_Home" performs the required reference point approach. When the homing switch is detected, the axis is homed according to the configuration. Active traversing motions are aborted.

- **Passive homing**

During passive homing, the motion control instruction "MC_Home" does not carry out any homing motion. The traversing motion required for this step must be implemented by the user via other motion control instructions. When the homing switch is detected, the axis is homed according to the configuration. Active traversing motions are not aborted upon start of passive homing.

- **Direct homing absolute**

The axis position is set regardless of the homing switch. Active traversing motions are not aborted. The value of input parameter "Position" of motion control instruction "MC_Home" is set immediately as the reference point of the axis.

- **Direct homing relative**

The axis position is set regardless of the homing switch. Active traversing motions are not aborted. The following statement applies to the axis position after homing:

New axis position = current axis position + value of parameter "Position" of instruction "MC_Home".

See also

CPU outputs relevant for motion control (Page 13)

How the pulse interface works (Page 15)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Hardware and software limit switches (Page 18)

Jerk limit (Page 19)

Integration of the axis technology object (Page 24)

Tools of the axis technology object (Page 27)

Homing (technology object "Axis" as of V2.0) (Page 47)

1.3 Guidelines on use of motion control

The guidelines described here present the basic procedure for using motion control with the CPU S7-1200.

Requirements

To use the "Axis" technology object, you must create a project with a CPU S7-1200.

Procedure

Follow the steps below in the order given to use motion control with the CPU S7-1200. Use the following links for this purpose:

1. Add technological object Axis (Page 29)
2. Working with the configuration dialog (Page 30)
3. Download to CPU (Page 74)
4. Function test of the axis in the commissioning window (Page 76)
5. Programming (Page 79)
6. Diagnostics of the axis control (Page 100)

1.4 Overview of versions

The relationship between the relevant versions for S7-1200 Motion Control can be found in the following table:

Technology version

You can check the technology version currently selected in Task Card Instructions > Technology > Motion Control > S7-1200 Motion Control and in the "Add new object" dialog. Select the technology version in Task Card Instructions > Technology > Motion Control > S7-1200 Motion Control. If a technology object with an alternative version is added in the "Add new object" dialog, the technology version will also be changed.

Note

The selection of an alternative technology version will also affect the Motion Control Instructions version (task card). The technology objects and Motion Control instructions will only be converted to the selected version upon compilation or "Load to device".

Version of the technology object

The version of a technology object can be checked in the inspector window under "Properties > General > Information" in the "Version" field.

To change the version, select the relevant version in the task card under Instructions > Technology and then the menu command Edit > Compile. If a technology object with an alternative version is added in the "Add new object" dialog, the technology object version will also be changed.

Deal with the causes of any errors if error information is displayed during compilation. Repeat compilation until it can be completed without errors.

Check the configuration of the technology objects once you have done so.

Motion Control instruction version

Proceed as follows to check the version of a Motion Control instruction:

1. Open the Program blocks > System blocks > Program resources folders in the navigator and select the required Motion Control instruction.
2. Select the Edit > Properties menu command.
3. You will find the Motion Control instruction version in the Version field of the Information tab.

If the Motion Control instruction version used is not in line with the following compatibility list, the relevant Motion Control instructions will be highlighted in the program editor.

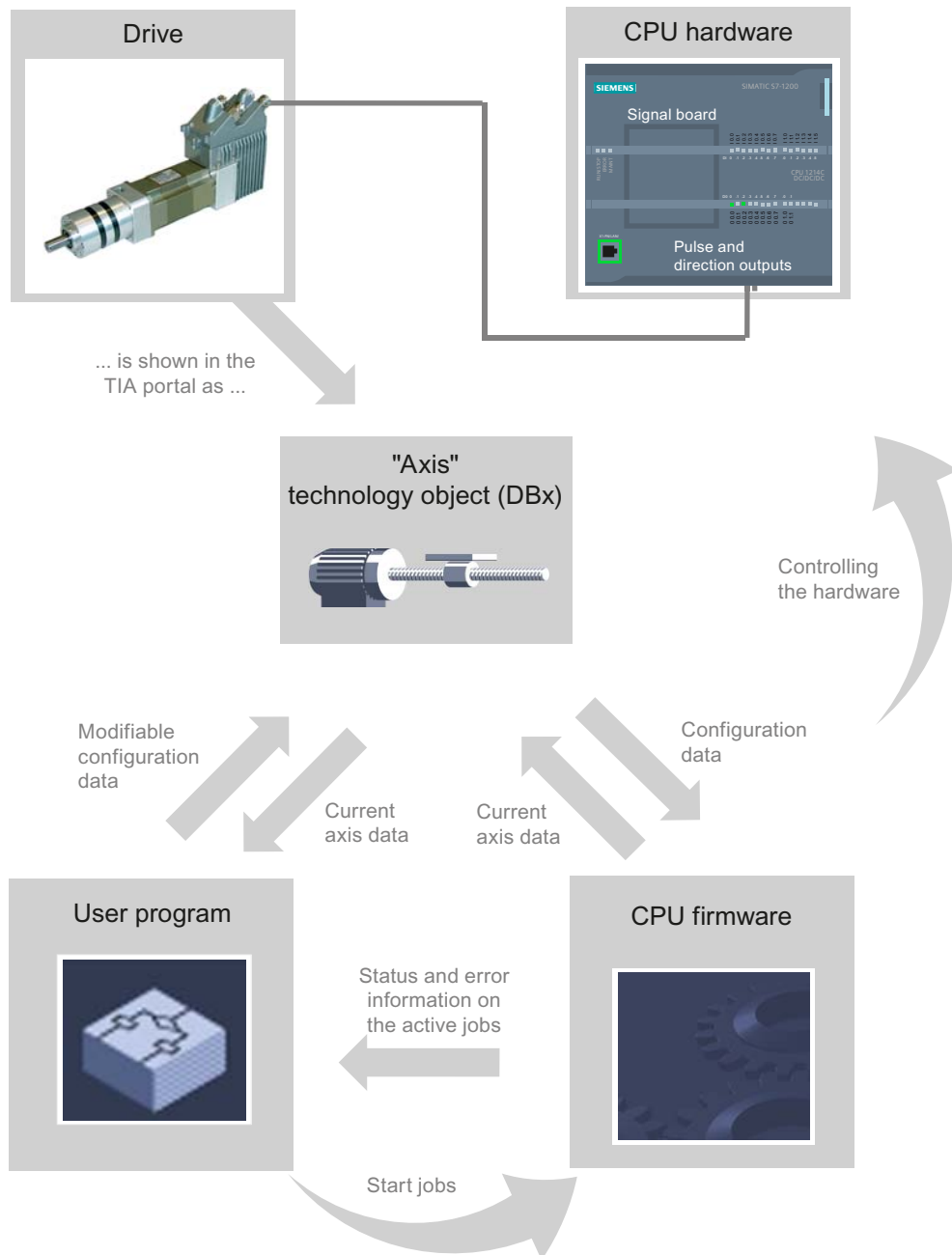
Compatibility list

Technology		CPU	Technology object	Motion Control Instruction
V1.0		V1.0, V2.0, V2.1, V2.2	Axis V1.0	MC_Power V1.0 MC_Reset V1.0 MC_Home V1.0 MC_Halt V1.0 MC_MoveAbsolute V1.0 MC_MoveRelative V1.0 MC_MoveVelocity V1.0 MC_MoveJog V1.0
V2.0	Innovations: <ul style="list-style-type: none"> • Jerk control • Command table • MC_ChangeDynamic 	V2.1, V2.2	Axis V2.0, Command table V2.0	MC_Power V2.0 MC_Reset V2.0 MC_Home V2.0 MC_Halt V2.0 MC_MoveAbsolute V2.0 MC_MoveRelative V2.0 MC_MoveVelocity V2.0 MC_MoveJog V2.0 MC_CommandTable V2.0 MC_ChangeDynamic V2.0
V3.0	Innovation: Load in RUN operating mode	V2.2	Axis V3.0, Command table V3.0	MC_Power V3.0 MC_Reset V3.0 MC_Home V3.0 MC_Stop V3.0 MC_MoveAbsolute V3.0 MC_MoveRelative V3.0 MC_MoveVelocity V3.0 MC_MoveJog V3.0 MC_CommandTable V3.0 MC_ChangeDynamic V3.0

1.5 Technology object axis

1.5.1 Integration of the axis technology object

The following representation shows the relations between the hardware and software components which are implemented when using the "Axis" technology object:



CPU hardware

The physical drive is controlled and monitored by the CPU hardware.

Drive

The drive represents the unit of power unit and motor. Stepper motors or servo motors with a pulse interface may be used.

Technology object "Axis"

The physical drive including mechanics is mapped in the TIA Portal as an "axis" technology object. Configure the "Axis" technology object with the following parameters for this:

- Selection of the PTOs (Pulse Train Output) to be used and configuration of the drive interface
- Parameter for mechanics and gear transmission of the drive (or the machine or system)
- Parameter for position monitoring, for dynamic parameters and for homing

The configuration of the "Axis" technology object is saved in the technology object (data block). This data block also forms the interface between the user program and the CPU firmware. The current axis data is saved in the data block of the technology object at the runtime of the user program.

User program

You start motion control instructions jobs in the CPU firmware with the user program. The following jobs for controlling the axis are possible:

- Position axis absolutely
- Position axis relatively
- Move axis with velocity set point
- Run axis jobs as movement sequence (as of technology V2.0)
- Move axis in jog mode
- Stop axis
- Reference axis; set reference point
- Acknowledge error

You determine the command parameters with the input parameters of the Motion Control instructions and the axis configuration. The output parameters of the instruction give you up to date information about the status and any errors of the command.

Before starting a command for the axis, you must enable the axis with the motion control instruction "MC_Power".

You can read out configuration data and current axis data with the tags of the technology object. You can change single, changeable tags of the technology object (e.g. the current acceleration) from the user program.

CPU firmware

The motion control jobs started in the user program are processed in the CPU firmware. When using the axis control table, Motion Control jobs are triggered by operating the axis control table. The CPU firmware performs the following jobs depending on the configuration:

- Calculate the exact motion profile for motion jobs and emergency stop situations
- Control the drive enable and the pulse and direction signal
- Monitor the drive and the hardware and software limit switches
- Up to date feedback of status and error information to the motion control instructions in the user program
- Writing of current axis data into the data block of the technology object

See also

CPU outputs relevant for motion control (Page 13)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Tools of the axis technology object (Page 27)

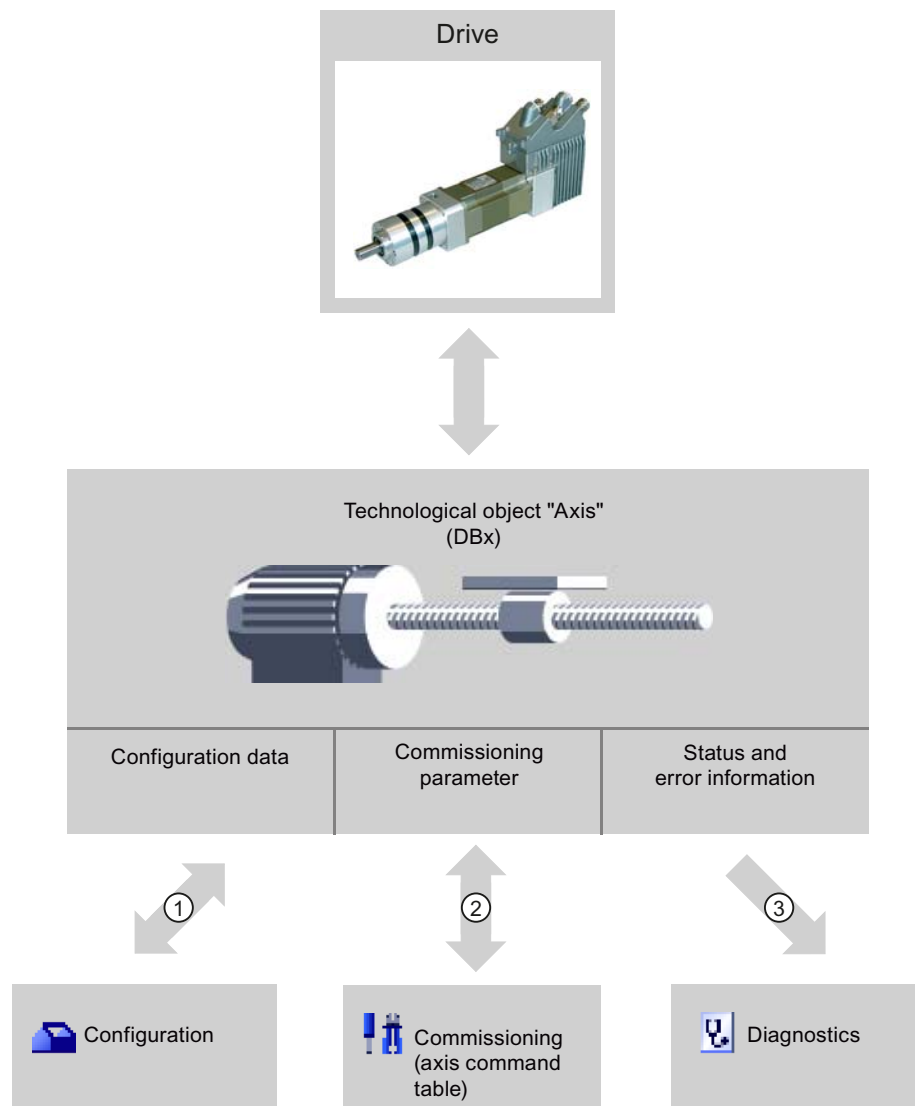
Hardware and software limit switches (Page 18)

Homing (Page 20)

Tag of the Axis technology object (Page 130)

1.5.2 Tools of the axis technology object

The TIA Portal provides the "Configuration", "Commissioning", and "Diagnostics" tools for the "Axis" technology object. The following representation shows the interaction of the three tools with the technology object and the drive:



①	Reading and writing of configuration data of the technology object;
②	Drive control via the technology object; Reading axis status for display on the control panel
③	Readout of the current status and error information of the technology object.

Configuration

Use the "Configuration" tool to configure the following properties of the "Axis" technology object:

- Selection of the PTO to be used and configuration of the drive interface
- Properties of the mechanics and the transmission ratio of the drive (or machine or system)
- Properties for position monitoring, dynamics, and homing

Save the configuration in the data block of the technology object.

Commissioning

Use the "Commissioning" tool to test the function of your axis without having to create a user program. When the tool is started, the axis control table will be displayed. The following commands are available on the axis control table:

- Releasing and blocking the axis
- Move axis in jog mode
- Position axis in absolute and relative terms
- Home axis
- Acknowledge errors

The dynamic values can be adjusted accordingly for the motion commands. The axis control table also shows the current axis status.

Diagnostics

Use the "Diagnostics" tool to keep track of the current status and error information for the axis and drive.

See also

CPU outputs relevant for motion control (Page 13)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Integration of the axis technology object (Page 24)

Hardware and software limit switches (Page 18)

Homing (Page 20)

Configuring the axis technology object (Page 30)

Commissioning the axis - Axis control panel (Page 76)

Axis - Diagnostics (Page 100)

1.5.3 Add technological object Axis

Proceed as follows to add an "Axis" technology object in the project tree:

Requirements

A project with a CPU S7-1200 has been created.

Procedure

1. Open the CPU folder in the project tree.
2. Open the technology objects folder.
3. Double-click "Add new object".
The "Add new object" dialog opens.
4. Select the "Motion" technology.
5. Open the "Motion Control" folder.
6. Open the "S7-1200 Motion Control" folder.
7. Click on the version and select an alternative version of the technology if you want to add an axis from an older version.
8. Select the "TO_Axis_PTO" object.
9. Change the name of the axis in the "Name" input field to suit your needs.
10. Select the "Manual" option if you want to change the suggested data block number.
11. Click "More information" if you want to supplement user information for the technology object.
12. Click "OK" to add the technology object.
Click "Cancel" to discard your entries.

Result

The new technology object is created and saved to the "Technology objects" folder in the project tree.

See also

Guidelines on use of motion control (Page 21)

1.5.4 Configuring the axis technology object

1.5.4.1 Working with the configuration dialog

You configure the properties of the technology object in the configuration window. Proceed as follows to open the configuration window of the technology object:

1. Open the group of the required technology object in the project tree.
2. Double-click the "Configuration" object.

The configuration is divided into the following categories:

- **Basic parameters**





The basic parameters contain all the parameters which must be configured for a functioning axis.

- **Extended parameters**

The advanced parameters include parameters to adapt to your drive or your plant.

Icons of the configuration window

Icons in the area navigation of the configuration show additional details about the status of the configuration:

	<p>The configuration contains default values and is complete.</p> <p>The configuration contains only default values. With these default values you can use the technology object without additional changes.</p>
	<p>The configuration contains values set by the user and is complete.</p> <p>All input fields of the configuration contain valid values and at least one preset value has changed.</p>
	<p>The configuration is incomplete or incorrect</p> <p>At least one input field or drop-down list contains an invalid value. The corresponding field or the drop-down list is displayed on a red background. Click the roll-out error message to indicate the cause of error.</p>
	<p>The configuration is valid but contains warnings</p> <p>Only one hardware limit switch is configured. Depending on the plant, the lacking configuration of a hardware limit switch may result in a hazard. The corresponding field or the drop-down list is displayed on a yellow background.</p>

See also

Guidelines on use of motion control (Page 21)

Basic parameters (Page 31)

Extended parameters (Page 33)

1.5.4.2 Basic parameters

Configuration - General

Configure the basic properties of the "Axis" technology object in the "General" configuration window.

Axis name:

Define the name of the axis or the name of the "Axis" technology object in this box. The technology object is listed under this name in the project navigation.

Hardware interface

The pulses are output to the power unit of the drive by fixed assigned digital outputs.

In CPUs with relay outputs, the pulse signal cannot be output on these outputs because the relays do not support the necessary switching frequencies. A signal board with digital outputs must be used to enable you to work with the PTO (Pulse Train Output) on these CPUs.

Note

The PTO requires the functionality of a fast counter (HSC) internally. The corresponding fast counter can therefore not be used elsewhere. The count can not be evaluated from its input address.

The assignment between PTO and HSC is fixed. When the user activates PTO1, it is connected to the HSC1. If the PTO2 is activated, this is connected with the HSC2.

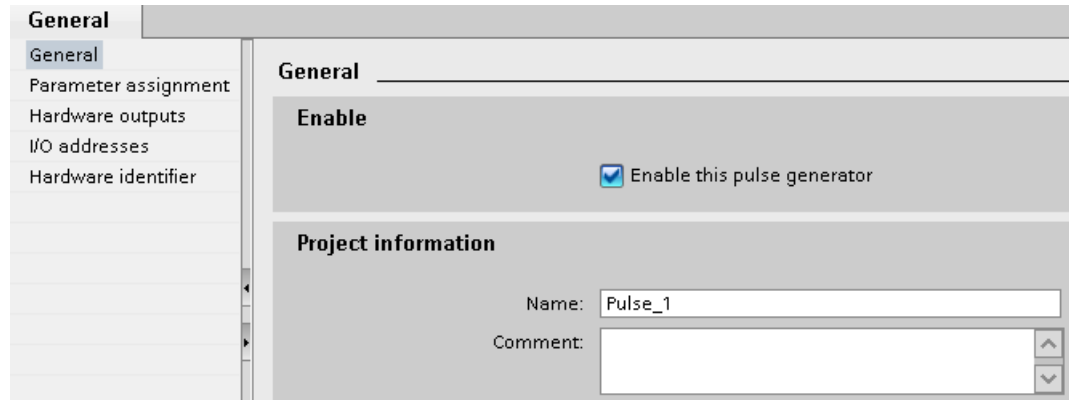
In the drop-down list "Pulse generator selection", select the PTO (Pulse Train Output) which are to provide the pulses for controlling the stepper motors or servo motors with a pulse interface. If the pulse generators and high-speed counters are not used elsewhere in the device configuration, the hardware interface can be configured automatically. In this case, the PTO selected in the drop-down list is displayed with a white background. The interfaces used will be listed in the "Output source", "Pulse output", "Direction output" and "Assigned fast counter" output fields.

Proceed as follows if you wish to change the interfaces or if the PTO could not be automatically configured (entry in the "Pulse generator selection" drop-down list is highlighted in red):

1. Click on the "Device configuration" button.

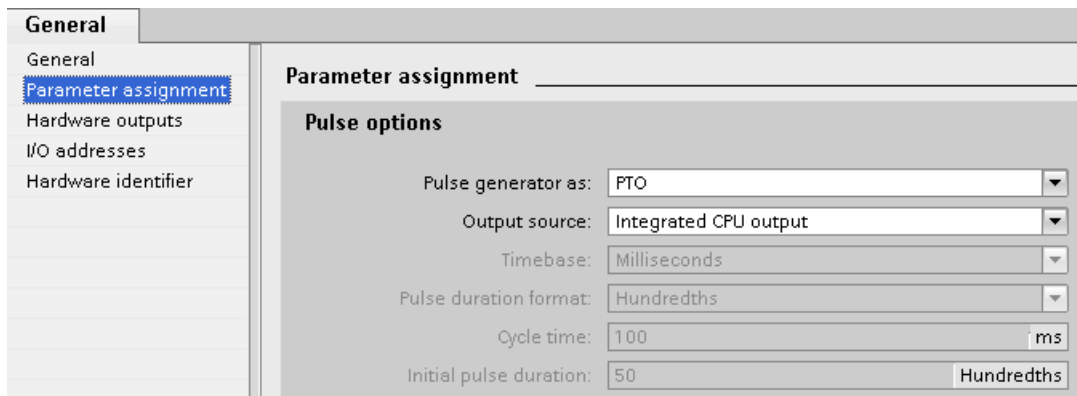
The pulse generator device configuration opens.

Enlarge the property window of the device configuration if the configuration of the pulse generator is not visible.



2. Select the "Enable this pulse generator" check box.
3. Select the "Parameter assignment" entry in the block navigator.

The "Parameter assignment" opens.



4. In the "Pulse generator as:" dropdown list select the "PTO" entry.
5. In the "Output source:" dropdown list select the "Integrated CPU output" or "Signal board output" entry. The "Signal board output" entry can only be selected for PTO1 or for PTO2 and PTO2 depending on the plugged signal board. For more detailed information, see chapter: CPU outputs relevant for motion control (Page 13)
6. Go back to the axis configuration.

Unless the corresponding fast counter has already been used elsewhere, the PTO boxes of the "General" axis configuration are not shaded red. Correct the configuration based on the error messages if this is not the case.

User unit

Select the desired unit for the dimension system of the axis in the dropdown list. The selected unit is used for the further configuration of the "Axis" technology object and for the display of the current axis data.

The values at the input parameters (Position, Distance, Velocity, ...) of the Motion Control instructions also refer to this unit.

NOTICE

Later changing of the dimension system may not be converted correctly in all the configuration windows of the technology object. In this case check the configuration of all axis parameters.

The values of the input parameters of the Motion Control instructions may have to be adapted to the new unit of measurement in the user program.

1.5.4.3 Extended parameters

Configuration - Drive interface

Configure the output for drive enable and the input for the "Drive ready" feedback signal of the drive in the "Drive signals" configuration window.

Drive enable is controlled by Motion Control instruction "MC_Power" and enables power to the drive. The signal is provided to the drive via the output to be configured.

The drive signals "Drive ready" to the CPU if it is ready to start executing travel after receipt of drive enable. The "Drive ready" signal is reported back to the CPU via the input to be configured.

If the drive does not have any interfaces of this type, you will not have to configure the parameters. In this case, select the value TRUE for the ready input.

See also

Configuration - Mechanics (Page 34)

Position limits (Page 35)

Dynamics (Page 40)

Homing (technology object "Axis" as of V2.0) (Page 47)

Configuration - Mechanics

Configure the mechanical properties of the drive in the "Mechanics" configuration window.

Increments per motor revolution

Configure the number of pulses required for one revolution of the motor in this field.

Limits (independent of the selected unit of measurement):

- $0 < \text{Pulse per motor revolution} \leq 2147483647$

Load distance per motor revolution

In this field, configure the load distance per motor revolution covered by the mechanical system of your unit.

Limits (independent of the selected unit of measurement):

- $0.0 < \text{Load distance per motor revolution} \leq 1.0e12$

Invert direction signal

You can adjust the direction output to the direction logic of the drive using the "Invert direction signal" check box.

- **Invert direction signal: deactivated**
 - 0 V level = negative travel direction
 - 5 V / 24 V level = positive travel direction
(the actual voltage depends on the hardware used)
- **Invert direction signal: activated**
 - 0 V level = positive travel direction
 - 5 V / 24 V level = negative travel direction
(the actual voltage depends on the hardware used)

See also

Configuration - Drive interface (Page 33)

Position limits (Page 35)

Dynamics (Page 40)

Homing (technology object "Axis" as of V2.0) (Page 47)

Relationship between the travel direction and voltage level at the direction output (Page 16)

Position limits

Requirements for hardware limit switches

Use only hardware limit switches that remain permanently switched after being approached. This switching status may only be revoked after a return to the valid travel range.

See also

Configuration - Position limits (Page 35)

Behavior of axis when position limits is tripped (Page 37)

Changing the position limits configuration in the user program (Page 39)

Configuration - Position limits

Configure the hardware and software limit switches of the axis in the "Position limits" configuration window.

Enable hardware limit switch

Activate the function of the low and high hardware limit switch with this check box. The hardware limit switches can be used for purposes of direction reversal during a reference point approach. For details, refer to the configuration description for homing.

Low / high HW limit switch input

Select the digital input for the low or high hardware limit switch from the drop-down list. The input must be interrupt-capable. The digital onboard CPU inputs and the digital inputs of a plugged signal board can be selected as inputs for the HW limit switches.

CAUTION

The digital inputs are set to a filter time of 6.4 ms by default. If these are used as hardware limit switches, undesired decelerations may occur. If this occurs, reduce the filter time for the relevant digital inputs.

The filter time can be set under "Input filter" in the device configuration of the digital inputs.

Active level

In the drop-down list, select the signal level available at the CPU when the hardware limit switch is approached.

- "Low level" selected
0 V (FALSE) at CPU input corresponds to hardware limit switch approached
- "High level" selected
5 V / 24 V (TRUE) at the CPU input = hardware limit switch approached
(the actual voltage depends on the hardware used)

Enable software limit switch

Activate the function of the low and high software limit switch with this check box.

NOTICE
The enabled software limit switch only affects a homed axis.

High and low software limit switch

Enter the position value of the low and high software limit switch in these boxes.

Limits (independent of the selected unit of measurement):

- $-1.0e12 \leq \text{low software limit switch} \leq 1.0e12$
- $-1.0e12 \leq \text{high software limit switch} \leq 1.0e12$

The value of the high software limit switch must be greater than or equal to the value of the low software limit switch.

See also

Requirements for hardware limit switches (Page 35)

Behavior of axis when position limits is tripped (Page 37)

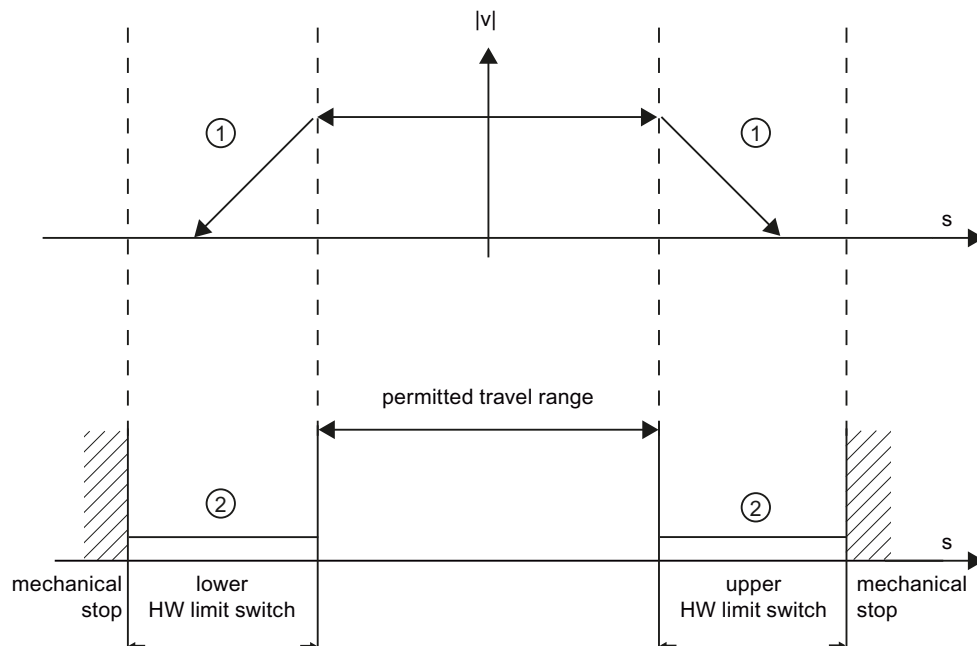
Changing the position limits configuration in the user program (Page 39)

Configuration - Homing - Active (Page 48)

Behavior of axis when position limits is tripped

Behavior of axis when hardware limit switches are approached

When the hardware limit switches are approached, the axis brakes to a standstill at the configured emergency stop deceleration. The specified emergency stop deceleration must be sufficient to reliably stop the axis before the mechanical stop. The following diagram presents the behavior of the axis after it approaches the hardware limit switches:



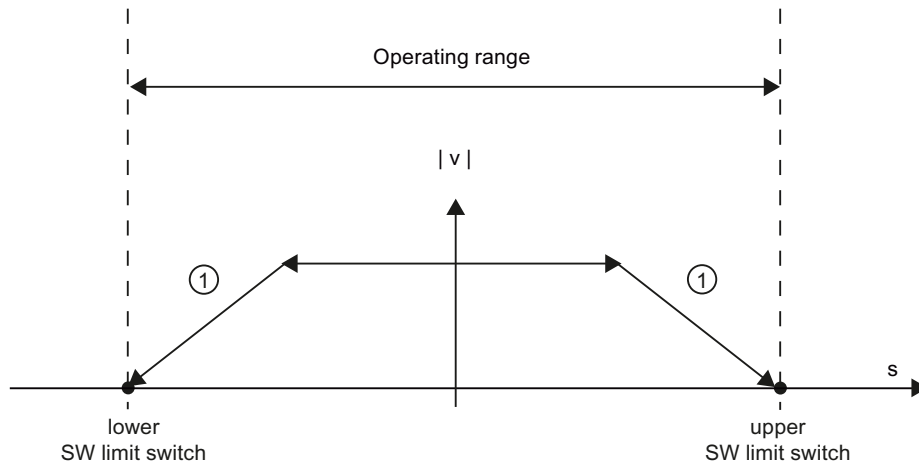
①	The axis brakes to a standstill at the configured emergency stop deceleration.
②	Range in which the HW limit switches signal the status "approached".

The "HW limit switch approached" error is displayed in the motion control instruction to be initiated, in "MC_Power", and in the technology object tags. Instructions for eliminating errors can be found in the Appendix under "List of ErrorIDs and ErrorInfos".

Behavior of axis when software limit switches are reached

If software limit switches are activated, an active motion is stopped at the position of the software limit switch. The axis is braked at the configured deceleration.

The following diagram presents the behavior of the axis until it reaches the software limit switches:



① The axis brakes to a standstill at the configured deceleration.

The "SW limit switch reached" error is displayed in the motion control instruction to be initiated, in "MC_Power", and in the technology object tags. Instructions for eliminating errors can be found in the Appendix under "List of ErrorIDs and ErrorInfos".

The circumstances under which the "SW limit switch exceeded" error is displayed can be obtained in the topics "Software limit switches in conjunction with a homing operation (Page 112)" and "Software limit switches in conjunction with dynamic changes (Page 117)".

Use additional hardware limit switches if a mechanical endstop is located after the software limit switches and there is a risk of mechanical damage.

See also

Requirements for hardware limit switches (Page 35)

Configuration - Position limits (Page 35)

Changing the position limits configuration in the user program (Page 39)

Changing the position limits configuration in the user program

You can change the following configuration parameters during user program runtime in the CPU:

Hardware limit switches

You can also activate and deactivate the hardware limit switches during runtime of the user program. Use the following technology object tag for this purpose:

- <Axis name>.Config.PositionLimits_HW.Active

Please refer to the description of the technology object tags in the Appendix for information on when changes to the configuration parameter become effective.

Software limit switches

You can also activate and deactivate the software limit switches and change their position values during runtime of the user program. Use the following technology object tags for this purpose:

- <Axis name>.Config.PositionLimits_SW.Active
for activating and deactivating the software limit switches
- <Axis name>.Config.PositionLimits_SW.MinPosition
for changing the position of the low software limit switch
- <Axis name>.Config.PositionLimits_SW.MaxPosition
for changing the position of the high software limit switch

Refer to the description of technology object tags in the Appendix for information on when changes to the configuration parameters take effect.

See also

MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)

Requirements for hardware limit switches (Page 35)

Configuration - Position limits (Page 35)

Behavior of axis when position limits is tripped (Page 37)

Dynamics

Configuration - General dynamics

Configure the maximum velocity, the start/stop velocity, the acceleration and deceleration and the jerk limit (as of technology object "Axis" V2.0) in the "General dynamics" configuration window.

Velocity limiting unit

Select the unit of measurement with which you want to set the velocity limits in the dropdown list. The unit set here depends on the unit of measurement set under "Configuration - General" and serves only for easier input.

Maximum velocity / Start/stop velocity

Define the maximum permissible velocity and the start/stop velocity of the axis in these boxes. The start/stop velocity is the minimum permissible velocity of the axis.

Limit values:

The limits indicated below refer to the "Pulses/s" unit of measurement:

- **Technology object Axis V2.0**
 - $2 \leq \text{start/stop velocity} \leq 20000$ (signal board 20kHz)
 - $2 \leq \text{start/stop velocity} \leq 200000$ (signal board 200kHz)
 - $2 \leq \text{start/stop velocity} \leq 100000$ (on-board CPU outputs)
 - $2 \leq \text{maximum velocity} \leq 20000$ (signal board 20kHz)
 - $2 \leq \text{maximum velocity} \leq 200000$ (signal board 200kHz)
 - $2 \leq \text{maximum velocity} \leq 100000$ (on-board CPU outputs)
- **Technology object Axis V1.0**
 - $2 \leq \text{start/stop velocity} \leq 20000$ (signal board 20kHz)
 - $2 \leq \text{start/stop velocity} \leq 100000$ (signal board 200kHz)
 - $2 \leq \text{start/stop velocity} \leq 100000$ (on-board CPU outputs)
 - $2 \leq \text{maximum velocity} \leq 20000$ (signal board 20kHz)
 - $2 \leq \text{maximum velocity} \leq 100000$ (signal board 200kHz)
 - $2 \leq \text{maximum velocity} \leq 100000$ (on-board CPU outputs)

The value of the maximum velocity must be greater or equal to the value of the start/stop velocity.

The limit values for other units of measurement must be converted by the user to conform to the given mechanics.

Acceleration / Delay - Ramp-up time / Ramp-down time

Set the desired acceleration in the "Ramp-up time" or "Acceleration" boxes. The desired deceleration can be set in the "Deceleration time" or "Deceleration" boxes.

The relation between the ramp-up time and acceleration and the deceleration time and deceleration is shown in the following equations:

$$\text{Rampup time} = \frac{\text{Maximum velocity} - \text{Start/stop velocity}}{\text{Acceleration}}$$

$$\text{Deceleration time} = \frac{\text{Maximum velocity} - \text{Start/stop velocity}}{\text{Deceleration}}$$

Motion jobs started in the user program are performed with the selected acceleration / deceleration.

Limit values:

The limits indicated below refer to the "Pulses/s²" units of measurement:

- $0.28 \leq \text{acceleration} \leq 9.5e9$
- $0.28 \leq \text{deceleration} \leq 9.5e9$

The limits for other units of measurement must be converted to conform to the given mechanics.

Note

Changes to the velocity limits ("start/stop velocity" and "maximum velocity") influence the acceleration and deceleration values of the axis. The ramp-up and deceleration times are retained.

Activate jerk limit (as of technology object Axis V2.0)

Activate the jerk limit with this check box.

Note

If an error occurs, the axis decelerates with the configured emergency stop deceleration. An activated jerk limit is not considered here.

Smoothing time/jerk (as of technology object "Axis" V2.0)

You can input the parameters of the jerk limit in the "Smoothing time" field or alternatively in the "Jerk" field.

- Set the desired jerk for acceleration and deceleration ramp in the "Jerk" field.
- Set the desired smoothing time for the acceleration ramp in the "Rounding time" field.

Note

The set smoothing time visible in the configuration only applies to the acceleration ramp.

If the values for acceleration and deceleration differ, the smoothing time of the deceleration ramp is calculated according to the jerk of the acceleration ramp and used. (See also Behavior of the axis when using the jerk limit (Page 44))

The smoothing time of the deceleration is adapted as follows:

- **Acceleration > deceleration**

The smoothing time used for the deceleration ramp is shorter than that for the acceleration ramp.

- **Acceleration < deceleration**

The smoothing time used for the deceleration ramp is shorter than that for the acceleration ramp.

- **Acceleration = deceleration**

The smoothing times of the acceleration and deceleration ramp are equal.

The relation between smoothing times and jerk is shown in the following equation:

$$\text{Rounding off time (acceleration ramp)} = \frac{\text{Acceleration}}{\text{Step}}$$

$$\text{Rounding off time (deceleration ramp)} = \frac{\text{Deceleration}}{\text{Step}}$$

Motion jobs started in the user program are performed with the selected jerk.

Limit values:

The limits indicated below refer to the Pulses/s³ units of measurement:

- $0.04 \leq \text{jerk} \leq 1.5e8$

The limits for other units of measurement must be converted to conform to the given mechanics.

See also

Behavior of the axis when using the jerk limit (Page 44)

Configuration - Dynamics emergency stop (Page 43)

Changing the configuration of dynamics in the user program (Page 46)

Configuration - Dynamics emergency stop

Configure the emergency stop deceleration of the axis in the "Dynamics emergency stop" configuration window. When an error occurs and when the axis is disabled with motion control instruction "MC_Power" (input parameter StopMode = 0), the axis is brought to a standstill with this deceleration.

Velocity limits

The velocity values configured in the "General dynamics" configuration window are once again displayed in this information area.

Deceleration

Set the deceleration value for emergency stop in the "Emergency stop deceleration" or "Emergency stop ramp-down time" field.

The relation between emergency stop deceleration time and emergency stop deceleration is shown in the following equation:

$$\text{Emergency stop deceleration time} = \frac{\text{Maximum velocity} - \text{Start/stop velocity}}{\text{Emergency stop deceleration}}$$

The specified emergency stop deceleration must be sufficient to bring the axis to a standstill in a timely manner in the event of an emergency (for example, when the hardware limit switch is approached prior to reaching the mechanical endstop).

The configured maximum velocity of the axis must be used as a basis for selecting the emergency stop deceleration.

Limit values:

The limits indicated below refer to the "Pulses/s²" units of measurement:

- $0.28 \leq \text{emergency stop deceleration} \leq 9.5e9$

The limits for other units of measurement must be converted to conform to the given mechanics.

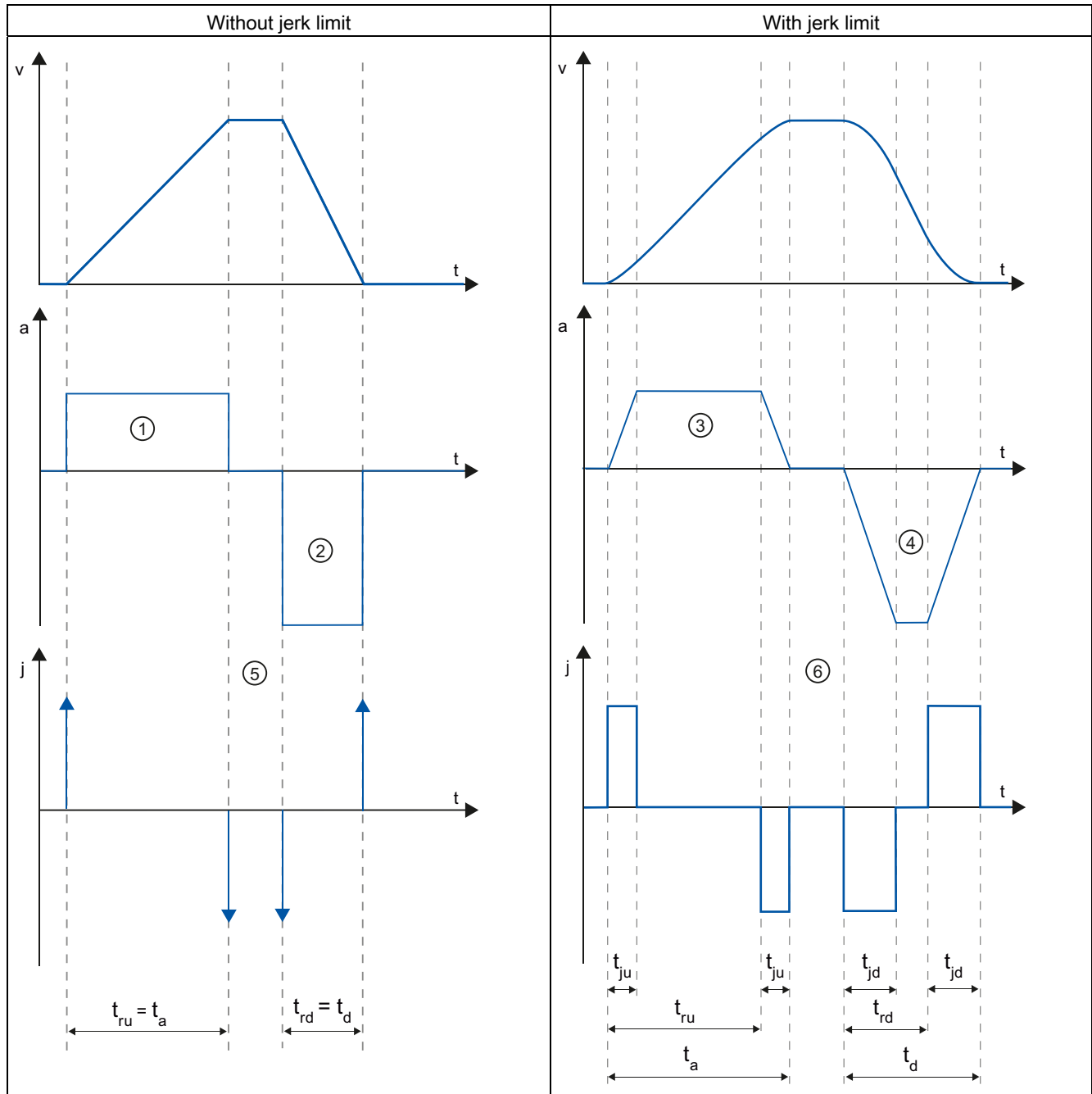
See also

Configuration - General dynamics (Page 40)

Changing the configuration of dynamics in the user program (Page 46)

Behavior of the axis when using the jerk limit

Axis acceleration and deceleration is not stopped abruptly when the jerk limit is activated; it is adjusted gently according to the set step or rounding off time. The diagram below details the behavior of the axis with and without activated jerk limit:



t	Time axis
v	Velocity
a	Acceleration
j	Step
t _{ru}	Rampup time
t _a	Time taken for the axis to accelerate
t _{rd}	Deceleration time
t _d	Time taken for the axis to decelerate
t _{ju}	Smoothing time of the acceleration ramp
t _{jd}	Smoothing time of the deceleration ramp

The example shows travel in which the deceleration value ② is twice the acceleration value ①. The resulting ramp-down time t_{rd} is therefore only half the length of the ramp-up time t_{ru}.

Acceleration ① and deceleration ② change abruptly without a jerk limit. Acceleration ① and deceleration ② change gradually with activated jerk limiter. As the jerk applies to entire motion, the rate is the same for the increase and decrease in acceleration and deceleration.

The step value j becomes infinitely high ⑤ as soon as the change is made without jerk limit. The step is limited to the configured value ⑥ when the jerk limit is activated.

The smoothing time t_{ju} given in the configuration applies to the acceleration ramp. The deceleration ramp smoothing time t_{jd} is calculated using the configured jerk value and the configured deceleration.

See also

Configuration - General dynamics (Page 40)

Changing the configuration of dynamics in the user program

You can change the following configuration parameters during user program runtime in the CPU:

Acceleration and deceleration

You can also change the values for acceleration and deceleration during runtime of the user program. Use the following technology object tags for this purpose:

- `<Axis name>.Config.DynamicDefaults.Acceleration`
for changing acceleration
- `<Axis name>.Config.DynamicDefaults.Deceleration`
for changing deceleration

Refer to the description of technology object tags in the Appendix for information on when changes to the configuration parameters take effect.

Emergency stop deceleration

You can also change the value for the emergency stop deceleration during runtime of the user program. Use the following technology object tag for this purpose:

- `<Axis name>.Config.DynamicDefaults.EmergencyDeceleration`

Please refer to the description of the technology object tags in the Appendix for information on when changes to the configuration parameter become effective.



WARNING

After changes to this parameter, it may be necessary to adapt the positions of the hardware limit switches and other safety-relevant settings.

Jerk limit (as of technology object "Axis" V2.0)

You can also activate and deactivate the jerk limit at runtime of the user program and change the value for the jerk. Use the following technology object tag for this purpose:

- `<Axis name>.Config.DynamicDefaults.JerkActive`
for activating and deactivating the jerk limit
- `<Axis name>.Config.DynamicDefaults.Jerk`
for changing the jerk

Please refer to the description of the technology object tags in the Appendix for information on when changes to the configuration parameter become effective.

See also

MC_ChangeDynamic: Changing the dynamic settings for the axis
(as of technology object "Axis" V2.0) (Page 194)

Configuration - General dynamics (Page 40)

Configuration - Dynamics emergency stop (Page 43)

Homing (technology object "Axis" as of V2.0)

Configuration - Homing - General

Configure the reference point switch input for active and passive homing in the "Homing - General" configuration window.

Reference point switch input

Select the digital input for the reference point switch from the drop-down list box. The input must be able to generate an interrupt. The onboard CPU inputs and inputs of an inserted signal board can be selected as inputs for the reference point switch.

Note

The digital inputs are set to a filter time of 6.4 ms by default.

When the digital inputs are used as a reference point switch, this can result in undesired decelerations and thus inaccuracies. Depending on the homing velocity and extent of the reference point switch, the reference point may not be detected. The filter time can be set under "Input filter" in the device configuration of the digital inputs.

The specified filter time must be less than the duration of the input signal at the reference point switch.

See also

Sequence - Active homing (Page 51)

Configuration - Homing - Passive

Configure the necessary parameters for passive homing in the "Homing - Passive" configuration window.

The movement for passive homing must be triggered by the user (e.g. using an axis motion command). Passive homing is started using Motion Control instruction "MC_Home" with input parameter "Mode" = 2.

Side of the reference point switch

This is where you select whether the axis is to be homed on the low or high side of the reference point switch.

Reference point position

The position for which parameters were assigned in the Motion Control instruction "MC_Home" is used as the reference point position.

Note

If passive homing is carried out without an axis motion command (axis at a standstill), homing will be executed upon the next rising or falling edge at the reference point switch.

Configuration - Homing - Active

Configure the necessary parameters for active homing in the "Active homing" configuration window. Active homing is started using Motion Control instruction "MC_Home" with input parameter "Mode" = 3.

Permit auto reverse at the hardware limit switch

Activate the check box to use the hardware limit switch as a reversing cam for the reference point approach. The hardware limit switches must be enabled for the reversal of direction (at least the hardware limit switch in the direction of approach must be configured).

If the hardware limit switch is reached during active homing, the axis brakes at the configured deceleration (not with the emergency stop deceleration) and reverses direction. The homing switch is then sensed in reverse direction.

If the direction reversal is not active and the axis reaches the hardware limit switch during active homing, the reference point approach is aborted with an error and the axis is braked at the emergency stop deceleration.

NOTICE
If possible, use one of the following measures to ensure that the machine does not travel to a mechanical endstop in the event of a direction reversal: <ul style="list-style-type: none">• Keep the approach velocity low• Increase the configured acceleration/deceleration• Increase the distance between hardware limit switch and mechanical stop

Approach/homing direction

With the direction selection, you determine the approach direction used during active homing to search for the homing switch, as well as the homing direction. The homing direction specifies the travel direction the axis uses to approach the configured side of the homing switch to carry out the homing operation.

Side of the homing switch

This is where you select whether the axis is to be homed on the low or high side of the homing switch.

Velocity

In this field, specify the velocity at which the homing switch is to be searched for during the reference point approach.

Limits (independent of the selected unit of measurement):

- Start/stop velocity \leq approach velocity \leq maximum velocity

Homing velocity

Specify in this field the velocity at which the homing switch is to be approached for homing.

Limits (independent of the selected unit of measurement):

- Start/stop velocity \leq Homing velocity \leq Maximum velocity

Home position offset

If the desired reference position deviates from the position of the homing switch, the home position offset can be specified in this field.

If the value does not equal 0, the axis executes the following actions following homing at the homing switch:

1. Move the axis at the homing velocity by the value of the home position offset
2. Upon reaching the "Home position offset", the axis is at the home position that was specified in input parameter "Position" of the "MC_Home" Motion Control instruction.

Limits (independent of the selected unit of measurement):

- $-1.0e12 \leq$ home position offset $\leq 1.0e12$

Home position

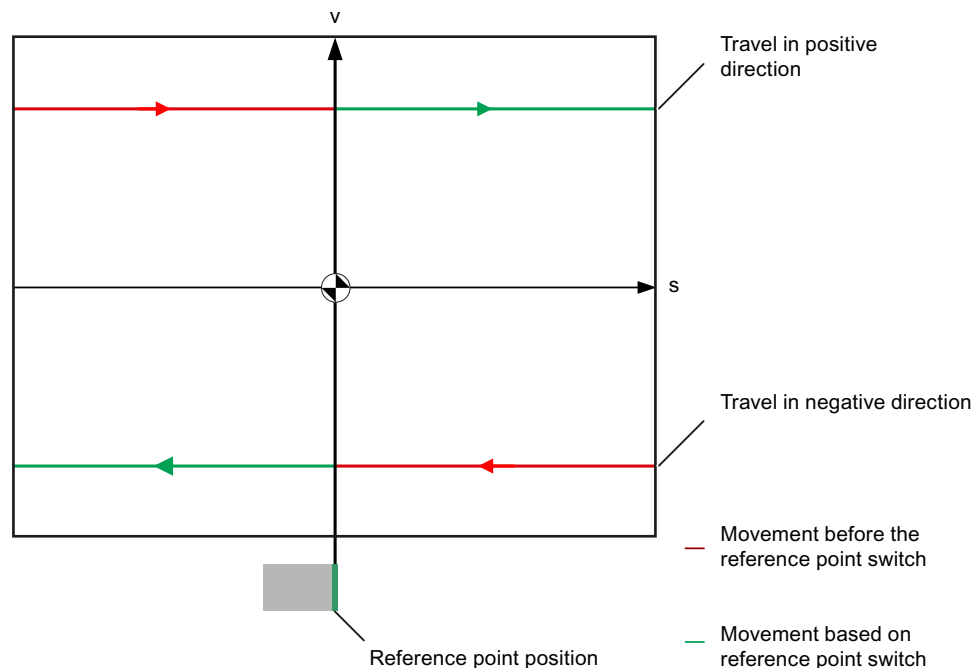
The position for which parameters were assigned in the Motion Control instruction "MC_Home" is used as the reference point position.

Sequence - Passive homing

Passive homing is started with Motion Control instruction "MC_Home" (input parameter Mode = 2). Input parameter "Position" specifies the absolute reference point position.

The diagram below shows an example of a characteristic curve for passive homing with the following configuration parameters:

- "Reference point switch side" = "High side"



Movement towards reference point switch (red section of curve)

The Motion Control instruction "MC_Home" does not itself carry out any homing motion when passive homing is started. The travel required for reaching the reference point switch must be implemented by the user via other motion control instructions such as "MC_MoveRelative". The tag <axis name>.StatusBits.HomingDone remains TRUE during passive homing if the axis has already been homed.

Axis homing (transition from red to green section of curve)

The axis is homed when the configured side of the reference point switch is reached. The current position of the axis is set to the reference point position. This is specified at the "Position" parameter of the "MC_Home" Motion Control instruction. The variable <axis name>.StatusBits.HomingDone will be set to "TRUE" if the axis has not been homed before. The travel previously started is not cancelled.

Movement beyond reference point switch (green section of curve)

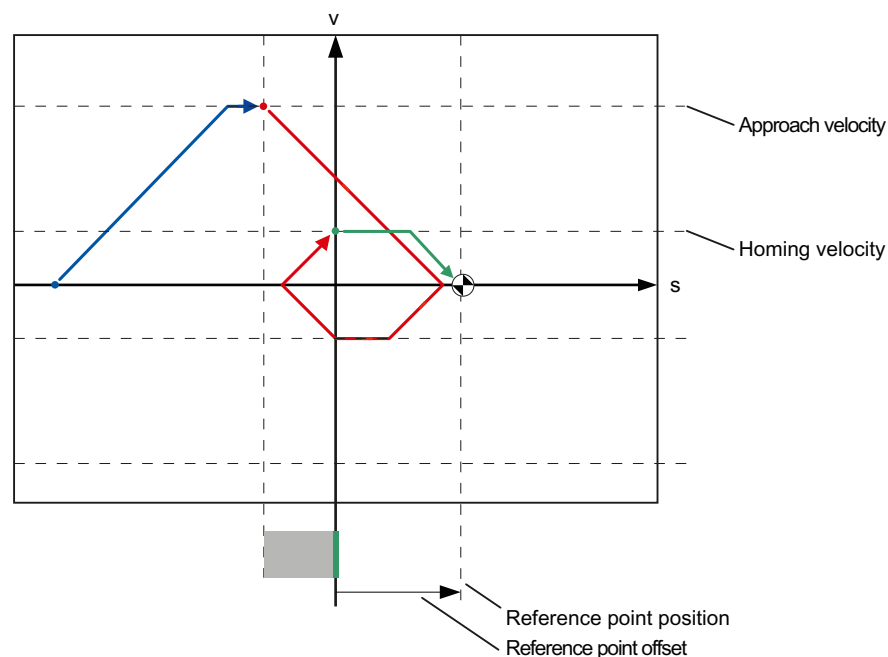
Following homing at the reference point switch, the axis continues and completes the previously started travel with the corrected axis position.

Sequence - Active homing

You start active homing with motion control instruction "MC_Home" (input parameter Mode = 3). The "Position" input parameter specifies the absolute home position. Alternatively, you can start active homing on the axis command table for test purposes.

The diagram below shows an example of a characteristic curve for an active reference point approach with the following configuration parameters:

- "Approach/homing direction" = "Positive direction"
- "Side of the homing switch" = "Top side"
- Value of "home position offset" > 0



Search for homing switch (blue curve section)

When active homing starts, the axis accelerates to the configured "approach velocity" and searches at this velocity for the homing switch. The tag <axis name>.StatusBits.HomingDone is set to FALSE.

Reference point approach (red curve section)

When the homing switch is detected, the axis in this example brakes and reverses, to be homed to the configured side of the homing switch at the configured homing velocity. Homing causes the tag <axis name>.StatusBits.HomingDone to change to TRUE.

Travel to home position offset (green curve segment)

After homing, the axis moves at the homing velocity along the path to the home position offset. There the axis is at the homing point position that was specified in input parameter "Position" of the "MC_Home" Motion Control instruction.

See also

Configuration - Homing - General (Page 47)

Changing the homing configuration in the user program

You can change the following configuration parameters in the CPU during user program runtime as of technology object "Axis" V2.0:

Passive homing

You can change the side of the homing switch for passive homing during the user program runtime. Use the following technology object tag for this purpose:

- `<Axis name>.Config.Homing.SidePassiveHoming`
for changing the side of the homing switch

Please refer to the description of the technology object tags in the Appendix for information on when changes to the configuration parameter become effective.

Active homing

You can change the direction of approach, the side of the homing switch, the approach velocity, the homing velocity, and the home position offset for active homing during the program runtime of the user program. Use the following technology object tags for this purpose:

- `<Axis name>.Config.Homing.AutoReversal`
for changing "auto reverse at the HW limit switch"
- `<Axis name>.Config.Homing.Direction`
for changing "approach / homing direction"
- `<Axis name>.Config.Homing.SideActiveHoming`
for changing the "side of the homing switch"
- `<Axis name>.Config.Homing.FastVelocity`
for changing the "velocity"
- `<Axis name>.Config.Homing.SlowVelocity`
for changing the "homing velocity"
- `<Axis name>.Config.Homing.Offset`
for changing "home position offset"

Please refer to the description of the technology object tags in the Appendix for information on when changes to the configuration parameter become effective.

See also

MC_ChangeDynamic: Changing the dynamic settings for the axis
(as of technology object "Axis" V2.0) (Page 194)

1.6 Technology object command table

1.6.1 Use of the command table technology object

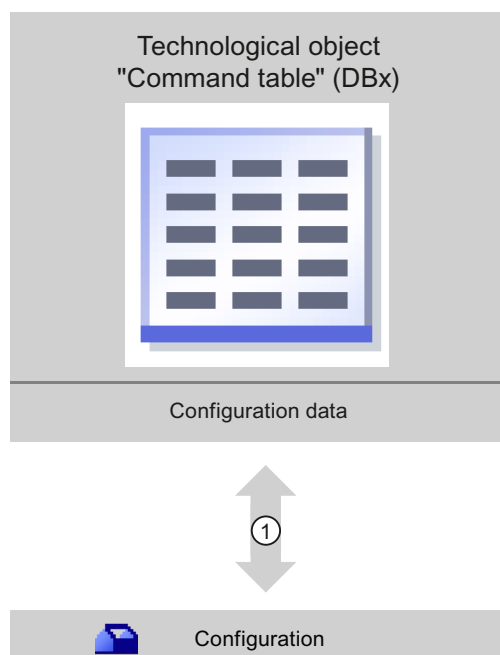
The Motion Control instruction "Command table" allows you to combine multiple individual axis control jobs in one movement sequence. The technology object can be used for technology as of Version V2.0.

You configure the movement sequence as a table in a configuration dialog.

The motion profile of the movement sequences can be checked on a graph before the project is loaded to the CPU. The command tables created are then linked to an axis and used in the user program with the "MC_CommandTable" Motion Control instruction. You can process part or all of the command table.

1.6.2 Command table technology object tools

The "Configuration" tool is provided in the TIA Portal for the "Command Table" technology object. The representation below shows the interaction of the tool with the technology object:



①	Writing and reading the configuration of the technology object
---	--

Configuration

Configure the following properties of the "Command Table" technology object with the "Configuration" tool:

- You can create one or more movement sequences by configuring individual jobs.
- You can configure the graphic display to check your movement sequence using an axis already configured or a configurable default axis.

The movement sequence data are saved in the data block of the technology object.

1.6.3 Adding the technological object command table

Proceed as follows to add a "Command table" technology object in the project tree:

Prerequisites

- A project with a CPU S7-1200 has been created.
- The CPU firmware version is V2.1 or higher

Procedure

1. Open the CPU folder in the project tree.
2. Open the technology objects folder.
3. Double-click "Add new object".
The "Add new object" dialog opens.
4. Select the "Motion" technology.
5. Open the "Motion Control" folder.
6. Open the "S7-1200 Motion Control" folder
7. Select the version "V2.0" of the "S7-1200 Motion Control" folder (click on the entry for the version).
8. Select the "TO_CommandTable" object.
9. Change the name of the command table in the "Name" input field to suit your needs.
10. Select the "Manual" option if you want to change the suggested data block number.
11. Click "More information" if you want to supplement user information for the technology object.
12. Click "OK" to add the technology object.
Click "Cancel" to discard your entries.

Result

The new technology object is created and saved to the "Technology objects" folder in the project tree.

1.6.4 Configuring the command table technology object

1.6.4.1 Working with the configuration dialog

You configure the properties of the technology object in the configuration window. Proceed as follows to open the configuration window of the technology object:

1. Open the group of the required technology object in the project tree.
2. Double-click the "Configuration" object.

The configuration is divided into the following categories:

- **Basic parameters**





The basic parameters contain all parameters which must be configured for a functional command table.

- **Extended parameters**

The extended parameters contain the parameters of the default axis or display the parameter values of the axis selected.

Icons of the configuration window

Icons in the area navigation of the configuration show additional details about the status of the configuration:

	<p>The configuration contains default values and is complete.</p> <p>The configuration contains only default values. With these default values, you can use the technology object without additional changes.</p>
	<p>The configuration contains values set by the user and is complete.</p> <p>All input fields of the configuration contain valid values and at least one preset value has changed.</p>
	<p>The configuration is incomplete or incorrect</p> <p>At least one input field or drop-down list contains an invalid value. The corresponding field or the drop-down list is displayed on a red background. Click the roll-out error message to display the cause of the error.</p>
	<p>The configuration contains mutually incompatible parameter values</p> <p>The configuration contains parameter values that contradict each other either in size or logic. The corresponding field or the drop-down list is displayed on a yellow background.</p>

See also

Guidelines on use of motion control (Page 21)

Basic parameters (Page 56)

Extended parameters (Page 71)

1.6.4.2 Basic parameters

Configuration - General

Configure the name of the technology object in the "General" configuration window.

Name

Define the name of the command table or the name of the "Command table" technology object in this field. The technology object will be listed under this name in the project tree.

See also

Configuration - Command table (Page 56)

Shortcut menu commands - Command table (Page 60)

Working with the trend diagram (Page 62)

Shortcut menu commands - Curve chart (Page 67)

Transition from "Complete command" to "Blend motion" (Page 68)

Changing the command table configuration in the user program (Page 70)

Configuration - Command table

Create the desired movement sequence in the "Command Table" configuration window and check the result against the graphic view in the trend diagram.

Note

Small deviations are possible between the time behavior and position in the trend shown and the real movement of the axis. Movements in response to software limit switches being reached are not shown.

Activate warnings

Activate the display of warnings in the command table with this checkbox.

Use axis parameters of

From the drop-down list, select which axis parameters are to be used for selecting the graphic view of and checking the movement sequence. Select "Default axis" if you have yet to add an axis to the "Technology object" folder or wish to use value which have not been configured in any of the available axes. You configure the properties of the default axis under "Advanced parameters".

The axis parameters of the axis selected at the "Axis" parameter will be used to process the command table in the user program.

Column: Step

Shows the step number of the command.

Column: Command type

In this column, select the command types which are to be used for processing the command table. Up to 32 jobs can be entered. The commands will be processed in sequence. You can choose between the following entries and command types:

- **Empty**
The entry serves as a placeholder for any commands to be added. The empty entry is ignored when the command table is processed.
- **Halt**
Pause axis
(the command only takes effect after a "Velocity setpoint" command)
- **Positioning Relative**
Position axis relatively
- **Positioning Absolute**
Position axis absolutely
- **Velocity setpoint**
Move axis at set velocity
- **Wait**
Waits until the given period is over. Wait does not stop an active traversing motion.
- **Separator**
Adds a Separator line above the selected line. The Separator line acts as a range limit for the graphic display of the trend view.

Use the Separator lines if you wish to process parts of the command table.

Column: Position

Enter the position or travel path for the selected command in this column:

- **Command "Positioning relative"**
The command will move the axis by the the given travel path.
- **Command "Positioning absolute"**
The command will move the axis by the the given position.
- **Separator**
The value given specifies the start position for the graphic display.

Limit values (independent of the selected user unit):

- $-1.0e12 \leq \text{position / distance} \leq -1.0e-12$
- $1.0e-12 \leq \text{position / distance} \leq 1.0e12$
- Position / travel path = 0.0

Column: Velocity

In this column, you enter the velocity for the selected command:

- **Command "Positioning relative"**
The command will move the axis at the the given velocity.
The given velocity will not be reached if the travel path selected is not large enough.
- **Command "Positioning absolute"**
The command will move the axis at the the given velocity.
The given velocity will not be reached if the target position is too close to the starting position.
- **Command " Velocity setpoint"**
The command will move the axis at the the given velocity.
The given velocity will not be reached during the command if too short a runtime is selected.

Limit values (independent of the selected user unit):

- For the jobs: "Positioning relative" and "Positioning absolute"
 - $1.0e-12 \leq \text{velocity} \leq 1.0e12$
- For the command: "Velocity setpoint"
 - $-1.0e12 \leq \text{velocity} \leq -1.0e-12$
 - $1.0e-12 \leq \text{velocity} \leq 1.0e12$
 - Velocity = 0.0

Column: Duration

Enter the duration of the selected command in this column:

- **Command " Velocity setpoint"**

The command will move the axis for the specified duration. The duration includes both the acceleration phase and the constant travel phase. The next command will be processed once the duration is over.

- **Command "Wait"**

Waits until the given duration is over.

Limit values (independent of the selected user unit):

- $0.001s \leq \text{duration} \leq 64800s$

Column: Next step

Select the mode of transition to the next step from the drop-down list:

- **Complete command**

The command will be completed. The next command will be processed immediately.

- **Blend motion**

The motion of the current command will be blended with the motion of the following command. The transition mode "Blend motion" is available with command types "Positioning Relative" and "Positioning Absolute".

Motion will be blended with motions of the following command types:

- Positioning Relative
- Positioning Absolute
- Velocity setpoint

No blending occurs with other command types.

For the exact behavior of the axis when a command is appended or overlapped, see: Transition from "Complete command" to "Blend motion" (Page 68)

Column: Step code

Enter a numerical value / bit pattern in this column which is to be output at the "StepCode" output parameter of the "MC_CommandTable" Motion Control instruction while the command is being processed.

Limit values:

- $0 \leq \text{code number} \leq 65535$

See also

Configuration - General (Page 56)

Shortcut menu commands - Command table (Page 60)

Working with the trend diagram (Page 62)

Shortcut menu commands - Curve chart (Page 67)

Transition from "Complete command" to "Blend motion" (Page 68)

Changing the command table configuration in the user program (Page 70)

Shortcut menu commands - Command table

The following shortcut menu commands are available in the command table:

Insert empty line

Adds an empty line above the selected line.

This shortcut menu command can only be executed if there are enough empty lines at the end of the command table.

Add empty line

Adds an empty line below the selected line.

This shortcut menu command can only be executed if there are enough empty lines at the end of the command table.

Insert separator line

Adds a separator line above the selected line.

You cannot have two consecutive separator lines.

Add separator line

Adds a separator line below the selected line.

You cannot have two consecutive separator lines, nor can you add a separator line at the end of the command table.

Cut

Removes the selected lines or content of the selected cell and saves them/it in the clipboard.

Selected lines will be deleted and the subsequent lines of the command table shifted up.

Copy

Copies the selected lines or content of the selected cell and saves them/it in the clipboard.

Paste

- Selected lines:
Pastes the lines from the clipboard into the table above the selected line.
- Selected cell:
Pastes the content of the clipboard into the selected line.

This shortcut menu command can only be executed if there are enough empty lines at the end of the command table.

Replace

Replaces the selected lines with the lines in the clipboard.

Delete

Deletes the selected lines. The lines below in the command table shift up.

See also

Configuration - General (Page 56)

Configuration - Command table (Page 56)

Working with the trend diagram (Page 62)

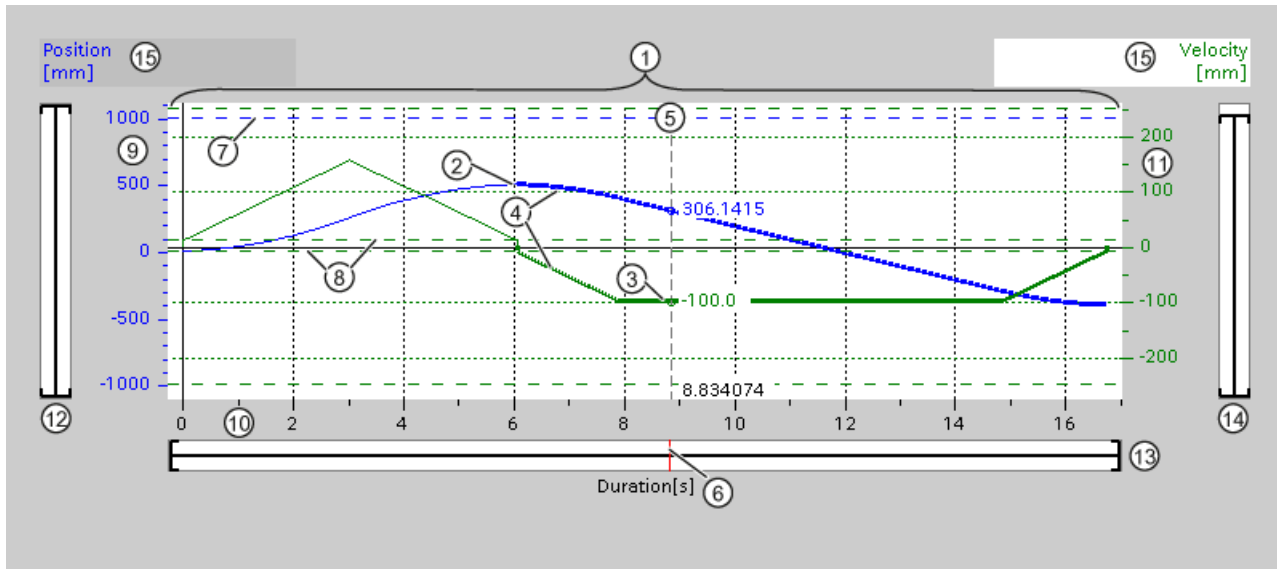
Shortcut menu commands - Curve chart (Page 67)

Transition from "Complete command" to "Blend motion" (Page 68)

Changing the command table configuration in the user program (Page 70)

Working with the trend diagram

Trend view and components



①	Trend view
②	Position curve
③	Velocity curve
④	Curve section of a selected command
⑤	Ruler
⑥	Ruler position marking
⑦	Software limit switch position
⑧	Start/stop velocity
⑨	Position axis scale range
⑩	Time axis scale range
⑪	Velocity axis scale range
⑫	Scroll bar, position axis
⑬	Scroll bar time axis
⑭	Scroll bar, velocity axis
⑮	Selecting the grid

Selecting separator sections

If the command table consists of multiple sections separated by separators, you can select these sections in the trend view by selecting a command in the section.

Selecting commands

Commands can be selected in the trend view and in the command table:

- Click on a point on the velocity or position curve in the trend view.
The corresponding command will be highlighted in the command table.
- Select a command in the command table.
The corresponding section of curve will be highlighted.

Selecting the visible range of the trend view

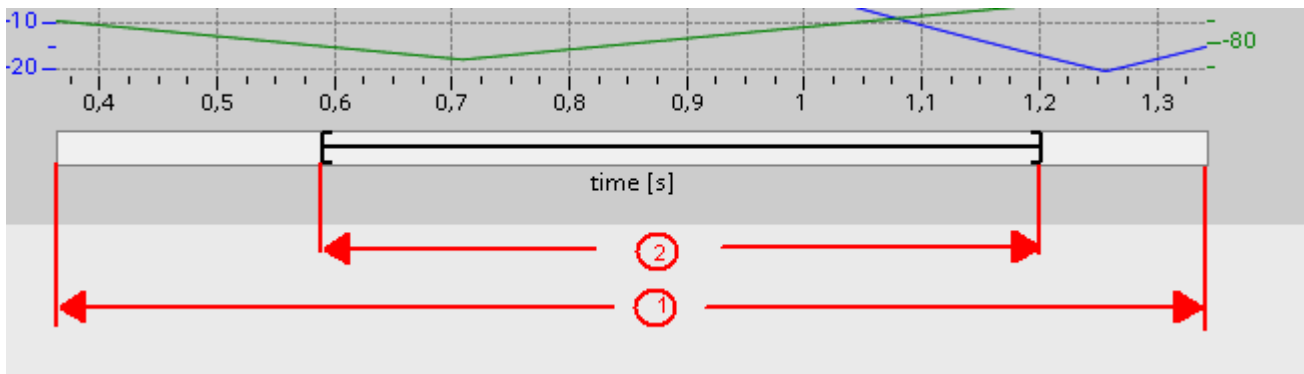
Follow the steps below to adjust the section of the trend view to be displayed:

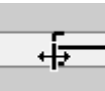

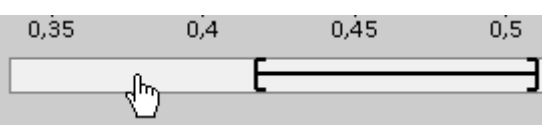
Select the scaling in the shortcut menu:

- Scale to curves:
 - Scales the axes so the position and velocity curves are visible.
- Scale to curves and limits:
 - Scales the axes so the position and velocity curves, the positions of the activated software limit switches and the minimum and maximum velocity limits are visible.

The view selected will be marked in the shortcut menu with a tick.

Selecting the section to be shown within the range:

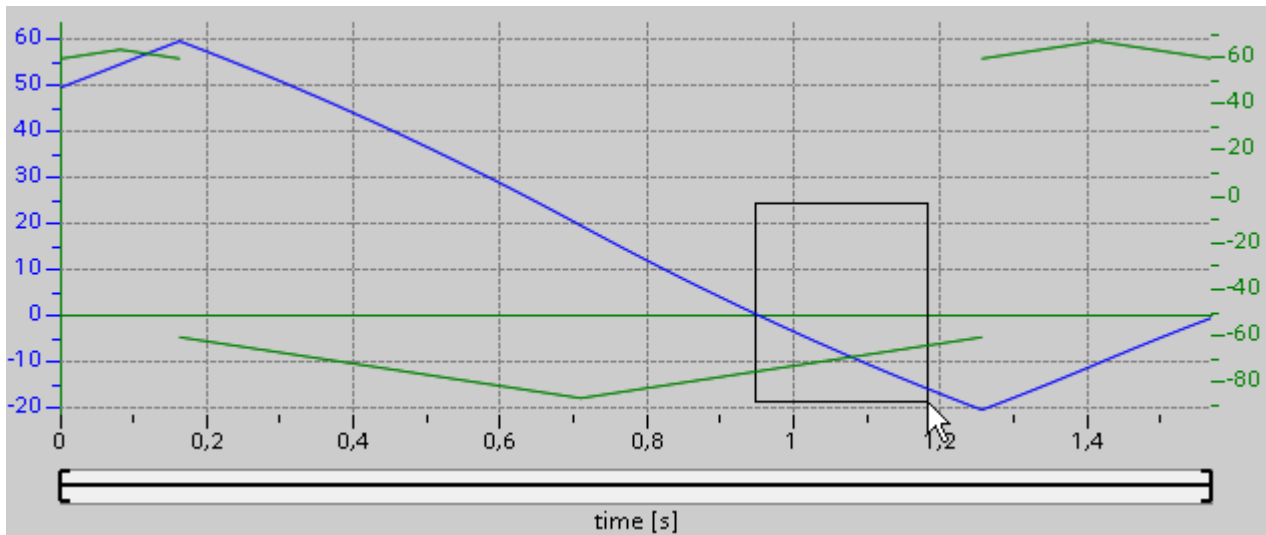


①	Range which the curve values and / or limits are within. (see Selecting in the shortcut menu)
②	<p>Selected range to be shown in the trend window. You set the range with the margin cursor at the right-hand and left-hand margin.</p>  <p>You set the position within range ① with the drag cursor.</p>  <p>You can also define the position by clicking in range ①.</p> 

Selecting the section to be shown with the mouse:

Drag a section of the trend view by clicking and dragging with the mouse.

The section of curve selected will be enlarged once you release the mouse.



Undoing the last change to the section:

Select the shortcut command "Undo zoom" to undo the last change to the section.

Synchronizing the grid

Click on the axis scales to select whether the grid is to be synchronized with the position axis or velocity axis.

Reading off curve values from the ruler

Activate the ruler using the shortcut menu command "Show ruler".

You can move the ruler to any point on the curves using the ruler cursor.



See also

Configuration - General (Page 56)

Configuration - Command table (Page 56)

Shortcut menu commands - Command table (Page 60)

Shortcut menu commands - Curve chart (Page 67)

Transition from "Complete command" to "Blend motion" (Page 68)

Changing the command table configuration in the user program (Page 70)

Shortcut menu commands - Curve chart

The following shortcut menu commands are available in the curve window:

Zoom 100%

Selects a zoom factor which will show 100% of the curve values and / or limits.

Undo zoom

Undoes the last zoom change.

Scale to curves

Scales the axes so the position and velocity curves are visible.

Scale to curves and limits

Scales the axes so the position and velocity curves, the positions of the activated software limit switches and the minimum and maximum velocity limits are visible.

Show velocity limits

Shows the lines of the velocity limits.

Show software limit switches

Shows the lines of the software limit switches.

Show ruler

Fades the ruler in / out

Use the ruler when you want to see the individual values of the curves.

See also

Configuration - General (Page 56)

Configuration - Command table (Page 56)

Shortcut menu commands - Command table (Page 60)

Working with the trend diagram (Page 62)

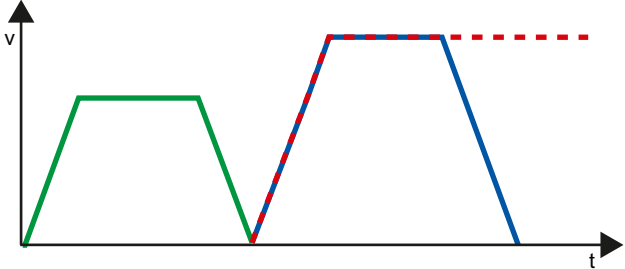
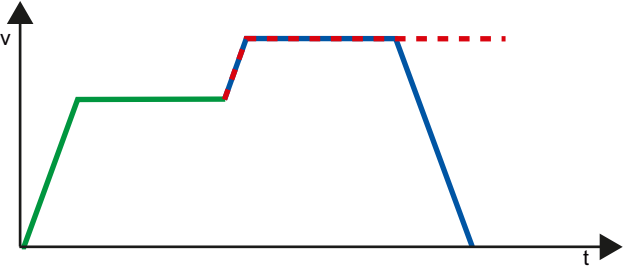
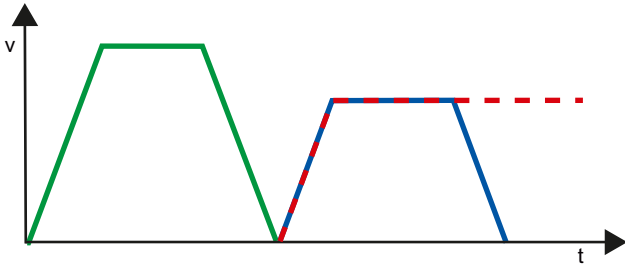
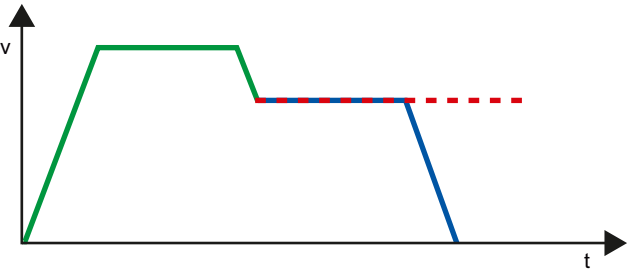
Transition from "Complete command" to "Blend motion" (Page 68)




Changing the command table configuration in the user program (Page 70)

Transition from "Complete command" to "Blend motion"

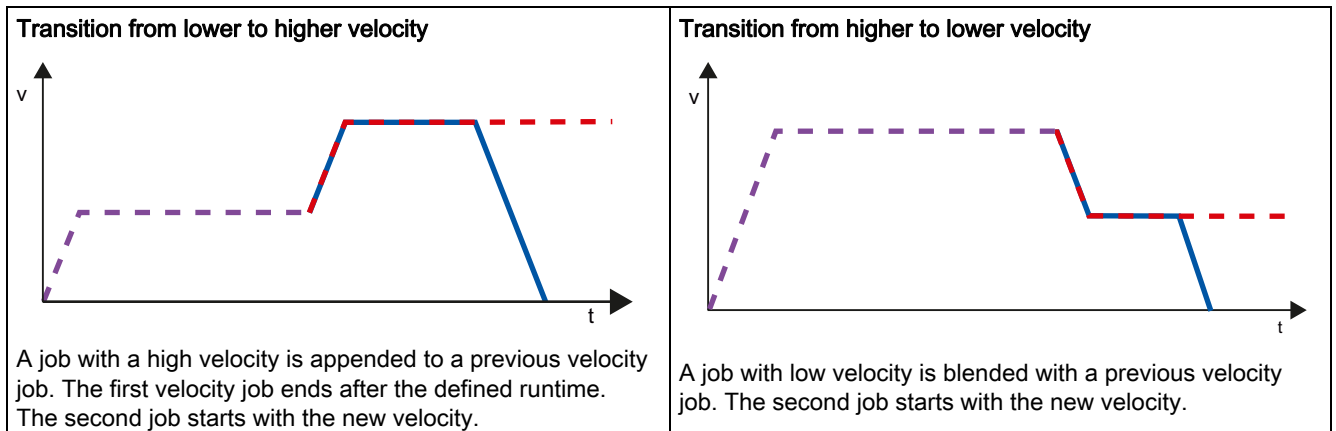
The charts below show the transition between movements in various different transition modes in the "Next step" column:




Motion transition with preceding positioning jobs

Complete job	Blend motion
<p>Transition from lower to higher velocity</p>  <p>A job with high velocity is appended to a previous positioning job. The positioning job terminates at its target position at velocity "0". The second job starts from standstill.</p>	<p>Transition from lower to higher velocity</p>  <p>A job with high velocity is overlapped with a previous positioning job. The first positioning job terminates without standstill at its target position. The second job starts with the new velocity.</p>
<p>Transition from higher to lower velocity</p>  <p>A job with low velocity is appended to a previous positioning job. The positioning job terminates at its target position at velocity "0". The second job starts from standstill.</p>	<p>Transition from higher to lower velocity</p>  <p>A job with low velocity is overlapped with a previous positioning job. The first positioning job terminates without standstill at its target position. The first job starts with the new velocity.</p>

	<p>1. Job "Positioning Relative" or "Positioning Absolute"</p>
	<p>2. Job "Velocity set point"</p>
	<p>2. Job "Positioning Relative" or "Positioning Absolute"</p>

Motion transition with preceding velocity jobs



	1. Job "Velocity set point"
	2. Job "Velocity set point"
	2. Job "Positioning Relative" or "Positioning Absolute"

See also

- Configuration - General (Page 56)
- Configuration - Command table (Page 56)
- Shortcut menu commands - Command table (Page 60)
- Working with the trend diagram (Page 62)
- Shortcut menu commands - Curve chart (Page 67)
- Changing the command table configuration in the user program (Page 70)

Changing the command table configuration in the user program

You can change the following configuration parameters during user program runtime in the CPU:

Jobs and corresponding values

You can also change the parameters of the command table during the runtime of the user program. Use the following technology object variables for this purpose:

- `<Table name>.Config.Commands[1..32].Command`
for changing the command type
- `<Table name>.Config.Commands[1..32].Position`
for changing the position / travel path
- `<Table name>.Config.Commands[1..32].Velocity`
for changing the velocity
- `<Table name>.Config.Commands[1..32].Duration`
for changing the duration
- `<Table name>.Config.Commands[1..32].BufferMode`
for changing the parameter "Next step"
- `<Table name>.Config.Commands[1..32].Code`
for changing the step code

Refer to the description of technology object variables in the Appendix for information on when changes to the configuration parameters take effect.

See also

Configuration - General (Page 56)

Configuration - Command table (Page 56)

Shortcut menu commands - Command table (Page 60)

Working with the trend diagram (Page 62)

Shortcut menu commands - Curve chart (Page 67)

Transition from "Complete command" to "Blend motion" (Page 68)

1.6.4.3 Extended parameters

Chart parameters

Configuration - General

Configure the basic properties of the chart view of the "Command table" technology object in the "General" configuration window.

Note

If the default axis has been selected under "Use axis parameters of", the unit of measurement can be edited. If a configured axis has been selected, the unit of measurement for this axis will be displayed.

Use axis parameters of

From the drop-down list, select which axis parameters are to be used for selecting the graphic view of and checking the movement sequence. Select "Default axis" if you have yet to add an axis to the "Technology object" folder or wish to use values which have not been configured in any of the available axes.

The axis parameters of the axis selected at the "Axis" parameter will be used to process the command table in the user program.

Unit of measurement

Enter the unit of measurement for the default axis in this field. If a preconfigured axis has been selected under "Use axis parameters of", the unit of measurement configured in these parameter will be displayed.

Configuration - Dynamics

Configure the acceleration and deceleration and the jerk limit for the default axis in the "Dynamics" configuration window.

Note

If the default axis has been selected under "Use axis parameters of", the following fields can be edited. If a configured axis has been selected, the values of this axis will be displayed.

Acceleration / deceleration

Set the desired acceleration of the default axis in the "Acceleration" field.
The desired deceleration can be set in the "Deceleration" field.

Motion jobs configured in the command table will be calculated with the selected acceleration / deceleration.

Limit values:

- $1.0e-12 \leq \text{acceleration} \leq 1.0e12$
- $1.0e-12 \leq \text{deceleration} \leq 1.0e12$

Activate jerk limit

Activate the jerk limit with this checkbox.

Step

Set the desired step for ramping up and ramping down in the "Step" field.

Motion jobs configured in the command table will be calculated with the selected step.

Limit values:

- $1.0e-12 \leq \text{jerk} \leq 1.0e12$

Configuration - Limit values

Configure the maximum velocity, the start/stop velocity and the software limit switches of the default axis in the "Limits" configuration window.

Note

If the default axis has been selected under "Use axis parameters of", the following fields can be edited. If a configured axis has been selected, the values of this axis will be displayed.

Maximum velocity / Start/stop velocity

Define the maximum permissible velocity and the start/stop velocity of the default axis in these fields. The start/stop velocity is the minimum permissible velocity of the default axis.

Limit values:

- $1.0e-12 \leq \text{start/stop velocity} \leq 1.0e12$

Start/stop velocity = 0.0

- $1.0e-12 \leq \text{maximum velocity} \leq 1.0e12$

Maximum velocity = 0.0

The value of the maximum velocity must be greater or equal to the value of the start/stop velocity.

Enable software limit switches

Activate the function of the low and high software limit switch with this checkbox. Movements in response to software limit switches being reached are not shown in the trend view.

Low / high software limit switch

Enter the position value of the low and high software limit switches in these fields.

Limits:

- $-1.0e12 \leq \text{low software limit switch} \leq -1.0e-12$
 $1.0e-12 \leq \text{low software limit switch} \leq 1.0e12$
Low software limit switch = 0.0
- $-1.0e12 \leq \text{high software limit switch} \leq -1.0e-12$
 $1.0e-12 \leq \text{high software limit switch} \leq 1.0e12$
High software limit switch = 0.0

The value of the high software limit switch must be greater than or equal to the value of the low software limit switch.

1.7 Download to CPU

When loading to the CPU S7-1200 always ensure that the project files are consistent after the online and offline loading. It is not possible to load single blocks. When selecting single blocks, all new and modified blocks are always loaded.

The following object groups can be loaded to the CPU:

Context menu command "Download to device"	Description
All	Download all new and modified blocks and a new or modified hardware configuration
Hardware configuration	Download a new or modified hardware configuration
Software	Download all new and modified blocks
Software (all blocks)	Download all blocks

The data of the Motion Control technology objects are saved in the data blocks. The conditions for downloading of "blocks" thus apply when loading a new or modified technology object.

CAUTION

Possible malfunctions of the axis when loading without hardware configuration

The hardware configuration is modified when the following modifications are made to the axis configuration:

- Modification of the pulse generator (PTO)
- Modification of the HW limit switch address
- Modification of the reference point switch address

If the modified configuration of the axis is loaded with the context menu commands "Software" or "Software (all blocks)" without downloading the hardware configuration, this can lead to malfunctions of the axis.

Ensure that the current hardware configuration is downloaded to the CPU under the conditions listed below.

Download in CPU S7-1200 RUN operating mode (from firmware version V2.2)

For CPU S7-1200 from firmware version V2.2, when loading in CPU RUN operating mode it is checked whether it is possible to load without stopping the CPU.

The following conditions apply when loading data blocks in RUN operating mode:

	Download to load memory	Download to work memory
Data block modified values	Yes	No
Data block modified structure	No	No
New data block	Yes	Yes
Data block deleted	Yes	Yes

From technology version V3.0, Motion Control technology objects (data blocks) can also be downloaded in CPU RUN operating mode.

Technology objects after V3.0 cannot be downloaded in CPU RUN operating mode.

Select one of the actions described below to download the modified version of a Motion Control technology object (from version V3.0) to the work memory:

- **Technology object axis and command table**
Change the CPU operating mode from STOP to RUN.
- **Technology object axis**
Disable the axis and execute a "Restart" using the Motion Control instruction "MC_Reset".
- **Technology object command table**
Ensure that the command table is not being used. Download the data block of the command table to the work memory using the extended instruction "READ_DBL".

Note

In contrast to downloading in STOP operating mode, no actual parameters are overwritten in RUN operating mode. Modifications to the actual parameters only take place at the next change of operating mode from STOP to RUN.

See also

Guidelines on use of motion control (Page 21)

MC_Reset: Acknowledge error (Page 166)

1.8 Commissioning the axis - Axis control panel

Use the axis command table to move the axis in manual mode, to optimize the axis settings, and to test your system.

The axis control table can only be used if an online connection to the CPU is established.

NOTICE

Response times of the axis control panel

The response time during axis control table operation depends on the communication load of the CPU. Close all other online windows of the TIA Portal to minimize the response time.

"Manual control" button

Click "Manual control" to move the axis in manual control mode. Start by disabling the axis in the user program using motion control instruction "MC_Power". In "Manual control" mode, the axis control table takes over control priority for the axis functions. The user program has no influence on the axis functions until manual control is ended.

WARNING

The Manual control is active for one axis only. A second axis could be moved in Automatic mode, but this would bring about a dangerous situation.

In this case, set the second axis out of operation.

"Automatic mode" button

Click "Automatic mode" to end the "Manual control" mode. The axis control table passes back the control priority and the axis can be controlled by the user program again. The axis must be re-enabled in the user program and homed, if required.

Complete all active traversing motions before switching to automatic control; otherwise, the axis will be braked with the emergency stop deceleration.

"Enable" button

Click "Enable" to enable the axis in "Manual control" mode. When the axis is enabled, the axis control panel functions can be used.

If the axis cannot be enabled because certain conditions are not met, note the error message in the "Error message" field. Information on eliminating errors is available in the Appendix under "List of ErrorIDs and ErrorInfos". After the error has been corrected, enable the axis again.

"Disable" button

Click "Disable" if you want to temporarily disable the axis in "Manual control" mode.

"Command" area

Operation in the "Command" area is only possible if the axis is enabled.
You can select one of the following command inputs:

- **Jogging**

This command is equivalent to motion control command "MC_MoveJog" in the user program.

- **Positioning**

This command is equivalent to the motion control jobs "MC_MoveAbsolute" and "MC_MoveRelative" in the user program. The axis must be homed for absolute positioning.

- **Homing**

This command is equivalent to motion control command "MC_Home" in the user program.

- The "Set reference point" button corresponds to Mode = 0 (direct homing absolute)
- The "Active homing" button corresponds to Mode = 3 (active homing)

For active homing, the homing switch must be configured in the axis configuration.

The values for approach velocity, homing velocity, and reference position offset are taken from the axis configuration unchanged.

Depending on the selection, the relevant fields for entry of setpoints and the buttons for starting the command are displayed.

"Axis status" area

If "Manual control" mode is activated, the current axis status and drive status are shown in the "Axis status" area. The current position and velocity of the axis are displayed at "Process values".

Click "Acknowledge" to acknowledge all cleared errors.

The "Info message" field displays advanced information about the status of the axis.

Error message

The "Error message" field shows the current error. In "Manual control" mode, the error entry can be deleted by pressing the "Acknowledge" button once the error is eliminated.

Note

Initial values for velocity, acceleration / deceleration and jerk

For safety reasons, the "Velocity", "Acceleration/deceleration" and "Jerk" parameters are initialized with values equivalent to only 10% of the configured values when the axis command table is activated. The "Jerk" parameter is only used for technology object "Axis" V2.0 and higher.

The values in the configuration view displayed when you select "Extended parameters > Dynamics > General" are used for initialization.

The "Velocity" parameter on the control panel is derived from the "Maximum velocity" and the "Acceleration/deceleration" parameters from "Acceleration" in the configuration.

The "Velocity", "Acceleration/deceleration" and "Jerk" parameters can be changed in the axis command table; this does not affect the values in the configuration.

See also

Guidelines on use of motion control (Page 21)

Working with watch tables (Page 104)

1.9 Programming

1.9.1 Overview of the Motion Control statements

You control the axis with the user program using motion control instructions. The instructions start Motion Control jobs that execute the desired functions.

The status of the motion control jobs and any errors that occur during their execution can be obtained from the output parameters of the Motion Control instructions. The following Motion Control instructions are available:

- MC_Power: Enable, disable axis (Page 161)
- MC_Reset: Acknowledge error (Page 166)
- MC_Home: Home axes, set home position (Page 168)
- MC_Halt: Halt axis (Page 172)
- MC_MoveAbsolute: Absolute positioning of axes (Page 176)
- MC_MoveRelative: Relative positioning of axes (Page 180)
- MC_MoveVelocity: Move axes at preset rotational speed (Page 184)
- MC_MoveJog: Move axes in jogging mode (Page 189)
- MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0) (Page 192)
- MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)

See also

Creating a user program (Page 80)

Programming notes (Page 83)

Behavior of the Motion Control commands after POWER OFF and restart (Page 85)

Monitoring active commands (Page 86)

Error displays of the Motion Control statements (Page 98)

1.9.2 Creating a user program

In the section below you learn how to create a user program with the basic configuration for controlling your axis. All available axis functions are controlled using the Motion Control instructions to be inserted.

Requirement

- The technology object has been created and configured without errors.

Before creating and testing the user program, it is advisable to test the function of the axis and the corresponding parts of the system with the axis command table.

Procedure

Proceed as follows to create the user program in accordance with the principles described below:

1. In the project tree, double-click your code block (the code block must be called in the cyclic program).

The code block is opened in the programming editor and all available instructions are displayed.

2. Open the "Technology" category and the "Motion Control" and "S7-1200 Motion Control" folders.
3. Use a drag-and-drop operation to move the "MC_Power" instruction to the desired network of the code block.

The dialog box for defining the instance DB opens.

4. In the next dialog box, select from the following alternatives:

Single instance

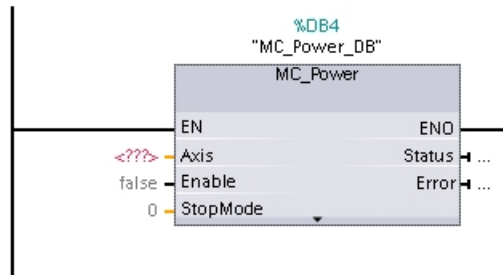
Click "Single instance" and select whether you want to define the name and number of the instance DB automatically or manually.

Multi-instance

Click "Multi-instance" and select whether you want to define the name of the multi-instance automatically or manually.

- Click "OK".

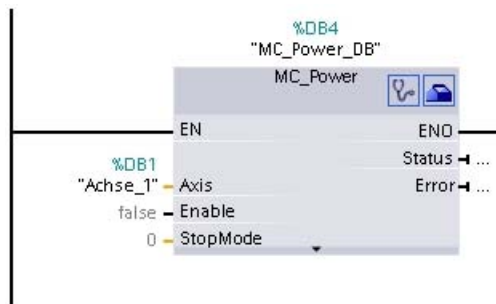
The Motion Control instruction "MC_Power" is inserted into the network.



Parameters marked with "<???'>" must be initialized; all other parameters are assigned default values.

Parameters displayed in black are required for use of the Motion Control instruction.

- Select technology object in the project tree and drag-and-drop it on <???'>.



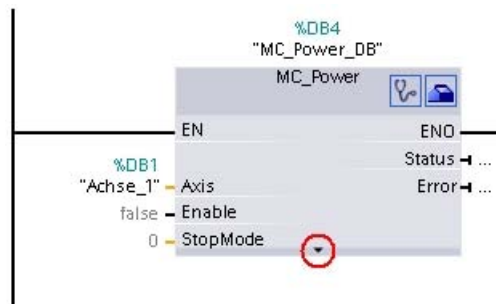
Following selection of the technology object data block, the following buttons are available:



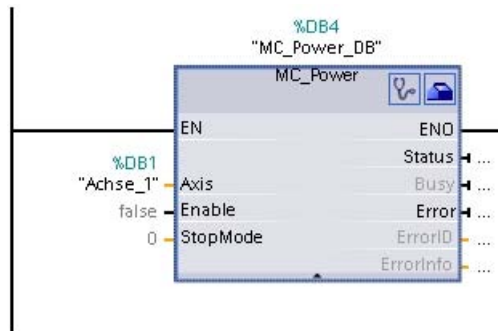
Click the stethoscope icon if you want to open the diagnostics dialog for the technology object.



Click the toolbox icon if you want to open the configuration view of the technology object.



Click the arrow down icon to view additional parameters of the Motion Control instruction.



The grayed-out parameters now visible can be used optionally.

7. Add your choice of Motion Control instructions in accordance with steps 3 to 6.

Result

You have created the basic configuration for axis control in the user program.

Initialize the input parameters of Motion Control instructions in other parts of the user program to initiate the desired jobs for the "Axis" technology object.

Evaluate the output parameters of the Motion Control instructions and the tags of the data block to track the initiated jobs and the status of the axis.

Refer to the detailed description for details on the parameters of Motion Control instructions.

See also

Overview of the Motion Control statements (Page 79)

Programming notes (Page 83)

Behavior of the Motion Control commands after POWER OFF and restart (Page 85)

Monitoring active commands (Page 86)

Error displays of the Motion Control statements (Page 98)

1.9.3 Programming notes

When creating your user program, note the following information:

- **Cyclic call of utilized motion control instructions**

The current status of command execution is available via the output parameters of the motion control instruction. The status is updated with every call of the motion control instruction. Therefore, make sure that the utilized motion control instructions are called cyclically.

- **Transfer of parameter values of a motion control instruction**

The parameter values pending for the input parameters are transferred with a positive edge at input parameter "Execute" when the block is called.

The motion control command is started with these parameter values. Parameter values that are subsequently changed for the motion control instruction are not transferred until the next start of the motion control command.

Exceptions to this are input parameters "StopMode" of motion control instruction "MC_Power" and "Velocity" of motion control instruction "MC_MoveJog". A change in the input parameter is also applied when "Enable" = TRUE, or "JogForward" and "JogBackward".

- **Programming under consideration of the status information**

In a stepwise execution of motion control jobs, make sure to wait for the active command to finish before starting a new command. Use the status messages of the motion control instruction and the "StatusBits" tag of the technology object to check for completion of the active command.

In the examples below, observe the indicated sequence. Failure to observe the sequence will display an axis or command error.

- **Axis enable with motion control instruction "MC_Power"**

You must enable the axis before it can take on motion jobs. Use an AND operation of tag <Axis name>.StatusBits.Enable = TRUE with output parameter Status = TRUE of motion control instruction "MC_Power" to verify that the axis is enabled.

- **Acknowledge error with motion control instruction "MC_Reset"**

Prior to starting a motion control command, errors requiring acknowledgement must be acknowledged with "MC_Reset". Eliminate the cause of the error and acknowledge the error with motion control instruction "MC_Reset". Verify that the error has been successfully acknowledged before initiating a new command. For this purpose, use an AND operation of tag <Axis name>.StatusBits.Error = FALSE with output parameter Done = TRUE of motion control instruction "MC_Reset".

- **Home axis with motion control instruction "MC_Home"**

Before you can start an MC_MoveAbsolute command, the axis must be homed. Use an AND operation of tag <Axis name>.StatusBits.HomingDone = TRUE with output parameter Done = TRUE of motion control instruction "MC_Home" to verify that the axis has been homed.

- **Override of motion control command processing**

Motion control jobs for moving an axis can also be executed as overriding jobs.

If a new motion control command is started for an axis while another motion control command is active, the active command is overridden by the new command before the existing command is completely executed. The overridden command signals this using `CommandAborted = TRUE` in the motion control instruction. It is possible to override an active `MC_MoveRelative` command with a `MC_MoveAbsolute` command.

- **Avoiding multiple use of the same instance**

All relevant information of a motion control command is stored in its instance.

Do not start a new command using this instance, if you want to track the status of the current command. Use different instances if you want to track the commands separately. If the same instance is used for multiple motion control commands, the status and error information of the individual commands will overwrite each other.

- **Call of motion control instructions in different priority classes (run levels)**

Motion Control instructions with the same instance may not be called in different priority classes without interlocking. To learn how to call locked motion control instructions, refer to "Tracking commands from higher priority classes (run levels) (Page 109)".

See also

Overview of the Motion Control statements (Page 79)

Creating a user program (Page 80)

Behavior of the Motion Control commands after POWER OFF and restart (Page 85)

Monitoring active commands (Page 86)

Error displays of the Motion Control statements (Page 98)

Tracking jobs from higher priority classes (execution levels) (Page 109)

1.9.4 Behavior of the Motion Control commands after POWER OFF and restart

A POWER OFF or CPU-STOP aborts all active motion control jobs. All CPU outputs, including pulse and direction outputs, are reset.

After a subsequent POWER ON or CPU restart (CPU RUN), the technology objects and the motion control jobs will be reinitialized.

All actual data of the technology objects as well as all status and error information of the previously active motion control jobs are reset to their initial values.

Before the axis can be reused, it must be enabled again using the Motion Control instruction "MC_Power". If homing is required, the axis must be homed again with Motion Control instruction "MC_Home".

See also

Overview of the Motion Control statements (Page 79)

Creating a user program (Page 80)

Programming notes (Page 83)

Monitoring active commands (Page 86)

Error displays of the Motion Control statements (Page 98)

1.9.5 Monitoring active commands

1.9.5.1 Monitoring active commands

There are three typical groups for tracking active motion control jobs:

- Motion control instructions with output parameter "Done"
- Motion control instruction "MC_MoveVelocity"
- Motion control instruction "MC_MoveJog"

1.9.5.2 Motion control instructions with "Done" output parameter

Motion control instructions with the output parameter "Done" are started via input parameter "Execute" and have a defined conclusion (for example, with Motion Control instruction "MC_Home": Homing was successful). The job is complete and the axis is at a standstill.

The jobs of the following Motion Control instructions have a defined conclusion:

- MC_Reset
- MC_Home
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_CommandTable (as of technology object V2.0)
- MC_ChangeDynamic (as of technology object V2.0)

The output parameter "Done" indicates the value TRUE, if the job has been successfully completed.

The output parameters "Busy", "CommandAborted", and "Error" signal that the job is still being processed, has been aborted or an error is pending. The Motion Control instruction "MC_Reset" cannot be aborted and thus has no "CommandAborted" output parameter. The Motion Control instruction "MC_ChangeDynamic" is completed immediately and therefore has no "Busy" or "CommandAborted" output parameters.

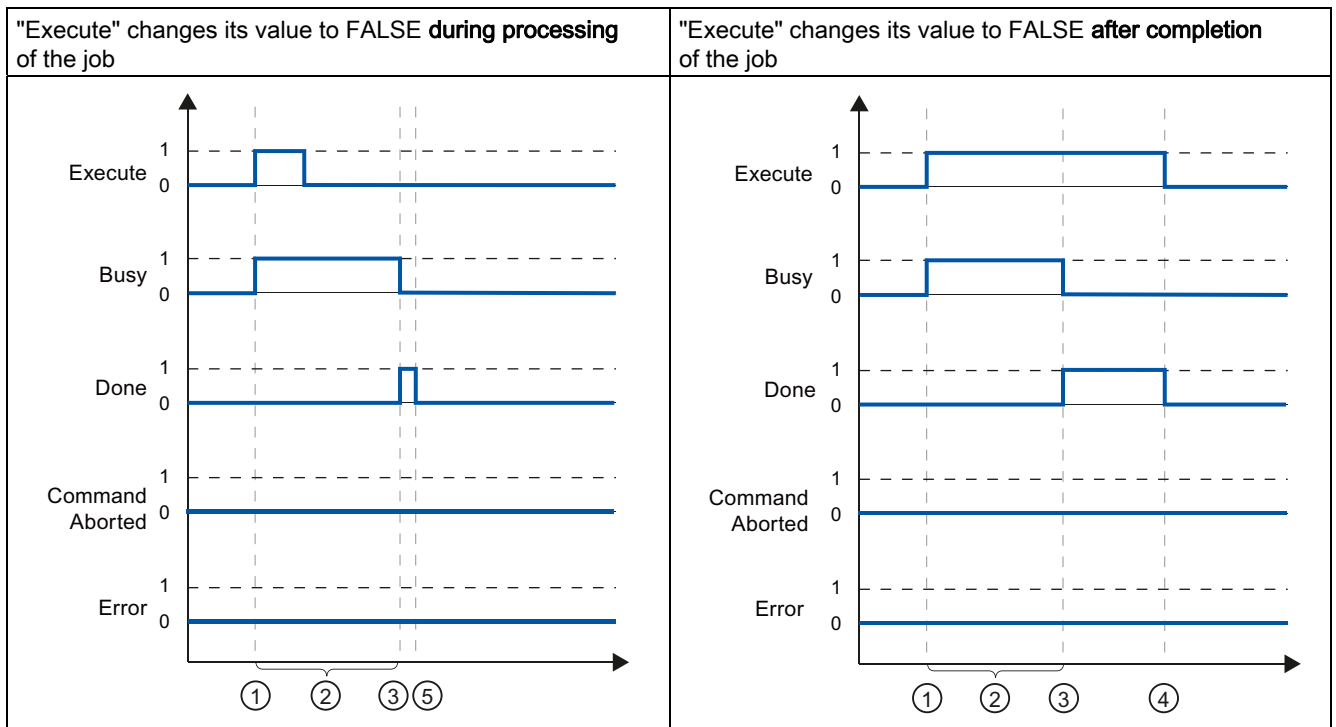
During processing of the motion control job, the output parameter "Busy" indicates the value TRUE. If the job has been completed, aborted, or stopped by an error, the output parameter "Busy" changes its value to FALSE. This change occurs regardless of the signal at input parameter "Execute".

Output parameters "Done", "CommandAborted", and "Error" indicate the value TRUE for at least one cycle. These status messages are latched while input parameter "Execute" is set to TRUE.

The behavior of the status bits is presented below for various example situations:

Complete execution of job

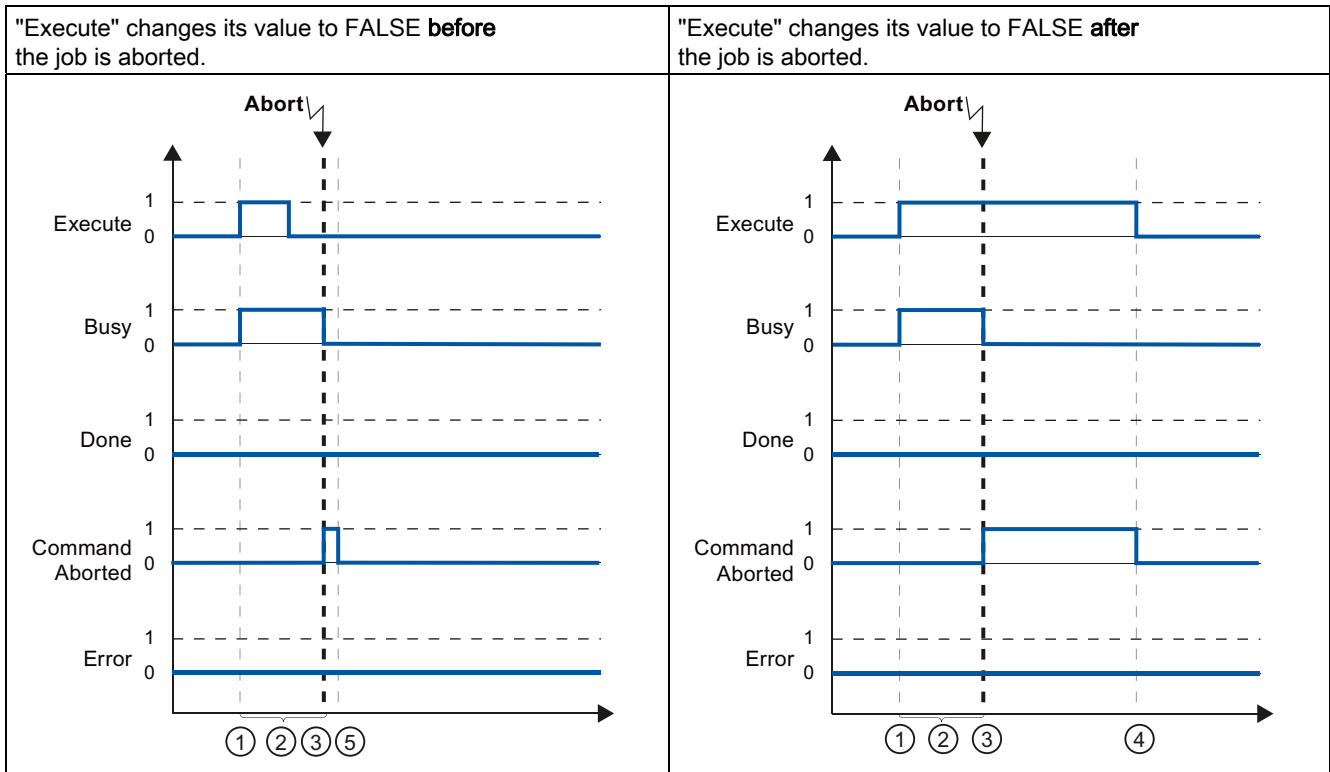
If the motion control job has been completely executed by the time of its conclusion, this is indicated by the value TRUE in output parameter "Done". The signal status of input parameter "Execute" influences the display duration in the output parameter "Done":



①	The job is started with a positive edge at the input parameter "Execute". Depending on the programming, "Execute" can still be reset to the value FALSE during the job, or the value TRUE can be retained until after completion of the job.
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	With conclusion of the job (for example, for Motion Control instruction "MC_Home": Homing was successful), output parameter "Busy" changes to FALSE and "Done" to TRUE.
④	If "Execute" retains the value TRUE until after completion of the job, then "Done" also remains TRUE and changes its value to FALSE together with "Execute".
⑤	If "Execute" has been set to FALSE before the job is complete, "Done" indicates the value TRUE for only one execution cycle.

Abort job

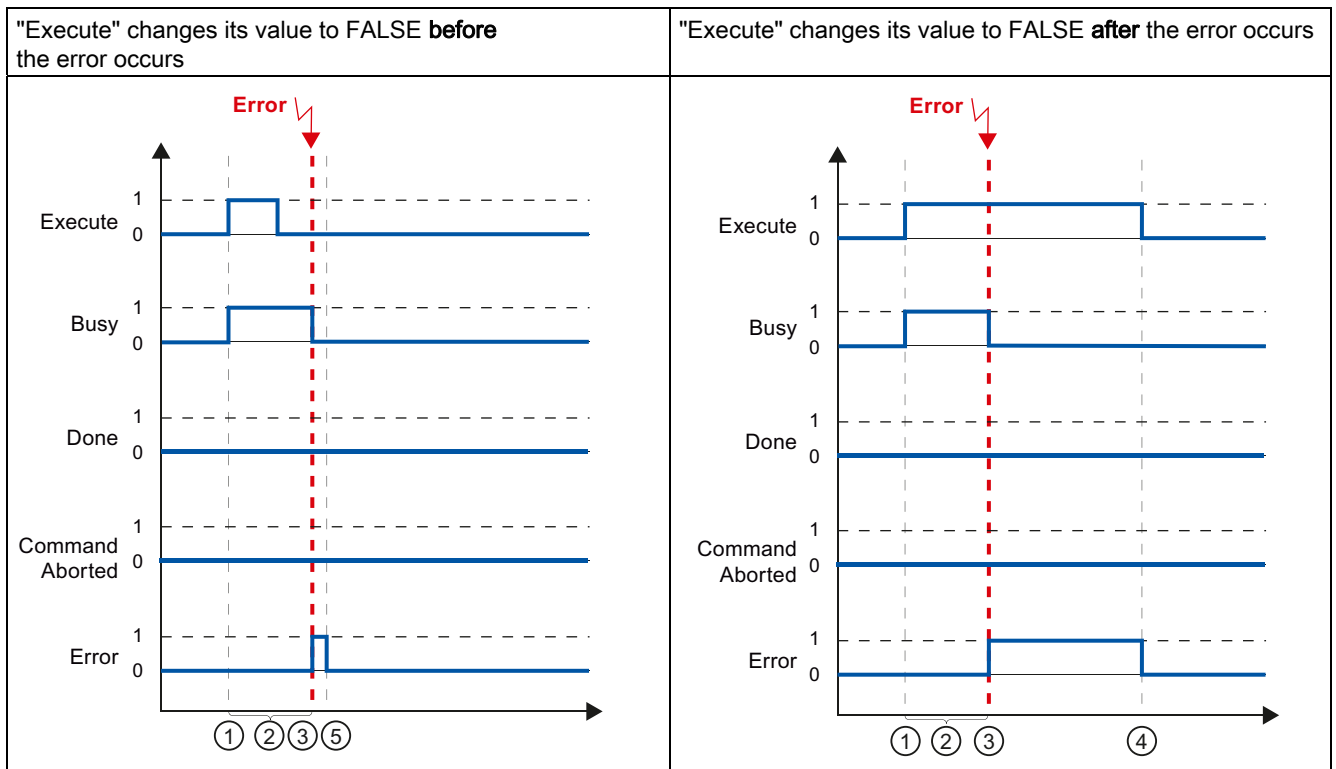
If the motion control job is aborted during execution, this is indicated by the value TRUE in output parameter "CommandAborted". The signal status of the input parameter "Execute" influences the display duration in the output parameter "CommandAborted":



①	The job is started with a positive edge at the input parameter "Execute". Depending on the programming, "Execute" can still be reset to the value FALSE during the job, or the value TRUE can be retained until after completion of the job.
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	During job execution, the job is aborted by another motion control job. If the job is aborted, output parameter "Busy" changes to FALSE and "CommandAborted" to TRUE.
④	If "Execute" retains the value TRUE until after the job is aborted, then "CommandAborted" also remains TRUE and changes its value to FALSE together with "Execute".
⑤	If "Execute" has been set to FALSE before the job is aborted, "CommandAborted" indicates the value TRUE for only one execution cycle.

Error during job execution

If an error occurs during execution of the motion control job, this is indicated by the value TRUE in the output parameter "Error". The signal status of the input parameter "Execute" influences the display duration in the output parameter "Error":



①	The job is started with a positive edge at the input parameter "Execute". Depending on the programming, "Execute" can still be reset to the value FALSE during the job, or the value TRUE can be retained until after completion of the job.
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	An error occurred during job execution. When the error occurs, the output parameter "Busy" changes to FALSE and "Error" to TRUE.
④	If "Execute" retains the value TRUE until after the error occurs, then "Error" also remains TRUE and only changes its value to FALSE together with "Execute".
⑤	If "Execute" has been set to FALSE before the error occurs, "Error" indicates the value TRUE for only one execution cycle.

1.9.5.3 Motion control instruction MC_MoveVelocity

The jobs of Motion Control instruction "MC_MoveVelocity" do not have a defined end. The job objective is fulfilled when the parameterized velocity is reached for the first time and the axis travels at constant velocity. When the parameterized velocity is reached, this is indicated by the value TRUE in output parameter "InVelocity".

The job is complete when the parameterized velocity has been reached and input parameter "Execute" has been set to the value FALSE. However, the axis motion is not yet complete upon completion of the job. For example, the axis motion can be stopped with motion control job "MC_Halt".

The output parameters "Busy", "CommandAborted", and "Error" signal that the job is still being processed, has been aborted or an error is pending.

During execution of the motion control job, output parameter "Busy" indicates the value TRUE. If the job has been completed, aborted, or stopped by an error, the output parameter "Busy" changes its value to FALSE. This change occurs regardless of the signal at input parameter "Execute".

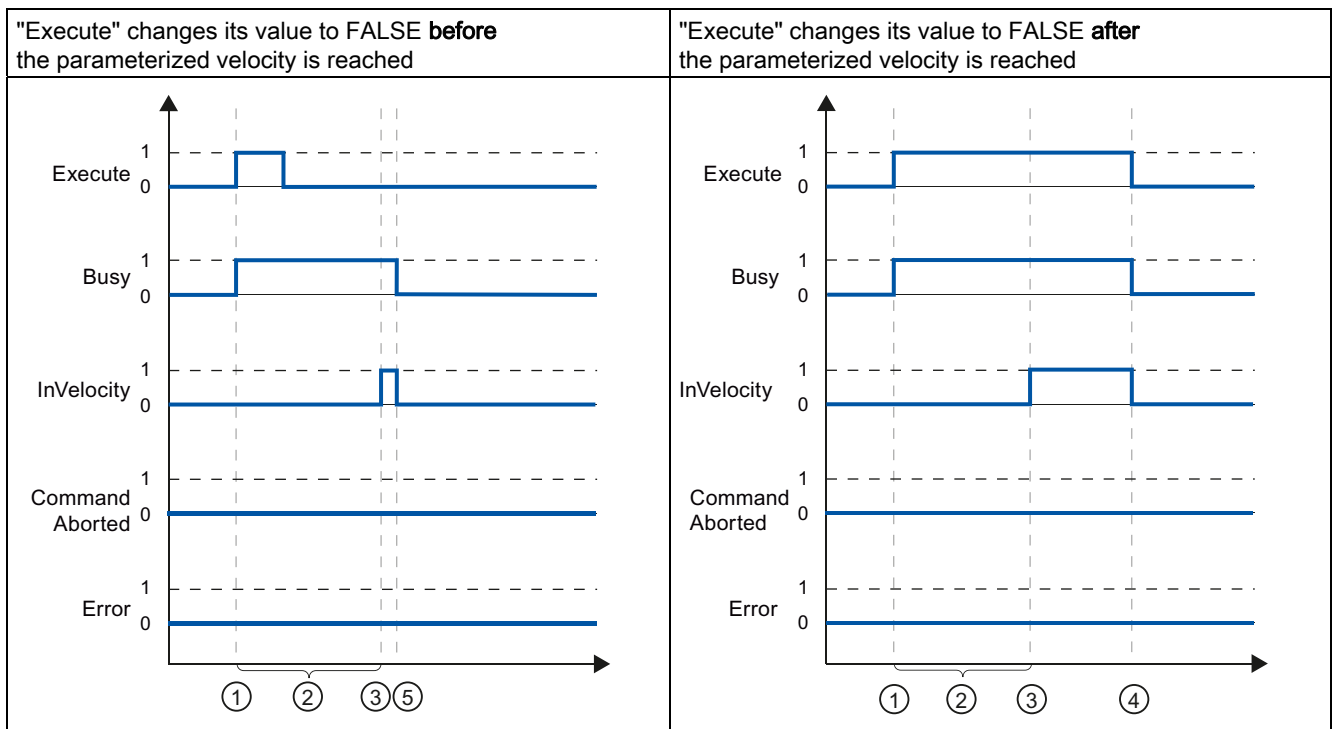
The output parameters "InVelocity", "CommandAborted", and "Error" indicate the value TRUE for at least one cycle, when their conditions are met. These status messages are latched while input parameter "Execute" is set to TRUE.

The behavior of the status bits is presented below for various example situations:

The parameterized velocity is reached

If the motion control job has been executed by the time the parameterized velocity is reached, this is indicated by the value TRUE in output parameter "InVelocity".

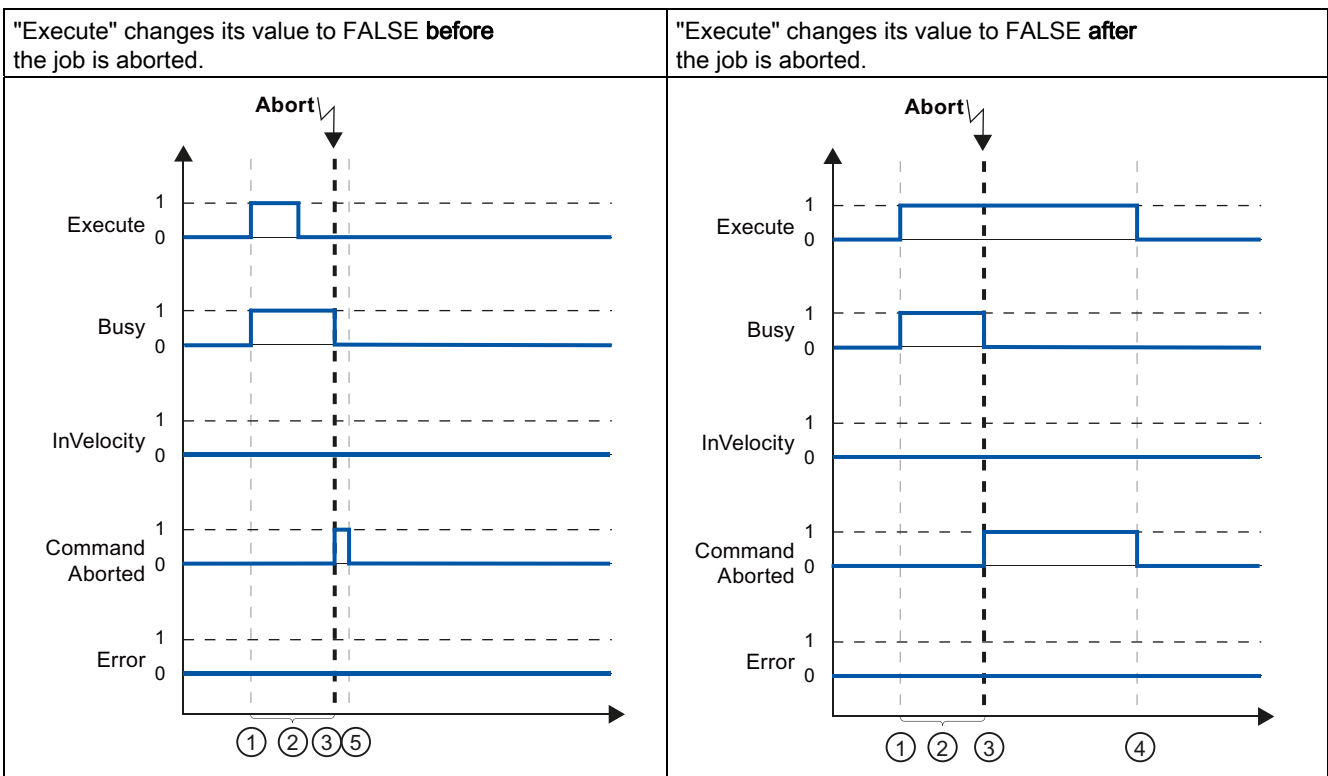
The signal status of the input parameter "Execute" influences the display duration in the output parameter "InVelocity":



①	The job is started with a positive edge at the input parameter "Execute". Depending on the programming, "Execute" can be reset to the value FALSE event before the parameterized velocity is reached, or alternatively only after it has been reached.
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	When the parameterized velocity is reached, the output parameter "InVelocity" changes to TRUE.
④	If "Execute" retains the value TRUE even after the parameterized velocity has been reached, the job remains active. "InVelocity" and "Busy" retain the value TRUE and only change their status to FALSE together with "Execute".
⑤	If "Execute" has been reset to FALSE before the parameterized velocity is reached, the job is complete when the parameterized velocity is reached. "InVelocity" indicates the value TRUE for one execution cycle and changes to FALSE together with "Busy".

The job is aborted prior to reaching the parameterized velocity

If the motion control job is aborted before the parameterized velocity is reached, this is indicated by the value TRUE in output parameter "CommandAborted". The signal status of input parameter "Execute" influences the display duration in output parameter "CommandAborted".



- | | |
|---|---|
| ① | The job is started with a positive edge at the input parameter "Execute". Depending on the programming, "Execute" can still be reset to the value FALSE during the job, or the value TRUE can be retained until after the job is aborted. |
| ② | While the job is active, the output parameter "Busy" indicates the value TRUE. |
| ③ | During job execution, the job is aborted by another motion control job. If the job is aborted, output parameter "Busy" changes to FALSE and "CommandAborted" to TRUE. |
| ④ | If "Execute" retains the value TRUE until after the job is aborted, then "CommandAborted" also remains TRUE and changes its status to FALSE together with "Execute". |
| ⑤ | If "Execute" has been reset to FALSE before the job is aborted, "CommandAborted" indicates the value TRUE for only one execution cycle. |

Note

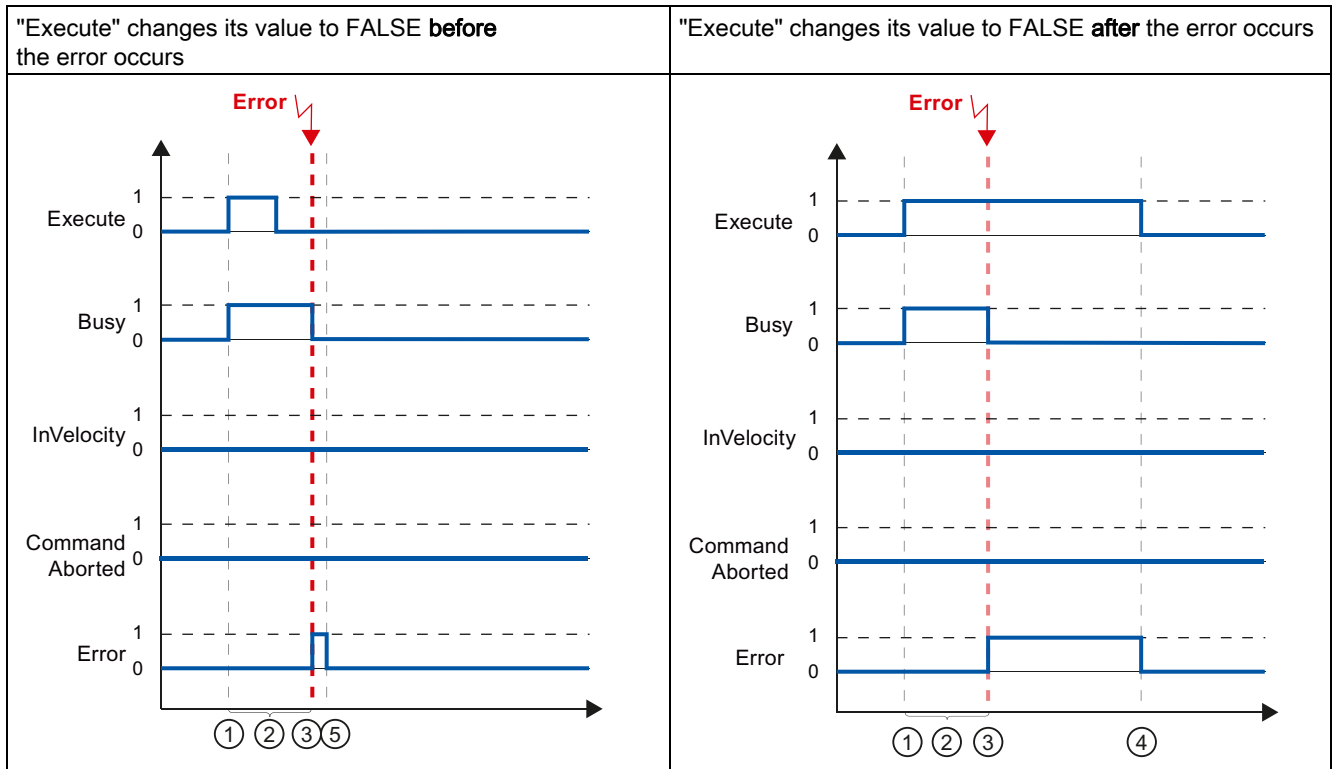
Under the following conditions, an abort is not indicated in output parameter "CommandAborted":

The parameterized velocity has been reached, input parameter "Execute" has the value FALSE, and a new motion control job is initiated.

When the parameterized velocity is reached and input parameter "Execute" has the value FALSE, the job is complete. Therefore, the start of a new job is not indicated as an abort.

An error has occurred prior to reaching the parameterized velocity

If an error occurs during execution of the motion control job before the parameterized velocity has been reached, this is indicated by the value TRUE in the output parameter "Error". The signal status of the input parameter "Execute" influences the display duration in the output parameter "Error":



①	The job is started with a positive edge at the input parameter "Execute". Depending on the programming, "Execute" can still be reset to the value FALSE during the job, or the value TRUE can be retained until after the error has occurred.
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	An error occurred during job execution. When the error occurs, the output parameter "Busy" changes to FALSE and "Error" to TRUE.
④	If "Execute" retains the value TRUE until after the error has occurred, then "Error" also remains TRUE and only changes its status to FALSE together with "Execute".
⑤	If "Execute" has been reset to FALSE before the error occurs, "Error" indicates the value TRUE for only one execution cycle.

Note

Under the following conditions, an error is not indicated in output parameter "Error":

The parameterized velocity has been reached, input parameter "Execute" has the value FALSE, and an axis error occurs (software limit switch is approached, for example).

When the parameterized velocity is reached and input parameter "Execute" has the value FALSE, the job is complete. After completion of the job, the axis error is only indicated in the Motion Control instruction "MC_Power".

1.9.5.4 Motion control instruction MC_MoveJog

The jobs of Motion Control instruction "MC_MoveJog" implement a jog operation.

The motion control jobs "MC_MoveJog" do not have a defined end. The job objective is fulfilled when the parameterized velocity is reached for the first time and the axis travels at constant velocity. When the parameterized velocity is reached, this is indicated by the value TRUE in output parameter "InVelocity".

The order is complete when input parameter "JogForward" or "JogBackward" has been set to the value FALSE and the axis has come to a standstill.

The output parameters "Busy", "CommandAborted", and "Error" signal that the job is still being processed, has been aborted or an error is pending.

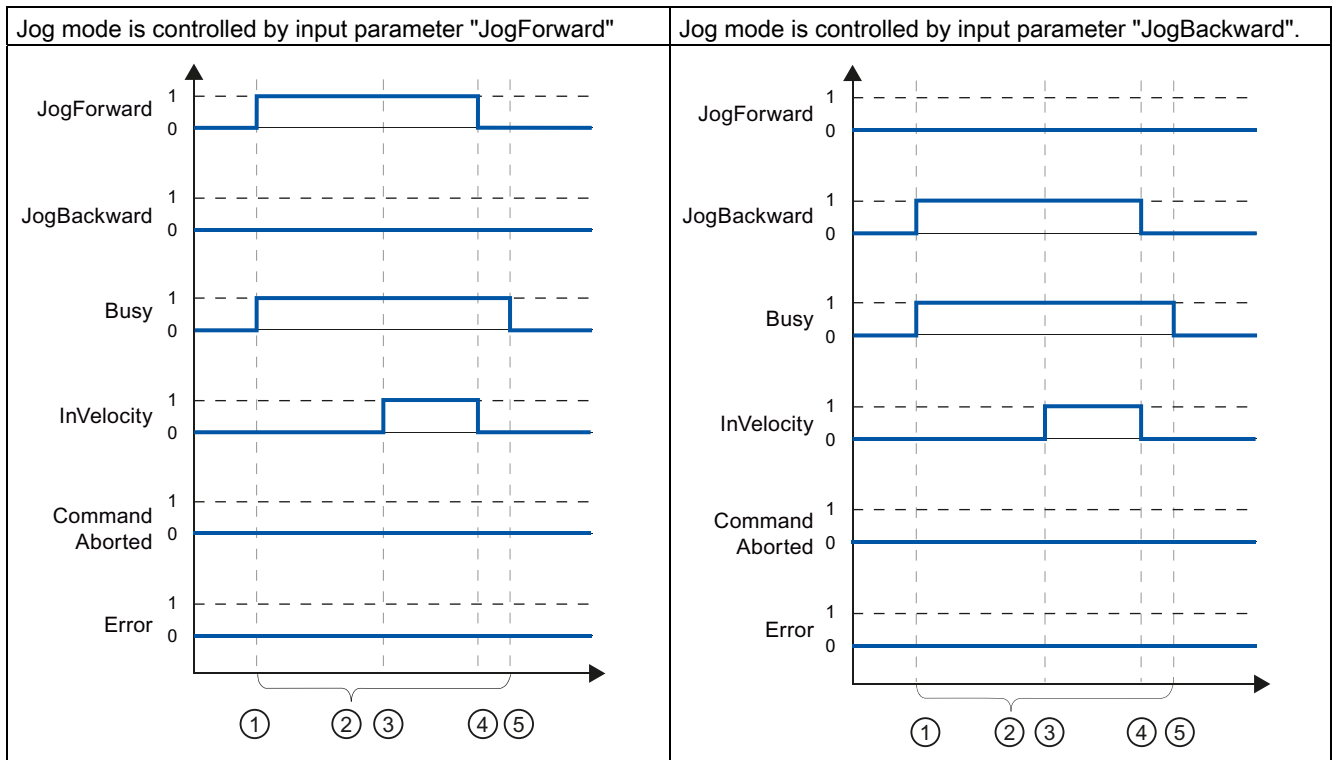
During processing of the motion control job, the output parameter "Busy" indicates the value TRUE. If the job has been completed, aborted, or stopped by an error, the output parameter "Busy" changes its value to FALSE.

The output parameter "InVelocity" indicates the status TRUE, as long as the axis is moving at the parameterized velocity. The output parameters "CommandAborted" and "Error" indicate the status for at least one cycle. These status messages are latched as long as either input parameter "JogForward" or "JogBackward" is set to TRUE.

The behavior of the status bits is presented below for various example situations:

The parameterized velocity is reached and maintained

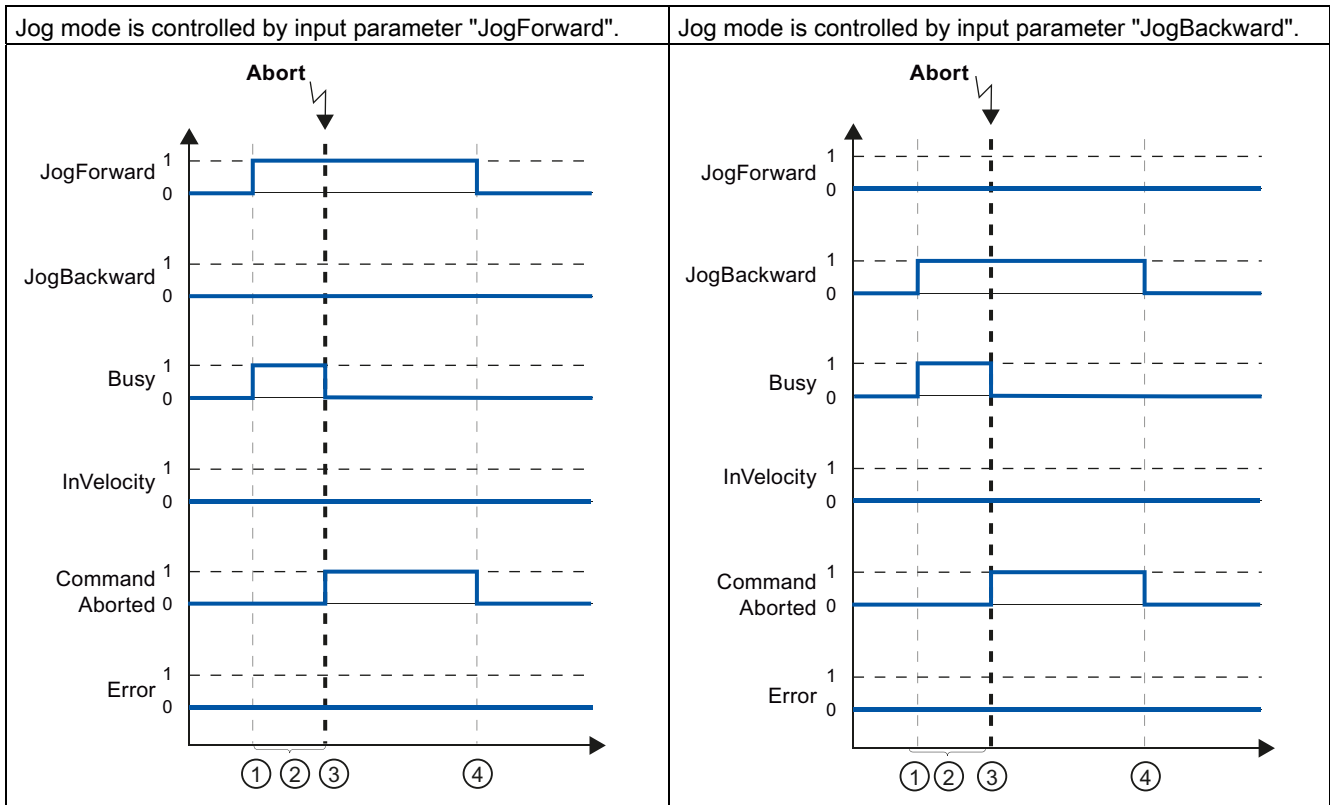
If the motion control job has been executed by the time the parameterized velocity is reached, this is indicated by the value TRUE in output parameter "InVelocity".



①	The job is started with a positive edge at the input parameter "JogForward" or "JogBackward".
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	When the parameterized velocity is reached, the output parameter "InVelocity" changes to TRUE.
④	When the input parameter "JogForward" or "JogBackward" is reset to the value FALSE, the axis motion ends. The axis starts to decelerate. As a result, the axis no longer moves at constant velocity and the output parameter "InVelocity" changes its status to FALSE.
⑤	If the axis has come to a standstill, the motion control job is complete and the output parameter "Busy" changes its value to FALSE.

The job is aborted during execution

If the motion control job is aborted during execution, this is indicated by the value TRUE in output parameter "CommandAborted". The behavior is independent of whether or not the parameterized velocity has been reached.



①	The job is started with a positive edge at the input parameter "JogForward" or "JogBackward".
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	During job execution, the job is aborted by another motion control job. If the job is aborted, output parameter "Busy" changes to FALSE and "CommandAborted" to TRUE.
④	When the input parameter "JogForward" or "JogBackward" is reset to the value FALSE, the output parameter "CommandAborted" changes its value to FALSE.

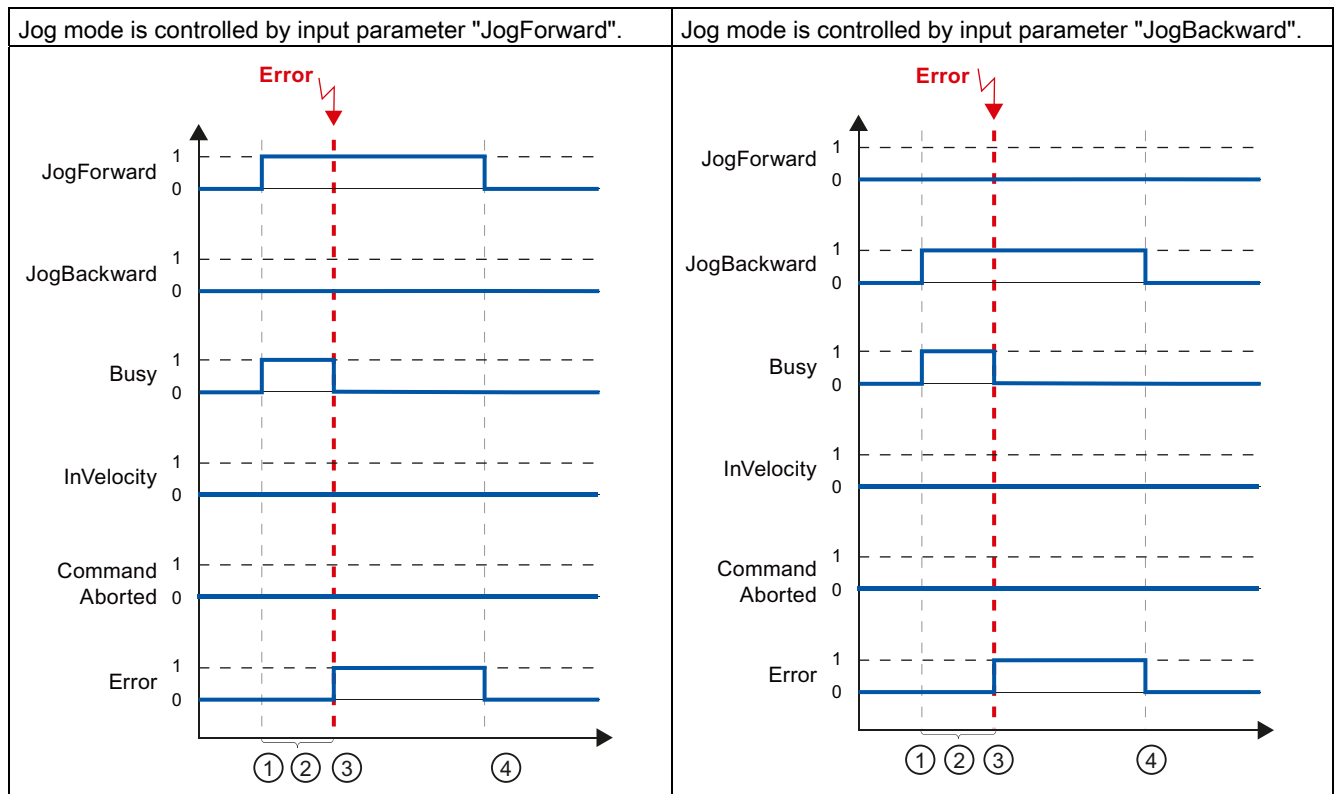
Note

The job abort is indicated in the output parameter "CommandAborted" for only one execution cycle, if all conditions below are met:

The input parameters "JogForward" and "JogBackward" have the value FALSE (but the axis is still decelerating) and a new motion control job is initiated.

An error has occurred during job execution

If an error occurs during execution of the motion control job, this is indicated by the value TRUE in output parameter "Error". The behavior is independent of whether or not the parameterized velocity has been reached.



①	The job is started with a positive edge at the input parameter "JogForward" or "JogBackward".
②	While the job is active, the output parameter "Busy" indicates the value TRUE.
③	An error occurred during job execution. When the error occurs, the output parameter "Busy" changes to FALSE and "Error" to TRUE.
④	When the input parameter "JogForward" or "JogBackward" is reset to the value FALSE, the output parameter "Error" changes its value to FALSE.

Note

An error occurrence is indicated in the output parameter "Error" for only one execution cycle, if all the conditions below are met:

The input parameters "JogForward" and "JogBackward" have the value FALSE (but the axis is still decelerating) and a new error occurs (software limit switch is approached, for example).

1.9.6 Error displays of the Motion Control statements

The Motion Control instructions indicate any motion control job and technology object errors in the output parameters "Error", "ErrorID" and "ErrorInfo" of the Motion Control instructions.

Error display at output parameters "Error", "ErrorID" and "ErrorInfo"

If the output parameter "Error" indicates the value TRUE, the complete job, or portions thereof, could not be executed. The cause of the error is indicated by the value in output parameter "ErrorID". Detailed information about the cause of the error is returned by the value in output parameter "ErrorInfo". We distinguish between the following error classes for error indication:

- **Operating error with axis stop (for example, "HW limit switch was approached")**

Operating errors with axis stop are errors that occur during runtime of the user program. If the axis is in motion, it is stopped with the configured deceleration or emergency stop deceleration, depending on the error. The errors are indicated in the error-triggering Motion Control instruction and in the Motion Control instruction "MC_Power".

- **Operating error without axis stop (for example, "Axis is not homed")**

Operating errors without axis stop are errors that occur during runtime of the user program. If the axis is in motion, the motion is continued. The errors are only indicated in the Motion Control instruction which triggers the error.

- **Parameterization error of Motion Control instruction " (for example, "Incorrect value in parameter "Velocity"")**

Parameterization errors occur when incorrect information is specified in the input parameters of Motion Control instructions. If the axis is in motion, the motion is continued. The errors are only indicated in the Motion Control instruction which triggers the error.

- **Configuration error on technology object "Axis" (for example, "Value for "Acceleration" is invalid")**

A configuration error exists if one or more parameters are incorrectly configured in the axis configuration or if editable configuration data have been modified incorrectly during runtime of the program. An axis in motion is stopped with the configured emergency stop deceleration. The error is indicated in the error-triggering Motion Control instruction and in Motion Control instruction "MC_Power".

- **Configuration error on technology object "Command table" (for example "Value for "Velocity" is invalid")**

There is a configuration error if one or more parameters are incorrectly set in the axis command table or if programmable configuration data have been modified incorrectly during runtime of the program. If the axis is in motion, the motion is continued. The errors are only indicated in the "MC_CommandTable" Motion Control instruction.

- **Internal error**

When an internal error occurs, the axis is stopped. The errors are indicated in the error-triggering Motion Control instruction and, in some cases, in the Motion Control instruction "MC_Power".

A detailed description of the ErrorIDs and ErrorInfos, as well as their remedies, is available in the Appendix.

See also

Overview of the Motion Control statements (Page 79)

Creating a user program (Page 80)

Programming notes (Page 83)

Behavior of the Motion Control commands after POWER OFF and restart (Page 85)

Monitoring active commands (Page 86)

1.10 Axis - Diagnostics

1.10.1 Status and error bits

You use the "Status and error bits" diagnostic function to monitor the most important status and error messages for the axis in the TIA Portal. The diagnostic function display is available in online mode in "Manual control" mode and in "Automatic control" when the axis is active. The status error messages have the following meaning:

Status of the axis

Status	Description
Enabled	The axis is enabled and ready to be controlled via motion control jobs. (Tag of technology object: <Axis name>.StatusBits.Enable)
Homed	The axis is homed and is capable of executing absolute positioning jobs of Motion Control instruction "MC_MoveAbsolute". The axis does not have to be homed for relative homing. Special situations: <ul style="list-style-type: none"> • During active homing, the status is FALSE. • If a homed axis undergoes passive homing, the status is set to TRUE during passive homing. (Tag of technology object: <Axis name>.StatusBits.HomingDone)
Axis error	An error has occurred in the "Axis" technology object. More information about the error is available in automatic control at the ErrorID and ErrorInfo parameters of the Motion Control instructions. In manual mode, the "Error message" field of the axis command table displays detailed information about the cause of error. (Tag of technology object: <Axis name>.StatusBits.Error)
Axis command table enabled	The "Manual control" mode was enabled in the axis command table. The axis command table has control priority over the "Axis" technology object. The axis cannot be controlled from the user program. (Tag of technology object: <Axis name>.StatusBits.ControlPanelActive)
Restart necessary	A modified configuration of the axis was downloaded to the load memory in CPU RUN operating mode. To download the modified configuration to the work memory, you need to restart the axis. Use the Motion Control instruction MC_Reset to do this.

Drive status

Status	Description
Drive ready	The drive is ready for operation. (Tag of technology object: <Axis name>.StatusBits.DriveReady)
Drive error	The drive has reported an error after failure of its "Drive ready" signal. (Tag of technology object: <Axis name>.ErrorBits.DriveFault)

Status of the axis motion

Status	Description
Standstill	The axis is at a standstill. (Tag of technology object: <Axis name>.StatusBits.StandStill)
Accelerating	The axis accelerates. (Tag of technology object: <Axis name>.StatusBits.Acceleration)
Constant velocity	The axis travels at constant velocity. (Tag of technology object: <Axis name>.StatusBits.ConstantVelocity)
Decelerating	The axis decelerates (slows down). (Tag of technology object: <Axis name>.StatusBits.Deceleration)

Status of the motion mode

Status	Description
Positioning	The axis executes a positioning job of the Motion Control instruction "MC_MoveAbsolute" or "MC_MoveRelative" or of the axis command table. (Tag of technology object: <Axis name>.StatusBits.PositioningCommand)
Travel with velocity specification	The axis executes a job with the velocity specification of the Motion Control instruction "MC_MoveVelocity" or "MC_MoveJog" or of the axis command table. (Tag of technology object: <Axis name>.StatusBits.SpeedCommand)
Homing	The axis executes a homing job of the Motion Control instruction "MC_Home" or the axis command table. (Tag of technology object: <Axis name>.StatusBits.Homing)
Command table active (as of technology object Axis V2.0)	The axis is controlled by Motion Control instruction "MC_CommandTable". (Tag of technology object: <Axis name>.StatusBits.CommandTableActive)

Error messages

Error	Description
Low software limit switch has been reached	The low software limit switch has been reached. (Tag of technology object: <Axis name>.ErrorBits.SwLimitMinReached)
Low software limit switch has been exceeded	The low software limit switch has been exceeded. (Tag of technology object: <Axis name>.ErrorBits.SwLimitMinExceeded)
High software limit switch has been reached	The high software limit switch has been reached. (Tag of technology object: <Axis name>.ErrorBits.SwLimitMaxReached)
High software limit switch has been exceeded	The high software limit switch has been exceeded. (Tag of technology object: <Axis name>.ErrorBits.SwLimitMaxExceeded)
Low hardware limit switch was approached	The low hardware limit switch has been approached. (Tag of technology object: <Axis name>.ErrorBits.HwLimitMin)
High hardware limit switch has been approached	The high hardware limit switch has been approached. (Tag of technology object: <Axis name>.ErrorBits.HwLimitMax)
PTO and HSC already in use	A second axis is using the same PTO (Pulse Train Output) and HSC (High Speed Counter) and is enabled with "MC_Power". (Tag of technology object: <Axis name>.ErrorBits.HwUsed)
Configuration error	The "Axis" technology object was incorrectly configured or editable configuration data were modified incorrectly during runtime of the user program. (Tag of technology object: <Axis name>.ErrorBits.ConfigFault)
Internal error	An internal error has occurred. (Tag of technology object: <Axis name>.ErrorBits.SystemFault)

See also

Motion status (Page 103)

StatusBits. tag (Page 143)

ErrorBits. tag (Page 146)

1.10.2 Motion status

Use the "Motion status" diagnostic function to monitor the motion status of the axis in the TIA Portal. The diagnostic function display is available in online mode in "Manual control" mode and in "Automatic control" when the axis is active. The displayed status information has the following meaning:

Status	Description
Position	The "Position" field indicates the current axis position. If the axis is not homed, the value indicates the position value relative to the enable position of the axis. (Tag of technology object: <Axis name>.MotionStatus.Position)
Velocity	The "Velocity" field indicates the current axis velocity. (Tag of technology object: <Axis name>.MotionStatus.Velocity)
Target position	The "Target position" field indicates the current target position of an active positioning job or of the axis command table. The value of the "Target position" is only valid during execution of a positioning job. (Tag of technology object: <Axis name>.MotionStatus.TargetPosition)
Remaining traversing distance	The "Remaining traversing distance" field indicates the traversing distance currently remaining for an active positioning job or the axis command table. The "Remaining traversing distance" value is only valid during execution of a positioning job. (Tag of technology object: <Axis name>.MotionStatus.Distance)

See also

Status and error bits (Page 100)

MotionStatus. tag (Page 142)

1.10.3 Dynamics settings

Use the "Dynamics settings" diagnostic function to monitor the dynamic limit values of the axis in the TIA Portal. The diagnostic function display is available in online mode in "Manual control" mode and in "Automatic control" when the axis is active. The status information displayed has the following meaning:

Dynamic limit	Description
Acceleration	The "Acceleration" field indicates the currently configured acceleration of the axis. (Technology object variable: <Axis name>.Config.DynamicDefaults.Acceleration)
Deceleration	The "Deceleration" field indicates the currently configured deceleration of the axis. (Technology object variable: <Axis name>.Config.DynamicDefaults.Deceleration)
Emergency stop deceleration	The "Emergency stop deceleration" field indicates the currently configured emergency stop deceleration of the axis. (Technology object variable: <Axis name>.Config.DynamicDefaults.EmergencyDeceleration)
Step (as of technology object Axis V2.0)	The "Velocity" field indicates the current axis step velocity configured. (Technology object variable: <Axis name>.Config.DynamicDefaults.Jerk)

1.11 Working with watch tables

Use watch tables if you want to monitor and modify tags of motion control instructions or the "Axis" technology object during commissioning.

To monitor and modify tags, you must specify the complete name of the tag, including object name and all structure names in a watch table.

Example: <Axis name>.Config.DynamicDefaults.Acceleration)

Tip:

You can use a copy & paste operation to avoid entering long tag names.

Procedure

To insert the tag names, follow the steps described below:

1. In the project tree, select the instance data block or the technology object of the axis.

2. **Parameters of the motion control instruction**

- Right-click and select the **Open** command in the shortcut menu.

Tags of the technology object

- Right-click and select the **Open in editor** command in the shortcut menu.

3. **Parameters of the motion control instruction**

- Select the lines of the tags in the Input or Output area

Tags of the technology object

- In the Static area, open the relevant structures and select the lines of the tags

4. Select the **Edit > Copy** menu command.

5. Double-click to open the watch table.

6. Select the line starting at which the tags are to be inserted

7. Select the **Edit > Paste** menu command.

Insert the tags with their complete names in the watch table.

WARNING

The watch table also gives you write access to tags whose use is blocked for safety reasons in the user program. Modifying these tags can result in damage to the current axis configuration and to undefined responses of the axis. Only modify those tags whose access is marked with "RW" in the tag list of the technology object.

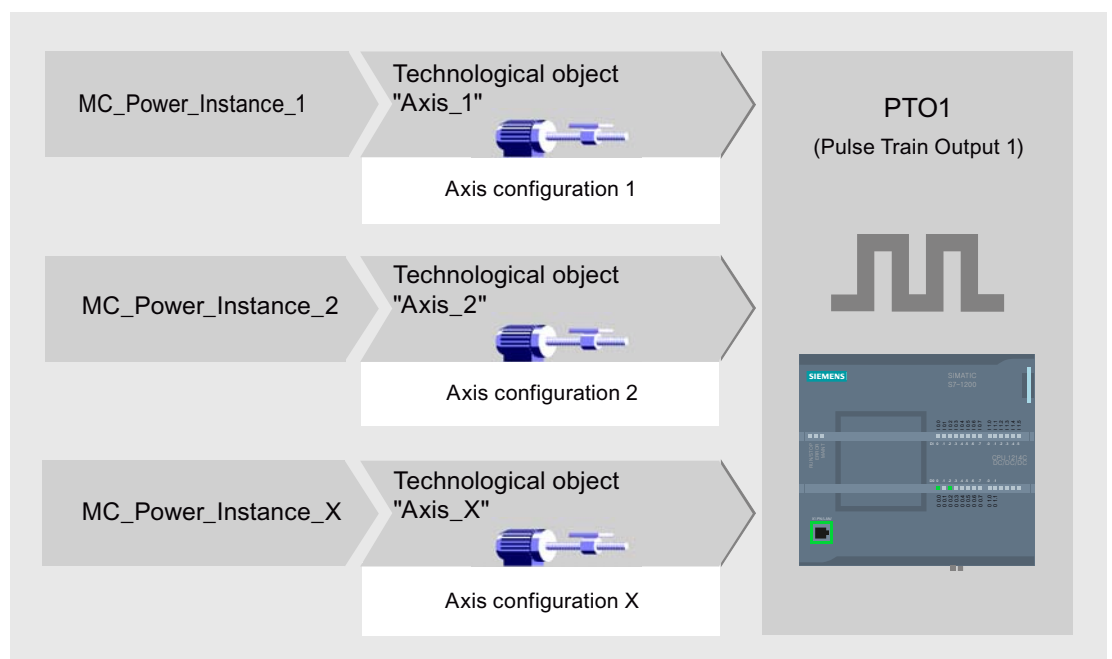
See also

Commissioning the axis - Axis control panel (Page 76)

1.12 Appendix

1.12.1 Using multiple axes with the same PTO

Use the motion control functionality of the CPU S7-1200 to run multiple "Axis" technology objects with the same PTO (Pulse Train Output) and thus with the same CPU outputs. This is appropriate, for example, if different axis configurations are to be used for different production sequences via one PTO. As described below, it is possible to switch between these axis configurations as often as necessary. The following diagram presents the basic functional relationships:

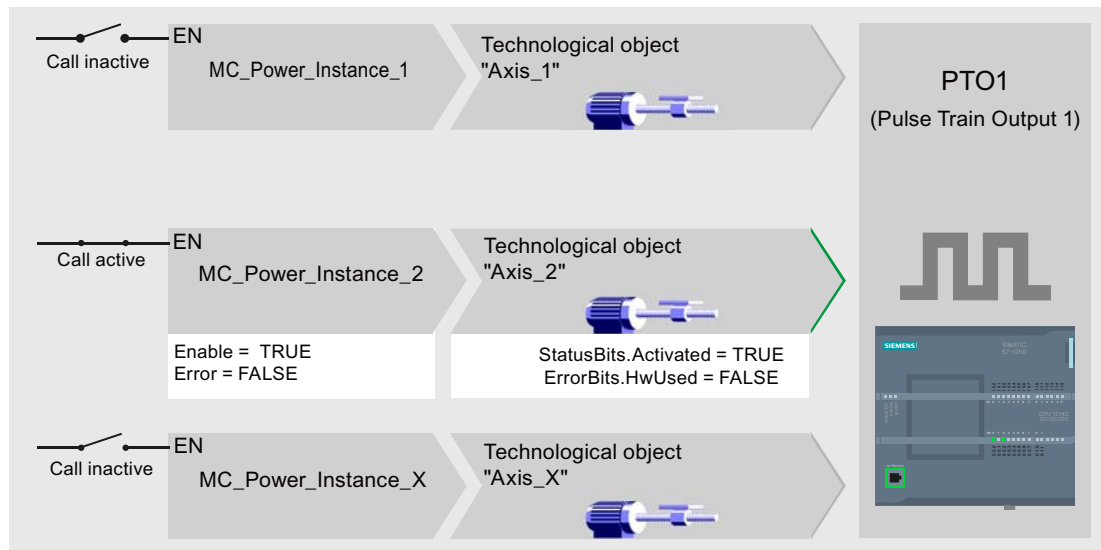


In this example, more than one "Axis" technology object, each with its own axis configuration, uses the same PTO. Each "Axis" must be called in the user program with a separate call of Motion Control instruction "MC_Power" with a separate instance data block. Only one "Axis" at a time may use the PTO. The axis that is currently using the PTO indicates this with tag <Axis name>.StatusBits.Activated = TRUE.

Switching between "Axis" technology objects

The program scheme described below shows you how to switch between different technology objects and, thus, between different axis configurations. To use the same PTO with multiple axes without error indications, only the Motion Control instructions of the axis currently being used may be called.

The following diagram presents this principle using Motion Control instruction "MC_Power" as an example:



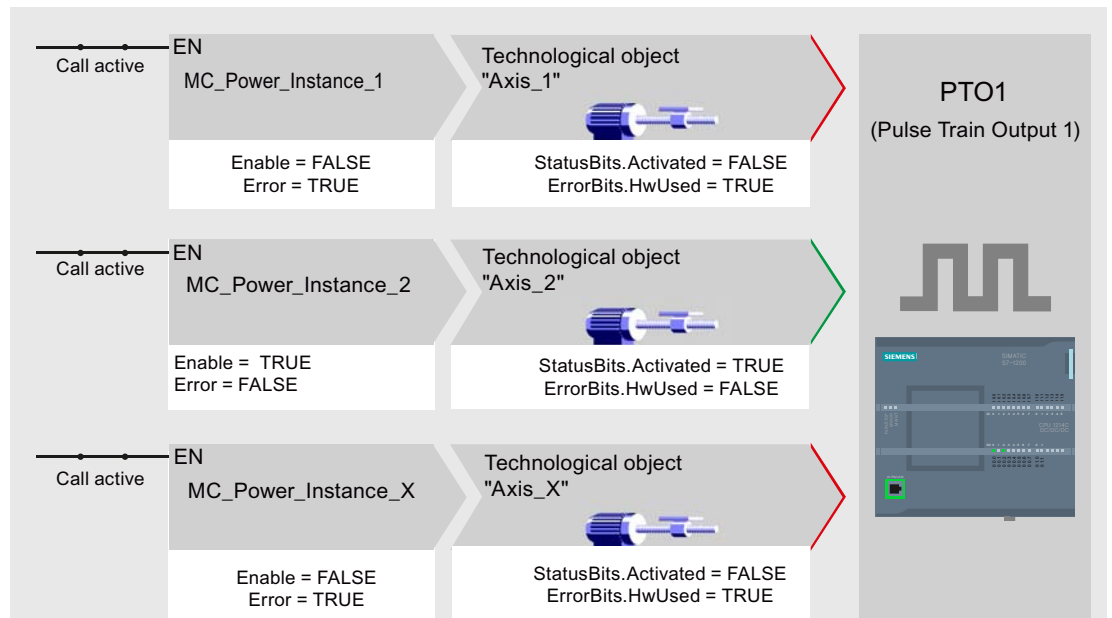
The tags of the activated axis ("Axis_2" here) show the following typical indicators in the user program:

- <Axis name>.StatusBits.Activated = TRUE
- <Axis name>.ErrorBits.HwUsed = FALSE

To switch to the "Axis" technology object, follow the steps described below. In the example, a switch is made from "Axis_2" to "Axis_1":

1. End any active traversing motions of activated "Axis_2"
2. Disable "Axis_2" with the associated Motion Control instruction "MC_Power" using input parameter Enable = FALSE
3. To verify that "Axis_2" has been disabled, use an AND operation of output parameter Status = FALSE of Motion Control instruction "MC_Power" and technology object tag <Axis name>.StatusBits.Enable = FALSE.
4. Deactivate the conditional call of the Motion Control instructions for "Axis_2".
5. Activate the conditional call of the Motion Control instructions for "Axis_1". On the first call of the corresponding Motion Control instruction "MC_Power", "Axis_2" becomes deactivated and "Axis_1" becomes activated.
6. Enable "Axis_1" with Motion Control instruction "MC_Power" using the input parameter Enable = TRUE
7. To verify that "Axis_1" has been enabled, use an AND operation of output parameter Status = TRUE of Motion Control instruction "MC_Power" and technology object tag <Axis name>.StatusBits.Enable = TRUE.

It is also always possible to cyclically call all Motion Control instructions of all axes working with a single PTO.



When an axis is enabled (here "Axis_2"), this axis becomes active.

In contrast to the conditional call, the Motion Control instructions of the deactivated axes (here "Axis_1" and "Axis_x") will indicate errors. The tags of these axes indicate the status `<Axis name>.StatusBits.Activated = FALSE` and `<Axis name>.ErrorBits.HwUsed = TRUE`.

Use the conditional call of the Motion Control instructions if you want to implement the user program without error indicators.

See also

Using multiple drives with the same PTO (Page 108)

Tracking jobs from higher priority classes (execution levels) (Page 109)

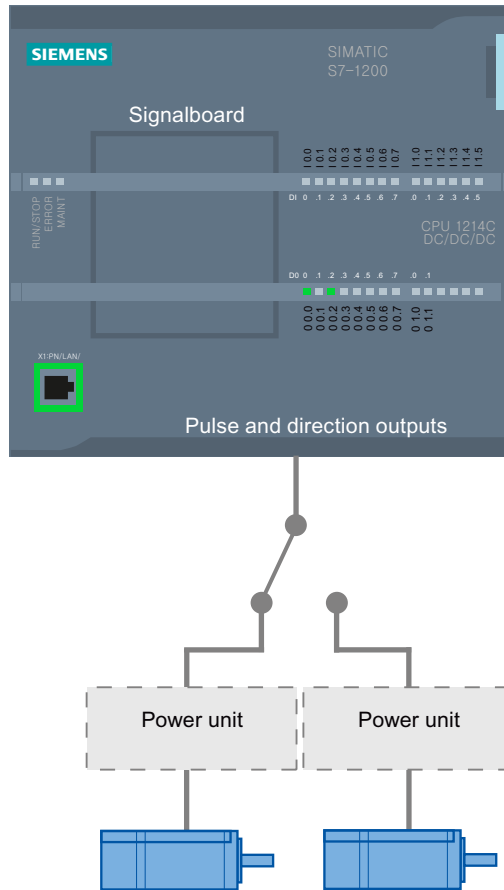
Special cases for use of software limit switches (Page 112)

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

Tag of the Axis technology object (Page 130)

1.12.2 Using multiple drives with the same PTO

If multiple drives are to be used, they can be run with a common PTO (Pulse Train Output) using changeover. The following diagram represents the basic circuit design:



The changeover between drives can be controlled, if required, by the user program via a digital output. If different axis configurations are required for the different drives, a changeover between these configurations is required for the PTO. For additional information on this topic, refer to "Using multiple axes with the same PTO (Page 105)".

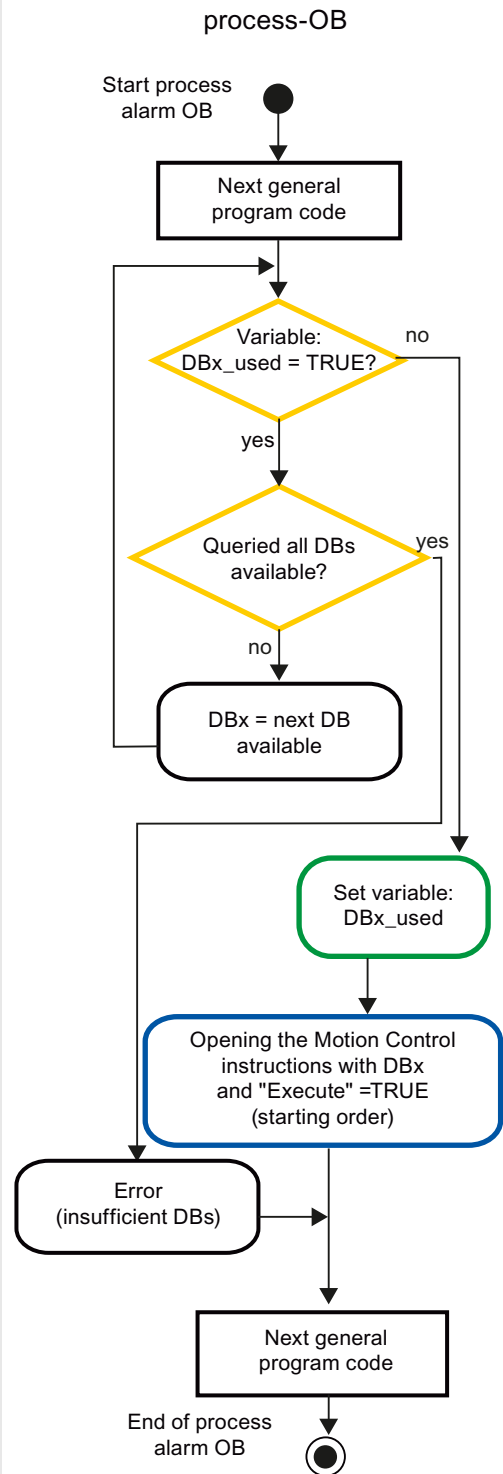
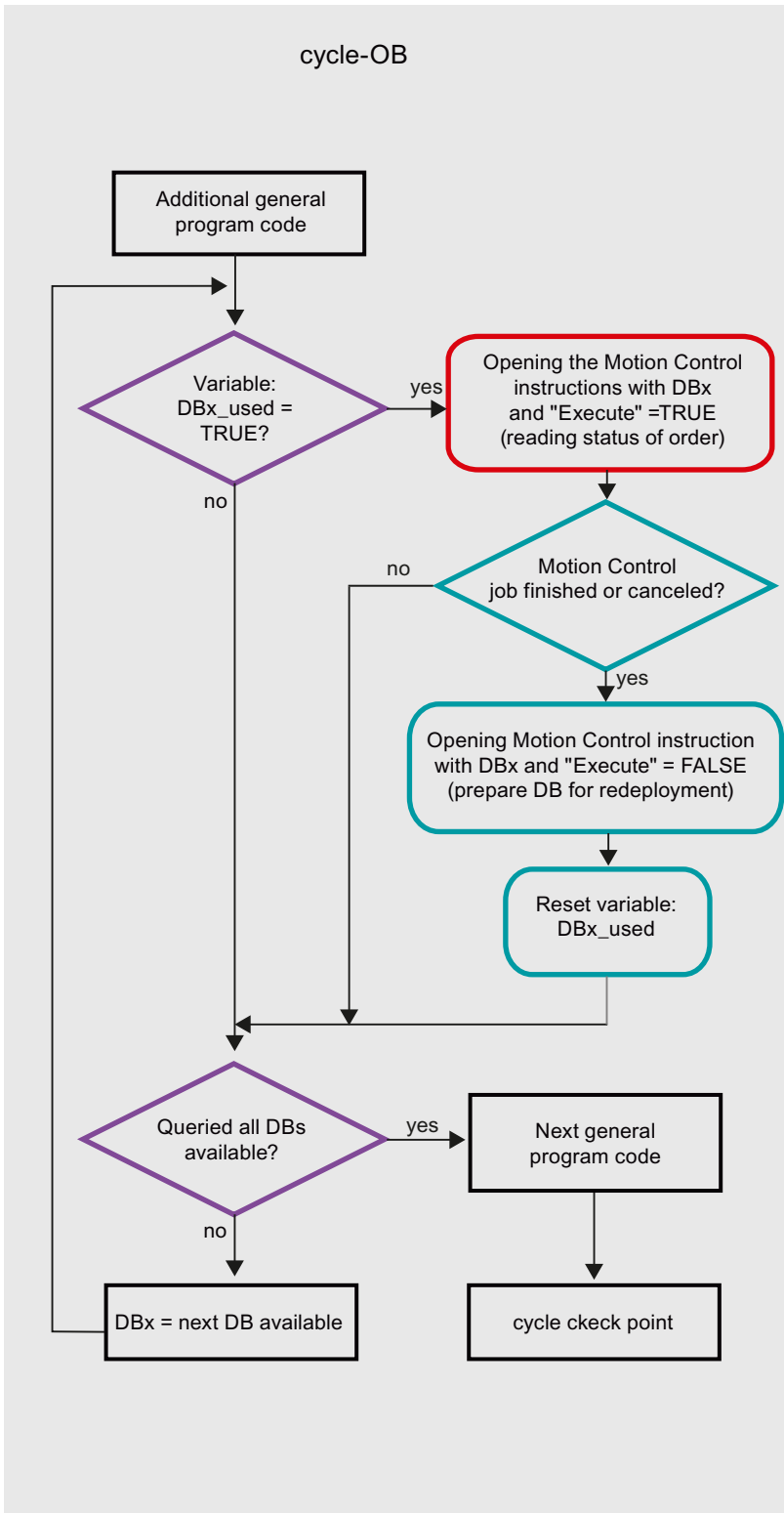
See also

- Using multiple axes with the same PTO (Page 105)
- Tracking jobs from higher priority classes (execution levels) (Page 109)
- Special cases for use of software limit switches (Page 112)
- List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)
- Tag of the Axis technology object (Page 130)

1.12.3 Tracking jobs from higher priority classes (execution levels)

Depending on the application, it may be necessary to start motion control jobs (for example, interrupt-controlled) in a higher priority class (execution level).

The Motion Control instructions must be called at short intervals for status monitoring. Motion Control commands cannot be sufficiently closely monitored if the higher priority Motion Control commands are called only once or at too great an interval. Tracking in such cases can be carried out in the cycle OB. An instance data block that is not currently being utilized must be available for each start of a higher priority motion control command. Refer to the following flow chart to see how you start motion control jobs in a higher priority class (for example, hardware interrupt OB) and continue tracking in the program cycle OB:



Depending on the frequency of the motion control jobs you want to start, you will have to generate a sufficient number of instance data blocks. Users determine which instance data block is currently used in the DBx_used tags.

Start of motion control job in the hardware interrupt OB

Binary queries of the DBx_used tags (orange) are used to find an instance data block not currently in use. If such an instance data block is found, the utilized instance data block is marked as "used" (green) and the Motion Control job is started with this instance data block.

Any other program sections of the hardware interrupt OB are then executed, followed by a return to the program cycle OB.

Tracking of started motion control jobs in the program cycle OB

All instance data blocks available in the cycle OB are checked to determine if they are currently in use by means of the DBx_used tag (violet).

If an instance data block is in use (motion control job is being processed), the motion control instruction with this instance data block and input parameter Execute = TRUE is called to read out the status messages (red).

If the job is complete or has been aborted, the following actions are taken next (blue green):

- Call of motion control instruction with input parameter Execute = FALSE
- Resetting the DBx_used tag

This completes the job tracking, and the instance data block is now available for use again.

See also

Using multiple axes with the same PTO (Page 105)

Using multiple drives with the same PTO (Page 108)

Special cases for use of software limit switches (Page 112)

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

Tag of the Axis technology object (Page 130)

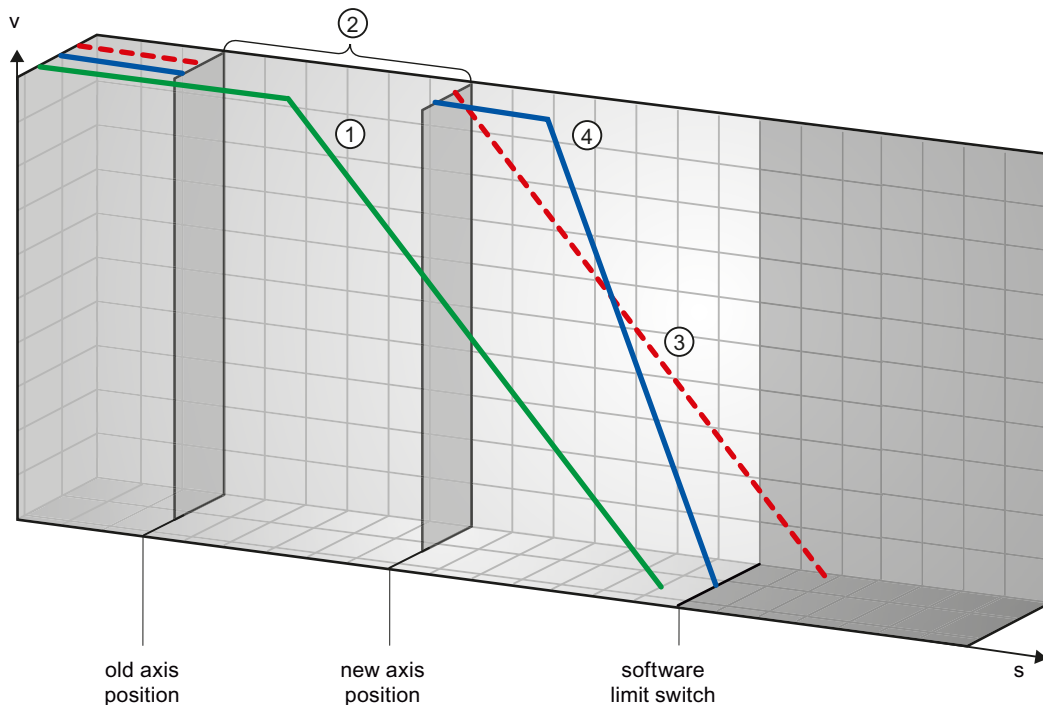
1.12.4 Special cases for use of software limit switches

1.12.4.1 Software limit switches in conjunction with a homing operation

Due to unfavorably parameterized homing jobs, the braking action of the axis may be influenced at the software limit switch. Take the following examples into consideration when developing your program.

Example 1:

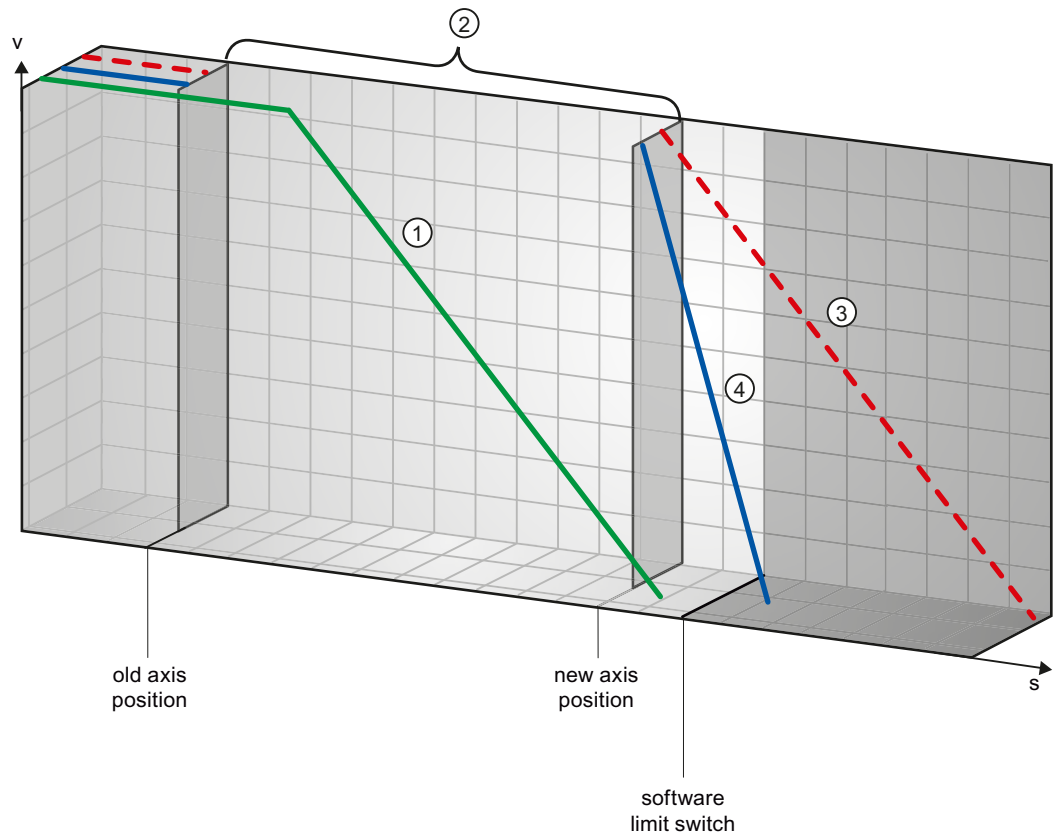
During a travel command, a homing job (for example, Set reference point) offsets the current axis position in the direction of the software limit switch. It is still possible to bring the axis to a standstill before reaching the software limit switch:



①	The green curve shows the motion without the homing job. The axis brakes at the configured deceleration and comes to a standstill at a position before the software limit switch.
②	A new axis position is set as a result of the homing job. The area between the old and new axis position is thus "skipped".
③	Based on the new axis position, the axis would theoretically be stopped with the configured deceleration at a position after the software limit switch (red curve).
④	Because braking with the configured deceleration is no longer sufficient, the axis actually follows the blue curve. Following a constant motion, the axis brakes at the emergency stop deceleration and comes to a standstill at the position of the software limit switch.

Example 2:

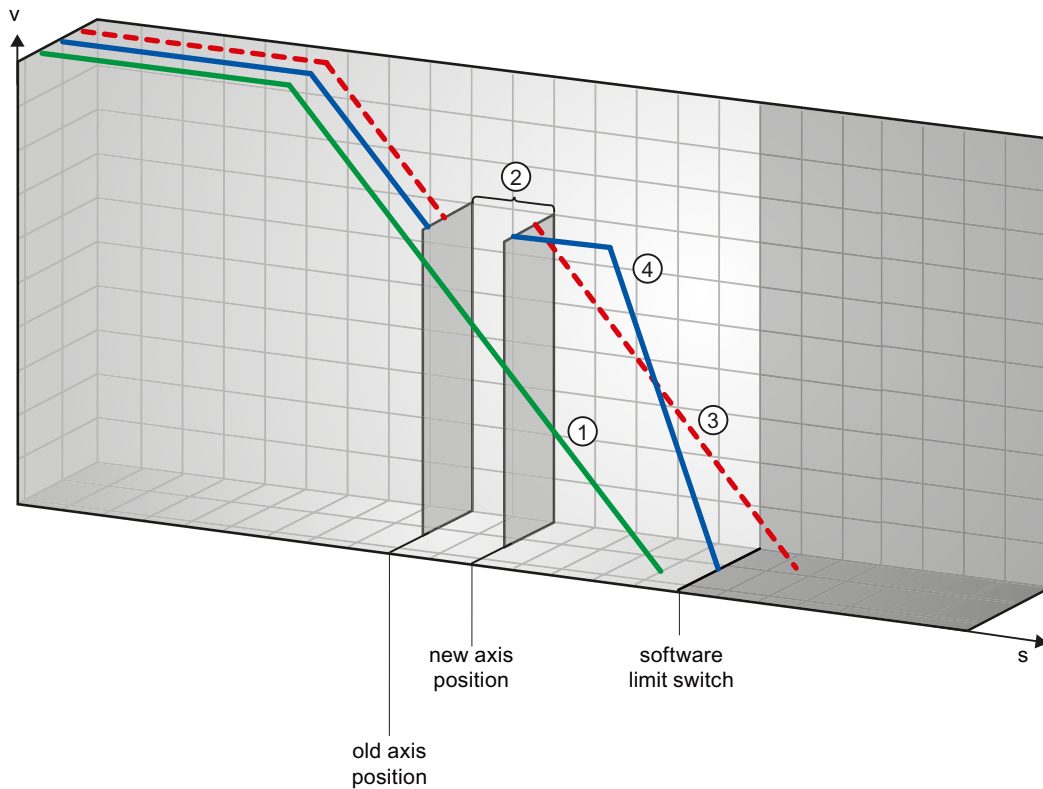
During a travel command, a homing job (for example, Set reference point) offsets the current axis position in the direction of the software limit switch. In contrast to example 1, it is no longer possible to bring the axis to a standstill before reaching the software limit switch. The axis overruns the position of the software limit switch.



①	The green curve shows the motion without the homing job. The axis brakes at the configured deceleration and comes to a standstill at a position before the software limit switch.
②	A new axis position is set as a result of the homing job. The area between the old and new axis position is thus "skipped".
③	Based on the new axis position, the axis would theoretically be stopped with the configured deceleration at a position well after the software limit switch (red curve).
④	Because braking with the configured deceleration is no longer sufficient, the axis actually follows the blue curve. The axis brakes at the emergency stop deceleration. However, the emergency stop deceleration is not sufficient to stop the axis at the position of the software limit switch. The position of the software limit switch is overrun.

Example 3:

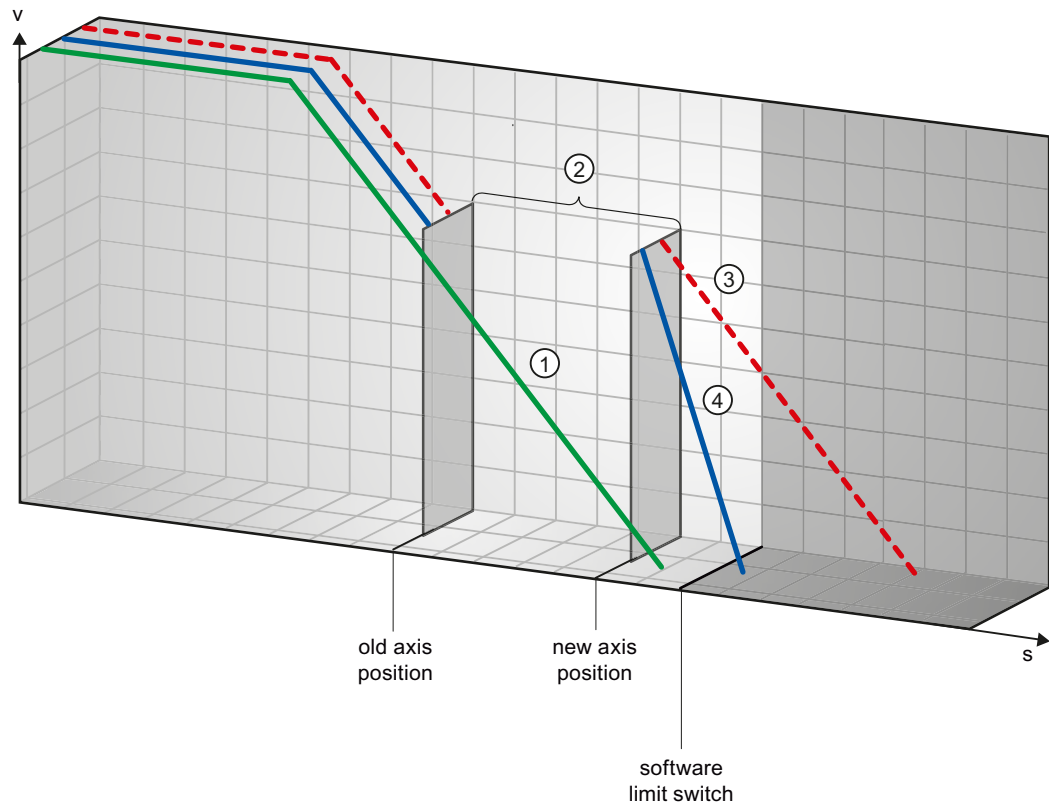
During a braking operation, a homing job (for example, Set reference point) offsets the current axis position in the direction of the software limit switch. It is still possible to bring the axis to a standstill before reaching the software limit switch:



①	The green curve shows the motion without the homing job. The axis brakes at the configured deceleration and comes to a standstill at a position before the software limit switch.
②	A new axis position is set as a result of the homing job. The area between the old and new axis position is thus "skipped".
③	Based on the new axis position, the axis would theoretically be stopped with the configured deceleration at a position after the software limit switch (red curve).
④	Because braking with the configured deceleration is no longer sufficient, the axis actually follows the blue curve. Following a constant motion, the axis brakes at the emergency stop deceleration and comes to a standstill at the position of the software limit switch.

Example 4:

During a braking operation, a homing job (for example, Set reference point) offsets the current axis position in the direction of the software limit switch. In contrast to example 3, it is no longer possible to bring the axis to a standstill before reaching the software limit switch. The axis overruns the position of the software limit switch.



①	The green curve shows the motion without the homing job. The axis brakes at the configured deceleration and comes to a standstill at a position before the software limit switch.
②	A new axis position is set as a result of the homing job. The area between the old and new axis position is thus "skipped".
③	Based on the new axis position, the axis would theoretically be stopped with the configured deceleration at a position well after the software limit switch (red curve).
④	Because braking with the configured deceleration is no longer sufficient, the axis actually follows the blue curve. The axis brakes at the emergency stop deceleration. However, the emergency stop deceleration is not sufficient to stop the axis at the position of the software limit switch. The position of the software limit switch is overrun.

See also

Software limit switches and software limit switch position changes. (Page 116)

Software limit switches in conjunction with dynamic changes (Page 117)

Behavior of axis when position limits is tripped (Page 37)

1.12.4.2 Software limit switches and software limit switch position changes.

An incorrect change in the position of the software limit switch during the runtime of the user program can abruptly reduce the distance between the current axis position and the position of the software limit switch.

The axis response is similar to that described in Software limit switches in conjunction with a homing operation (Page 112).

See also

Software limit switches in conjunction with a homing operation (Page 112)

Software limit switches in conjunction with dynamic changes (Page 117)

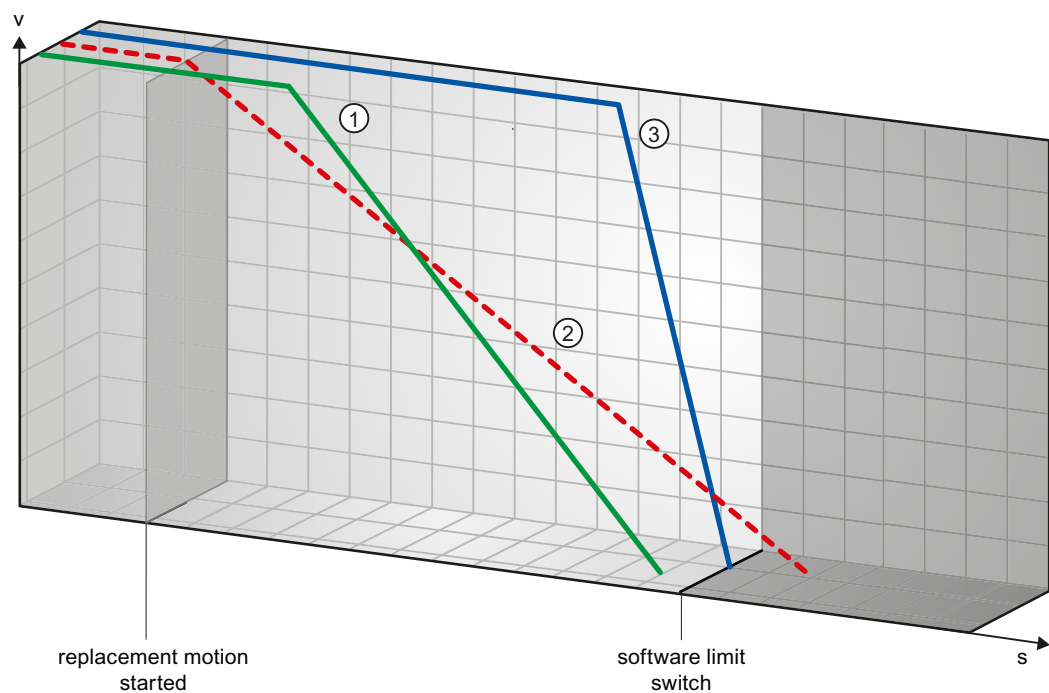
Behavior of axis when position limits is tripped (Page 37)

1.12.4.3 Software limit switches in conjunction with dynamic changes

It is possible to influence the deceleration of the axis in the area of the software limit switches in conjunction with overriding motion jobs. This applies when the overriding motion command is started with a lower deceleration (tag <Axis name>.Config.DynamicDefaults.Deceleration). Take the following examples into consideration when developing your program.

Example 1:

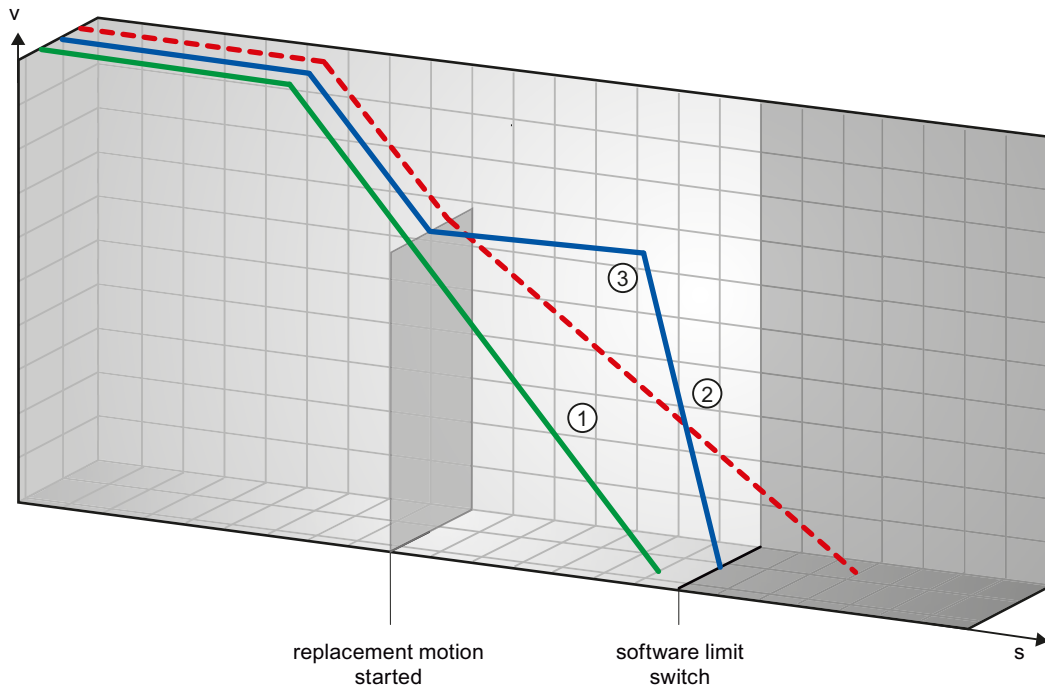
During axis motion, an active motion job is overridden by another motion job with a lower deceleration:



①	The green curve shows the motion of an active job without this job being overridden. The axis brakes at the configured deceleration and comes to a standstill at a position before the software limit switch.
②	Based on the overriding motion job with lower deceleration, the axis would theoretically be stopped with the configured deceleration at a position after the software limit switch (red curve).
③	Because braking with the configured deceleration of the overriding motion job is no longer sufficient, the axis actually follows the blue curve. Following a constant motion, the axis brakes at the emergency stop deceleration and comes to a standstill at the position of the software limit switch.

Example 2:

During braking of the axis, an active motion job is overridden by another motion job with a lower deceleration:



①	The green curve shows the motion of an active job without this job being overridden. The axis brakes at the configured deceleration and comes to a standstill at a position before the software limit switch.
②	Based on the overriding motion job with lower deceleration, the axis would theoretically be stopped at a position well after the software limit switch (red curve).
③	Because braking with the configured deceleration of the overriding motion job is no longer sufficient, the axis actually follows the blue curve. Following a constant motion, the axis brakes at the emergency stop deceleration and comes to a standstill at the position of the software limit switch.

See also

Software limit switches in conjunction with a homing operation (Page 112)

Software limit switches and software limit switch position changes. (Page 116)

Behavior of axis when position limits is tripped (Page 37)

1.12.5 Reducing velocity for a short positioning duration

The CPU can reduce the velocity of a positioning command when the planned positioning duration is < 2 ms.

The velocity of command execution will then be reduced for the entire duration.

The reduced velocity (pulses per s) is calculated as follows:

- Reduced velocity = Number of pulses to be output * 500Hz

Velocity is **not** reduced if the planned positioning duration is ≥ 2 ms.

1.12.6 Dynamic adjustment of start/stop velocity

The configuration of your velocity limits (start/stop velocity and maximum velocity), the dynamic values (acceleration, deceleration and jerk) and the target speed of the traversing command may under certain circumstances result in the start/stop velocity being dynamically adjusted by the CPU.

This is for example the case when, due to a low configured start/stop velocity, the time required for the first pulses would be longer than that possible for the entire acceleration. The first pulse is in these cases output at a greater velocity than the configured start/stop velocity. The subsequent pulses are also dynamically adjusted to ensure the acceleration process can be completed in the specified time.

Ensure in the event of any pulse loss that the hardware (drive) you use is adjusted to this situation, or change the dynamic settings of your axis to avoid dynamic adjustment of the start/stop velocity.

1.12.7 List of ErrorIDs and ErrorInfos (technology objects as of V2.0)

The following table lists all ErrorIDs and ErrorInfos that can be indicated in Motion Control instructions. In addition to the cause of the error, remedies for eliminating the error are also listed:

Operating error with stop of the axis

ErrorID	ErrorInfo	Description	Remedy
16#8000		Drive error, "Drive ready" failure	
	16#0001	-	Acknowledge error with instruction "MC_Reset"; provide drive signal; possibly restart command
16#8001		Low software limit switch has been tripped	
	16#000E	The position of the low software limit switch was reached with the currently configured deceleration	Acknowledge the error with instruction "MC_Reset"; use a motion command to move the axis in the positive direction out of the range of the software limit switch
	16#000F	The position of the low software limit switch was reached with the emergency stop deceleration	
	16#0010	The position of the low software limit switch was exceeded with the emergency stop deceleration	
16#8002		High software limit switch has been tripped	
	16#000E	The position of the high software limit switch was reached with the currently configured deceleration	Acknowledge the error with instruction "MC_Reset"; use a motion command to move the axis in the negative direction out of the range of the software limit switch
	16#000F	The position of the high software limit switch was reached with the emergency stop deceleration	
	16#0010	The position of the high software limit switch was exceeded with the emergency stop deceleration	
16#8003		Low hardware limit switch was approached	
	16#000E	The low hardware limit switch was approached. The axis was stopped with the emergency stop deceleration. (During an active reference point approach, the homing switch was not found)	Acknowledge the error for a released axis with instruction "MC_Reset"; use a motion command to move the axis in the positive direction out of the range of the hardware limit switch.
16#8004		High hardware limit switch has been approached	
	16#000E	The high hardware limit switch was approached. The axis was stopped with the emergency stop deceleration. (During an active reference point approach, the homing switch was not found)	Acknowledge the error for a released axis with instruction "MC_Reset"; use a motion command to move the axis in the negative direction out of the range of the hardware limit switch.

ErrorID	ErrorInfo	Description	Remedy
16#8005		PTO/HSC are already being used by another axis	
	16#0001	-	<p>The axis was configured incorrectly: Correct the configuration of the PTO (Pulse Train Output) / HSC (High Speed Counter) and download it to the controller</p> <p>More than one axis is to run with one PTO: Another axis is using the PTO/HSC. If the current axis is to assume the control, the other axis must be disabled with "MC_Power" Enable = FALSE. (see also Using multiple axes with the same PTO (Page 105))</p>
16#8006		A communication error in the control panel has occurred	
	16#0012	A timeout has occurred	Check the cable connection and press the "Manual control" button again.
16#8007		It is not possible to release the axis	
	16#0025	Restarting	Wait until the axis restart is complete.
	16#0026	Executing loading process in RUN mode	Wait until the loading process is complete.

Operating error without stop of the axis

ErrorID	ErrorInfo	Description	Remedy
16#8200		Axis is not released	
	16#0001	-	Release the axis; restart the command
16#8201		Axis has already been released by another "MC_Power" instance	
	16#0001	-	Release the axis with only one "MC_Power" instance
16#8202		The maximum number of simultaneously active motion control commands has been exceeded (maximum of 200 commands for all motion control technology objects)	
	16#0001	-	Reduce the number of simultaneously active jobs; restart the command A command is active if parameter "Busy" = TRUE in the Motion Control instruction.
16#8203		Axis is currently operated in "Manual control" (axis command table)	
	16#0001	-	Exit "Manual control"; restart the command
16#8204		Axis is not homed	
	16#0001	-	Home the axis with instruction "MC_Home"; restart the command
16#8205		The axis is currently controlled by the user program (the error is only displayed in the axis command table)	
	16#0013	The axis is released in the user program.	Disable axis with instruction "MC_Power" and select "Manual control" again in the axis command table
16#8206		Technology object not activated yet	
	16#0001	-	Release the axis with instruction "MC_Power" Enable = TRUE or release the axis in the axis command table.
16#8207		Command rejected	
	16#0016	Active homing is running; another homing method cannot be started.	Wait for active homing to finish or abort active homing with a motion command, for example, "MC_Halt".
	16#0018	The axis cannot be moved with a command table whilst it is being directly or passively homed.	Wait until direct or passive homing is complete.
	16#0019	The axis cannot be directly or passively homed whilst a command table is being processed.	Wait for command table to finish or abort the command table with a motion command, for example, "MC_Halt".
16#8208		Difference between maximum and start/stop velocity is invalid	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#000A	Value is less than or equal to 0	
16#8209		Invalid acceleration for technology object "Axis"	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#000A	Value is less than or equal to 0	
16#820A		It is not possible to restart the axis	
	16#0013	The axis is released in the user program.	Block the axis with the "MC_Power" instruction; restart again

ErrorID	ErrorInfo	Description	Remedy
	16#0027	The axis is currently being operated in "Manual control" (axis control panel)	Exit "Manual control"; restart again
16#820B		It is not possible to execute the command table	
	16#0026	Executing loading process in RUN mode	Wait until the loading process is complete.

Block parameter error

ErrorID	ErrorInfo	Description	Remedy
16#8400		Invalid value at parameter "Position" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#8401		Invalid value at parameter "Distance" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#8402		Invalid value at parameter "Velocity" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#0008	Value is greater than the configured maximum velocity	
	16#0009	Value is less than the configured start/stop velocity	
	16#0024	Value is less than 0	
16#8403		Invalid value at parameter "Direction" of the Motion Control instruction	
	16#0011	The selection value is invalid	Correct the selection value; restart the command
16#8404		Invalid value at parameter "Mode" of the Motion Control instruction	
	16#0011	The selection value is invalid	Correct the selection value; restart the command
	16#0015	Active/passive homing is not configured	Correct the configuration and download it to the controller; release the axis and restart the command
	16#0017	The direction reversal is activated at the hardware limit switch, despite the fact that the hardware limit switches are disabled	<ul style="list-style-type: none"> • Activate the hardware limit switch using the tag <code><Axis>.Config.PositionLimits_HW.Active = TRUE</code>, restart the command • Correct the configuration and download it to the controller; release the axis and restart the command
16#8405		Invalid value at parameter "StopMode" of the Motion Control instruction	
	16#0011	The selection value is invalid	Correct the selection value; release the axis again

ErrorID	ErrorInfo	Description	Remedy
16#8406		Simultaneous forward and backward jogging is not allowed	
	16#0001	-	Take steps to ensure that parameters "JogForward" and "JogBackward" do not have signal status TRUE simultaneously; restart the command.
16#8407		Switching to another axis with instruction "MC_Power" is only permitted after disabling the active axis.	
	16#0001	-	Disable the active axis; it is then possible to switch to the other axis and release it.
16#8408		Invalid value at parameter "Axis" of the Motion Control instruction	
	16#001A	The specified value does not match the required technology object version	Correct the value; restart the command
	16#001B	The specified value does not match the required technology object type	
	16#001C	The specified value is not a Motion Control technology data block	
16#8409		Invalid value at parameter "CommandTable" of the Motion Control instruction	
	16#001A	The specified value does not match the required technology object version	Correct the value; restart the command
	16#001B	The specified value does not match the required technology object type	
	16#001C	The specified value is not a Motion Control technology data block	
16#840A		Invalid value at parameter "StartStep" of the Motion Control instruction	
	16#000A	Value is less than or equal to 0	Correct the value; restart the command
	16#001D	The start step is greater than the end step	
	16#001E	Value is greater than 32	
16#840B		Invalid value at parameter "EndStep" of the Motion Control instruction	
	16#000A	Value is less than or equal to 0	Correct the value; restart the command
	16#001E	Value is greater than 32	
16#840C		Invalid value at parameter "RampUpTime" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#000A	Value is less than or equal to 0	
16#840D		Invalid value at parameter "RampDownTime" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#000A	Value is less than or equal to 0	
16#840E		Invalid value at parameter "EmergencyRampTime" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#000A	Value is less than or equal to 0	
16#840F		Invalid value at parameter "JerkTime" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the value; restart the command
	16#000A	Value is less than or equal to 0	

Configuration error of the axis

ErrorID	ErrorInfo	Description	Remedy
16#8600		Parameterization of pulse generator (PTO) is invalid	
	16#000B	The address is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#0014	The selected hardware is used by another application	
16#8601		Parameterization of the high-speed counter (HSC) is invalid	
	16#000B	The address is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#0014	The selected hardware is used by another application	
16#8602		Invalid parameter assignment of "Enable-Output"	
	16#000B	The address is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
16#8603		Invalid parameter assignment of "Ready-Input"	
	16#000B	The address is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
16#8604		Invalid "Pulses per motor revolution" value	
	16#000A	Value is less than or equal to zero	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
16#8605		Invalid "Load distance per motor revolution" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#000A	Value is less than or equal to zero	
16#8606		Invalid "Start / stop velocity" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#0003	Value exceeds the high hardware limit	
	16#0004	Value is less than the low hardware limit	
	16#0007	The start/stop velocity is greater than the maximum velocity	
16#8607		Invalid "maximum velocity" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#0003	Value exceeds the high hardware limit	
	16#0004	Value is less than the low hardware limit	

ErrorID	ErrorInfo	Description	Remedy
16#8608		Invalid "Acceleration" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0003	Value exceeds the high hardware limit	
	16#0004	Value is less than the low hardware limit	
16#8609		Invalid "Deceleration" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0003	Value exceeds the high hardware limit	
	16#0004	Value is less than the low hardware limit	
16#860A		Invalid "Emergency stop deceleration" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0003	Value exceeds the high hardware limit	
	16#0004	Value is less than the low hardware limit	
16#860B		Value for position of the low SW limit switch is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
	16#0007	The position value of the low software limit switch is greater than that of the high software limit switch	
16#860C		Value for position of the high SW limit switch is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#860D		Invalid address of the low HW limit switch	
	16#000C	The address of the falling edge is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#000D	The address of the rising edge is invalid	
16#860E		Invalid address of the high HW limit switch	
	16#000C	The address of the falling edge is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#000D	The address of the rising edge is invalid	

ErrorID	ErrorInfo	Description	Remedy
16#860F		Invalid "home position offset" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#8610		Invalid "approach velocity" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0008	The velocity is greater than the maximum velocity	
	16#0009	The velocity is less than the start/stop velocity	
16#8611		Invalid "Homing velocity" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#0008	The velocity is greater than the maximum velocity	
	16#0009	The velocity is less than the start/stop velocity	
16#8612		Invalid address of the homing switch	
	16#000C	The address of the falling edge is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power"
	16#000D	The address of the rising edge is invalid	
16#8613		During active homing, direction reversal at the hardware limit switch is activated although the hardware limit switches are not configured	
	16#0001	-	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
16#8614		Invalid "Jerk" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and if necessary restart the command
	16#001F	Value is greater than the maximum jerk	
	16#0020	Value is less than the minimum jerk	
16#8615		Value for "Unit of measurement" is invalid	
	16#0011	The selection value is invalid	Download error-free configuration to the controller; release the axis again with instruction "MC_Power"

Configuration error of the command table

ErrorID	ErrorInfo	Description	Remedy
16#8700		Value for "Command type" in the command table is invalid	
	16#0001	-	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online and if necessary restart the command
16#8701		Value for "Position / travel path" in the command table is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online and if necessary restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#8702		Value for "Velocity" in the command table is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online and if necessary restart the command
	16#0008	Value is greater than the configured maximum velocity	
	16#0009	Value is less than the configured start/stop velocity	
16#8703		Value for "Duration" in the command table is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online and if necessary restart the command
	16#0021	Value is greater than 64800 s	
	16#0022	Value is less than 0.001 s	
16#8704		Value for "Next step" in the command table is invalid	
	16#0011	The selection value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; release the axis again with instruction "MC_Power" Correct the incorrect value online and if necessary restart the command
	16#0023	The command transition is not permitted for this command	

Internal errors

ErrorID	ErrorInfo	Description	Remedy
16#8FFF		Internal error	
	16#F0**	-	POWER OFF and POWER ON the CPU If this does not work, contact Customer Support. Have the following information ready: <ul style="list-style-type: none"> • ErrorID • ErrorInfo • Diagnostic buffer entries

See also

Using multiple axes with the same PTO (Page 105)

Using multiple drives with the same PTO (Page 108)

Tracking jobs from higher priority classes (execution levels) (Page 109)

Special cases for use of software limit switches (Page 112)

Tag of the Axis technology object (Page 130)

1.12.8 Tag of the Axis technology object

1.12.8.1 Config. tag

Config.General. tag

Legend

Data type	Data type of the tag
Start value	Start value of tag The initial value can be overwritten by the axis configuration.
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.Config.General.PTO				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWORD	DW#16#00000000	-	-	-

<Axis name>.Config.General.HSC				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWORD	DW#16#00000000	-	-	-

<Axis name>.Config.General.LengthUnit ("Axis" technology object as of V2.0)				
The unit of measurement for the parameter selected in the configuration:				
<ul style="list-style-type: none"> • 1013 = "mm" • 1010 =: "m" • 1019 = "in" • 1018 = "ft" • 1005 = "°" (degrees) • -1 = "Pulse" 				
Data type	Start value	Access	Effective	HMI
Int	1013	R	-	X

Config.DriveInterface. tag**Legend**

Data type	Data type of the tag
Start value	Start value of tag The initial value can be overwritten by the axis configuration.
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.Config.DriveInterface.EnableOutput...				
Tags cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
-	-	-	-	-

<Axis name>.Config.DriveInterface.ReadyInput...				
Tags cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
-	-	-	-	-

Config.Mechanics. tag

Legend

Data type	Data type of the tag
Start value	Start value of tag The initial value can be overwritten by the axis configuration.
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.Config.Mechanics.PulsesPerDriveRevolution)				
Increments per motor revolution				
Data type	Start value	Access	Effective	HMI
DInt	L#1000	R	-	X

<Axis name>.Config.Mechanics.LeadScrew				
Load distance per motor revolution (specified in the configured unit of measurement)				
Data type	Start value	Access	Effective	HMI
Real	1.0E+001	R	-	X

<Axis name>.Config.Mechanics.InverseDirection				
Invert direction signal				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

Config.DynamicLimits. tag**Legend**

Data type	Data type of the tag
Start value	Start value of tag The initial value can be overwritten by the axis configuration.
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.Config.DynamicLimits.MinVelocity				
Start/stop velocity of axis (specified in the configured unit of measurement)				
Data type	Start value	Access	Effective	HMI
Real	1.0E+001	R	-	X

<Axis name>.Config.DynamicLimits.MaxVelocity				
Maximum velocity of axis (specified in the configured unit of measurement)				
Data type	Start value	Access	Effective	HMI
Real	2.5E+002	R	-	X

Config.DynamicDefaults. tag

Legend

Data type	Data type of the tag	
Start value	Start value of tag The initial value can be overwritten by the axis configuration.	
Access	Access to the tag in the user program:	
	RW	The tag can be read and written in the user program.
	R	The tag can be read in the user program.
	-	The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.	
	1	When axis is activated (tag <Axis name>.StatusBits.Activated changes from FALSE -> TRUE), blocked or released
	2	When axis is enabled
	5	The next time a MC_MoveAbsolute-, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt, MC_CommandTable or active MC_Home -command is started (Mode = 3).
	6	When a MC_MoveJog command is stopped
HMI	The tag can be used in an HMI system.	

<Axis name>.Config.DynamicDefaults.Acceleration					
Acceleration of axis (specified in the configured dimension unit)					
Data type	Start value	Access	Effective		HMI
Real	4.8E+001	RW	5	CPU Firmware V1.0	X
			1, 5, 6	CPU firmware as of V2.0	

<Axis name>.Config.DynamicDefaults.Deceleration					
Deceleration of axis (specified in the configured dimension unit)					
Data type	Start value	Access	Effective		HMI
Real	4.8E+001	RW	5, 6	CPU Firmware V1.0	X
			1, 5, 6	CPU firmware as of V2.0	

<Axis name>.Config.DynamicDefaults.EmergencyDeceleration				
Emergency stop deceleration of axis (specified in the configured dimension unit)				
Data type	Start value	Access	Effective	HMI
Real	1.2E+002	RW	2, CPU Firmware V1.0	X
			5, 6	
			1, CPU firmware as of V2.0	
			5, 6	

<Axis name>.Config.DynamicDefaults.JerkActive ("Axis" technology object as of V2.0)				
TRUE = the jerk limit is activated				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	RW	1, 5	X

<Axis name>.Config.DynamicDefaults.Jerk ("Axis" technology object as of V2.0)				
Jerk during axis acceleration and deceleration ramp (specified in the configured unit of measurement)				
Data type	Start value	Access	Effective	HMI
Real	1.92E+002	RW	1, 5	X

Config.PositionLimits_SW. tag

Legend

Data type	Data type of the tag
Start value	Start value of tag The initial value can be overwritten by the axis configuration.
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
	1 When axis is activated (tag <Axis name>.StatusBits.Activated changes from FALSE -> TRUE), blocked or released
	4 Upon the next start of a Motion Control command after a standstill of the axis. The axis standstill can be checked with tag <Axis name>. StatusBits.Standstill.
	5 The next time a MC_MoveAbsolute, MC_MoveRelative-, MC_MoveVelocity, MC_MoveJog, MC_Halt, MC_CommandTable or active MC_Homecommand is started (Mode = 3).
HMI	The tag can be used in an HMI system.

<Axis name>.Config.PositionLimits_SW.Active					
TRUE = The software limit switches are activated					
Data type	Start value	Access	Effective		HMI
Bool	FALSE	RW	4	CPU Firmware V1.0	X
			1, 5, 6	CPU firmware as of V2.0	

<Axis name>.Config.PositionLimits_SW.MinPosition					
Position of low software limit switch (specified in the configured unit of measurement)					
Data type	Start value	Access	Effective		HMI
Real	-1.0E+004	RW	4	CPU Firmware V1.0	X
			1, 5, 6	CPU firmware as of V2.0	

<Axis name>.Config.PositionLimits_SW.MaxPosition					
Position of high software limit switch (specified in the configured unit of measurement)					
Data type	Start value	Access	Effective		HMI
Real	1.0E+004	RW	4	CPU Firmware V1.0	X
			1, 5, 6	CPU firmware as of V2.0	

Config.PositionLimits_HW. tag

Legend

Data type	Data type of the tag	
Start value	Start value of tag The initial value can be overwritten by the axis configuration.	
Access	Access to the tag in the user program:	
	RW	The tag can be read and written in the user program.
	R	The tag can be read in the user program.
	-	The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.	
	1	When axis is activated (tag <Axis name>.StatusBits.Activated changes from FALSE -> TRUE), blocked or released
	3	After axis enable (the axis must have previously been at a standstill). The axis standstill can be checked with tag <Axis name>. StatusBits.Standstill.
	4	Upon the next start of a Motion Control command after a standstill of the axis. The axis standstill can be checked with tag <Axis name>. StatusBits.Standstill.
	5	The next time aMC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt, MC_CommandTable or active MC_Home command is started (Mode = 3).
HMI	The tag can be used in an HMI system.	

<Axis name>.Config.PositionLimits_HW.Active				
TRUE = The hardware limit switches are active.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	RW	3, 4	CPU Firmware V1.0
			1, 5, 6	CPU firmware as of V2.0
				X

<Axis name>.Config.PositionLimits_HW.MinSwitchedLevel				
TRUE = 24 V at CPU input corresponds to low hardware limit switch approached				
FALSE = 0 V at CPU input corresponds to low hardware limit switch approached				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.Config.PositionLimits_HW.MinFallingEvent				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWord	DW#16#00000000	-	-	-

<Axis name>.Config.PositionLimits_HW.MinRisingEvent				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWord	DW#16#00000000	-	-	-

<Axis name>.Config.PositionLimits_HW.MaxSwitchedLevel				
TRUE = 24 V at CPU input corresponds to high hardware limit switch approached				
FALSE = 0 V at CPU input corresponds to high hardware limit switch approached				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.Config.PositionLimits_HW.MaxFallingEvent				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWord	DW#16#00000000	-	-	-

<Axis name>.Config.PositionLimits_HW.MaxRisingEvent				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWord	DW#16#00000000	-	-	-

Config.Homing. tag**Legend**

Data type	Data type of the tag	
Start value	Start value of tag The initial value can be overwritten by the axis configuration.	
Access	Access to the tag in the user program:	
	RW	The tag can be read and written in the user program.
	R	The tag can be read in the user program.
	-	The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.	
	1	When axis is activated (tag <Axis name>.StatusBits.Activated changes from FALSE -> TRUE), blocked or released
	7	When a passive homing command is started
	8	When an active homing command is started
HMI	The tag can be used in an HMI system.	

<Axis name>.Config.Homing.AutoReversal					
TRUE = Direction reversal at hardware limit switch enabled (active homing) FALSE = Direction reversal at hardware limit switch disabled (active homing)					
Data type	Start value	Access	effective		HMI
Bool	TRUE	R	-	Technology object "Axis" V1.0	X
		RW	1, 8	Technology object "Axis" V2.0	

<Axis name>.Config.Homing.Direction					
TRUE = Positive approach direction to search for homing switch and positive homing direction (active homing) FALSE = Negative approach direction to search for homing switch and positive homing direction (active homing)					
Data type	Start value	Access	effective		HMI
Bool	TRUE	R	-	Technology object "Axis" V1.0	X
		RW	1, 8	Technology object "Axis" V2.0	

<Axis name>.Config.Homing.SideActiveHoming ("Axis" technology object as of V2.0)					
TRUE = Homing on high side of the homing switch (active homing) TRUE = Homing on lower side of the homing switch (active homing)					
Data type	Start value	Access	Effective		HMI
Bool	TRUE	RW	1, 8		X

<Axis name>.Config.Homing.SidePassiveHoming ("Axis" technology object as of V2.0)				
TRUE = Homing on high side of the homing switch (passive homing) TRUE = Homing on lower side of the homing switch (passive homing)				
Data type	Start value	Access	Effective	HMI
Bool	TRUE	RW	1, 7	X

<Axis name>.Config.Homing.RisingEdge (as of technology object "Axis" V1.0)				
TRUE = Homing with negative signal edge of the homing switch (active homing) FALSE = Homing with positive signal edge of the homing switch (active homing) For information on the effect of the tag on passive homing, refer to the description in "Configuration - Homing".				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.Config.Homing.Offset				
Home position offset /specified in the configured unit of measurement (active homing)				
Data type	Start value	Access	Effective	HMI
Real	0.0	R	-	Technology object "Axis" V1.0
		RW	1, 8	

<Axis name>.Config.Homing.FastVelocity				
Approach velocity / specified in the configured unit of measurement (active homing)				
Data type	Start value	Access	Effective	HMI
Real	2.0E+002	R	-	Technology object "Axis" V1.0
		RW	1, 8	

<Axis name>.Config.Homing.SlowVelocity				
Homing velocity / specified in the configured unit of measurement (active homing)				
Data type	Start value	Access	Effective	HMI
Real	4.0E+001	R	-	Technology object "Axis" V1.0
		RW	1, 8	

<Axis name>.Config.Homing.FallingEvent				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	Effective	HMI
DWord	DW#16#00000000	-	-	-

<Axis name>.Config.Homing.RisingEvent				
Tag cannot be evaluated in the user program.				
Data type	Start value	Access	effective	HMI
DWord	DW#16#00000000	-	-	

1.12.8.2 MotionStatus. tag

Legend

Data type	Data type of the tag
Start value	Start value of tag
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.MotionStatus.Position				
Current position of the axis (specified in the configured unit of measurement) If the axis is not homed, the tag indicates the position value relative to the enable position of the axis.				
Data type	Start value	Access	Effective	HMI
Real	0.0	R	-	X

<Axis name>..MotionStatus.Velocity				
Current velocity of the axis (specified in the configured unit of measurement)				
Data type	Start value	Access	Effective	HMI
Real	0.0	R	-	X

<Axis name>.MotionStatus.Distance				
Current distance to the target position of the axis (specified in the configured unit of measurement) The value of the tag is only valid during execution of a positioning command with "MC_MoveAbsolute" or "MC_MoveRelative" or of the axis command table.				
Data type	Start value	Access	Effective	HMI
Real	0.0	R	-	X

<Axis name>.MotionStatus.TargetPosition				
Target position of axis (specified in the configured unit of measurement) The value of the tag is only valid during execution of a positioning command with "MC_MoveAbsolute" or "MC_MoveRelative" or of the axis command table.				
Data type	Start value	Access	Effective	HMI
Real	0.0	R	-	X

See also

Motion status (Page 103)

1.12.8.3 StatusBits.tag

Legend

Data type	Data type of the tag
Start value	Start value of tag
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.StatusBits.Activated				
TRUE = The axis is activated. It is connected to the assigned PTO (Pulse Train Output). The data of the technology data block will be updated cyclically.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.Enable				
TRUE = The axis is enabled and ready to take on Motion Control jobs.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.HomingDone				
TRUE = The axis is homed and is capable of executing absolute positioning jobs. The axis does not have to be homed for relative homing. The status is FALSE during active homing. The status will remain TRUE during passive homing if the axis has already been homed.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.Done				
TRUE = No Motion Control command is active on the axis.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.Error				
TRUE = An error occurred in the axis technology object. Detailed information about the error is available in automatic mode in the ErrorID" and ErrorInfo" parameters of the motion control instructions. In manual mode, the "Error message" field of the axis command table displays detailed information about the cause of error.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.StandStill				
TRUE = The axis is at a standstill.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.PositioningCommand				
TRUE = The axis is executing a positioning command.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.SpeedCommand				
TRUE = The axis is executing a travel command at predefined velocity.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.Homing				
TRUE = The axis is executing a homing command of the "MC_Home" Motion Control instruction or axis command table.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.CommandTableActive				
TRUE = The axis is controlled by Motion Control instruction "MC_CommandTable".				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.ConstantVelocity				
TRUE = The axis travels at constant velocity.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.Acceleration				
TRUE = The axis accelerates.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.Deceleration				
TRUE = The axis decelerates (slows down).				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.ControlPanelActive				
TRUE = The "Manual control" mode has been enabled in the axis command table. The axis command table has control priority over the "Axis" technology object. The axis cannot be controlled from the user program.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.DriveReady				
TRUE = The drive is ready.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.StatusBits.RestartRequired				
TRUE = Values were modified in the load memory. To download the values in the CPU RUN operating mode to the work memory, you need to restart the axis. Use the Motion Control instruction MC_Reset to do this.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

See also

Status and error bits (Page 100)

1.12.8.4 ErrorBits. tag

Legend

Data type	Data type of the tag
Start value	Start value of tag
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.ErrorBits.SystemFault				
TRUE = Internal system error.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.ConfigFault				
TRUE = Incorrect configuration of axis.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.DriveFault				
TRUE = The drive has reported an error after failure of its "Drive ready" signal.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.SwLimitMinReached				
TRUE = The low software limit switch has been reached.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.SwLimitMinExceeded				
TRUE = The low software limit switch has been exceeded.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.SwLimitMaxReached				
TRUE = The high software limit switch has been reached.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.SwLimitMaxExceeded				
TRUE = The high software limit switch has been exceeded.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.HwLimitMin				
TRUE = The low hardware limit switch has been approached.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.HwLimitMax				
TRUE = The high hardware limit switch has been approached.				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

<Axis name>.ErrorBits.HwUsed				
TRUE = A second axis is using the same PTO (Pulse Train Output) and is enabled with "MC_Power".				
Data type	Start value	Access	Effective	HMI
Bool	FALSE	R	-	X

See also

Status and error bits (Page 100)

1.12.8.5 Internal. tag

The "Internal" tags contain no user-relevant data; these tags cannot be accessed in the user program.

1.12.8.6 ControlPanel tag

The "ControlPanel" tags contain no user-relevant data; these tags cannot be accessed in the user program.

1.12.8.7 Update of the technology object tags

The status and error information of the axis indicated in the technology object tags is updated at each cycle control point.

The change in values of editable configuration tags does not take effect immediately. For information on the conditions under which a change takes effect, refer to the detailed description of the relevant tag.

1.12.9 Command table technology object tag

1.12.9.1 Config.Command.Command[1 ... 32] tag

Legend

Data type	Data type of the tag
Start value	Start value of tag The start value can be overwritten by the configuration of the command table.
Access	Access to the tag in the user program:
	RW The tag can be read and written in the user program.
	R The tag can be read in the user program.
	- The tag cannot be used in the user program.
Effective	Specifies when a change in the tag takes effect.
HMI	The tag can be used in an HMI system.

<Axis name>.Config.Command.Command[x].Type				
Command type				
<ul style="list-style-type: none"> • 0 = "Empty" command • 2 = "Hold" command • 5 = "Relative positioning" command • 6 = "Absolute positioning" command • 7 = "Velocity setpoint" command • 151 = "Wait" command 				
Data type	Start value	Access	Effective	HMI
Int	0	RW	-	X

<Axis name>. Config.Command.Command[x].Position				
Command target position / travel path				
Data type	Start value	Access	Effective	HMI
Real		RW	-	X

<Axis name>. Config.Command.Command[x].Velocity				
Command velocity				
Data type	Start value	Access	Effective	HMI
Real	0.0	RW	-	X

<Axis name>. Config.Command.Command[x].Duration				
Command duration				
Data type	Start value	Access	Effective	HMI
Real	0.0	RW	-	X

<Axis name>. Config.Command.Command[x].BufferMode				
Value for command "Next step"				
<ul style="list-style-type: none"> • 0 = "Complete command" • 1 = "Blend movement" 				
Data type	Start value	Access	Effective	HMI
Int	0	RW	-	X

<Axis name>. Config.Command.Command[x].StepCode				
Command step code				
Data type	Start value	Access	Effective	HMI
Word	0	RW	-	X

1.12.10 Documentation for functions from previous versions

1.12.10.1 Configuration - Homing (technology object "Axis" V1.0)

Configure the parameters for active and passive homing in the "Homing" configuration window. The homing method is set using the "Mode" input parameter of the motion control instruction. Here, Mode = 2 means passive homing and Mode = 3 means active homing.

Homing switch input

Select the digital input for the homing switch from the drop-down list. The input must be interrupt-capable. The onboard CPU inputs and the inputs of an inserted signal board can be selected as inputs for the homing switch.

Note

The digital inputs are set to a filter time of 6.4 ms by default.

When the digital inputs are used as a homing switch, this can result in undesired decelerations and thus inaccuracies. Depending on the homing velocity and extent of the homing switch, the reference point may not be detected. The filter time can be set under "Input filter" in the device configuration of the digital inputs.

The specified filter time must be less than the duration of the input signal at the homing switch.

Permitting direction reversal after reaching the HW limit switch (active homing only)

Activate the check box to use the hardware limit switch as a reversing cam for the reference point approach. The hardware limit switches must be activated for direction reversal. If the CPU firmware V1.0 is used, both hardware limit switches must be configured. If CPU firmware as of V2.0 is used, only the hardware limit switches in the approach direction must be configured.

If the hardware limit switch is reached during active homing, the axis brakes at the configured deceleration (not with the emergency stop deceleration) and reverses direction. The homing switch is then sensed in reverse direction.

If the direction reversal is not active and the axis reaches the hardware limit switch during active homing, the reference point approach is aborted with an error and the axis is braked at the emergency stop deceleration.

NOTICE
<p>Use one of the following measures to ensure that the machine does not travel to a mechanical endstop in the event of a direction reversal:</p> <ul style="list-style-type: none"> • Keep the approach velocity low • Increase the configured acceleration/deceleration • Increase the distance between hardware limit switch and mechanical stop

Approach / homing direction (active and passive homing)

With the direction selection, you determine the "approach direction" used during active homing to search for the homing switch, as well as the homing direction. The homing direction specifies the travel direction the axis uses to approach the configured side of the homing switch to carry out the homing operation.

Refer to the table under "Homing switches" for the effect of the approach direction setting on passive homing.

Side of the homing switch (active and passive homing)

- **Active homing**

This is where you select whether the axis is homed on the low or high side of the homing switch.

Note

Depending on the start position of the axis and the configuration of the homing parameters, the reference point approach sequence can differ from the diagram in the configuration window.

- **Passive homing**

With passive homing, the traversing motions for purposes of homing must be implemented by the user via motion commands. The side of the homing switch on which homing occurs depends on the following factors:

- "Approach direction" configuration
- "Homing switch" configuration
- Current travel direction during passive homing

The table below presents details on the effect of factors:

Influencing factors:			Result:
Configuration Approach direction	Configuration Homing switch	Current travel direction	Homing on Homing switch
Positive	"Bottom side"	Positive direction	Top side
		Negative direction	Bottom side
Positive	"Top side"	Positive direction	Bottom side
		Negative direction	Top side
Negative	"Bottom side"	Positive direction	Bottom side
		Negative direction	Top side
Negative	"Top side"	Positive direction	Top side
		Negative direction	Bottom side

Velocity (active homing only)

In this field, specify the velocity at which the homing switch is to be searched for during the reference point approach.

Limits (independent of the selected unit of measurement):

- Start/stop velocity \leq approach velocity \leq maximum velocity

Homing velocity (active homing only)

In this field, specify the velocity at which the axis approaches the homing switch for homing.

Limits (independent of the selected unit of measurement):

- Start/stop speed \leq Homing velocity \leq Maximum velocity

Home position offset (active homing only)

If the desired reference position deviates from the position of the homing switch, the home position offset can be specified in this field.

If the value does not equal 0, the axis executes the following actions following homing at the homing switch:

1. Move the axis at the homing velocity by the value of the home position offset
2. Upon reaching the "home position offset", the axis is at the home position that was specified in input parameter "Position" of the "MC_Home" Motion Control instruction.

Limits (independent of the selected unit of measurement):

- $-1.0e12 \leq$ home position offset $\leq 1.0e12$

Home position

The position for which parameters are assigned in the Motion Control instruction "MC_Home" is used as the home position.

1.12.10.2 List of ErrorIDs and ErrorInfos (technology objects V1.0)

The following table lists all ErrorIDs and ErrorInfos that can be indicated in motion control instructions. In addition to the cause of the error, remedies for eliminating the error are also listed:

Operating error with stop of the axis

ErrorID	ErrorInfo	Description	Remedy
16#8000		Drive error, "Drive ready" failure	
	16#0001	-	Acknowledge error with instruction "MC_Reset"; provide drive signal; possibly restart command
16#8001		Low software limit switch has been tripped	
	16#000E	The position of the low software limit switch was reached with the currently configured deceleration	Acknowledge the error with instruction "MC_Reset"; use a motion command to move the axis in the positive direction out of the range of the software limit switch
	16#000F	The position of the low software limit switch was reached with the emergency stop deceleration	
	16#0010	The position of the low software limit switch was exceeded with the emergency stop deceleration	
16#8002		High software limit switch has been tripped	
	16#000E	The position of the high software limit switch was reached with the currently configured deceleration	Acknowledge the error with instruction "MC_Reset"; use a motion command to move the axis in the negative direction out of the range of the software limit switch
	16#000F	The position of the high software limit switch was reached with the emergency stop deceleration	
	16#0010	The position of the high software limit switch was exceeded with the emergency stop deceleration	
16#8003		Low hardware limit switch was approached	
	16#000E	The low hardware limit switch was approached. The axis was stopped with the emergency stop deceleration. (During an active reference point approach, the homing switch was not found)	Acknowledge the error for an enabled axis with instruction "MC_Reset"; use a motion command to move the axis in the positive direction out of the range of the hardware limit switch.
16#8004		High hardware limit switch was approached	
	16#000E	The high hardware limit switch was approached. The axis was stopped with the emergency stop deceleration. (During an active reference point approach, the homing switch was not found)	Acknowledge the error for an enabled axis with instruction "MC_Reset"; use a motion command to move the axis in the negative direction out of the range of the hardware limit switch.

ErrorID	ErrorInfo	Description	Remedy
16#8005		PTO/HSC are already being used by another axis	
	16#0001	-	<p>The axis was configured incorrectly: Correct the configuration of the PTO (Pulse Train Output) / HSC (High Speed Counter) and download it to the controller</p> <p>More than one axis is to run with one PTO: Another axis is using the PTO/HSC. If the current axis is to assume the control, the other axis must be disabled with "MC_Power" Enable = FALSE. (see also Using multiple axes with the same PTO (Page 105))</p>

Operating error without stop of the axis

ErrorID	ErrorInfo	Description	Remedy
16#8200		Axis is not enabled	
	16#0001	-	Enable the axis; restart the command
16#8201		Axis has already been enabled by another "MC_Power" instance	
	16#0001	-	Enable the axis with only one "MC_Power" instruction
16#8202		The maximum number of simultaneously active motion control jobs was exceeded (maximum of 200 jobs for all motion control technology objects)	
	16#0001	-	Reduce the number of simultaneously active jobs; restart the command A command is active if parameter "Busy" = TRUE in the motion control instruction.
16#8203		Axis is currently operated in "Manual control" (axis command table)	
	16#0001	-	Exit "Manual control"; restart the command
16#8204		Axis is not homed	
	16#0001	-	Home the axis with instruction "MC_Home"; restart the command
16#8205		The axis is currently controlled by the user program (the error is only displayed in the axis command table)	
	16#0001	-	Disable axis with instruction "MC_Power" and select "Manual control" again in the axis command table
16#8206		Technology object Axis not yet enabled	
	16#0001	-	Enable the axis with instruction "MC_Power" Enable = TRUE or enable the axis in the axis command table.
16#8207		Command rejected	
	16#0016	Active homing is running; another homing method cannot be started.	Wait for active homing to finish or abort the active homing with a motion command, for example, "MC_Halt". The other homing type can then be started.

Block parameter error

ErrorID	ErrorInfo	Description	Remedy
16#8400		Invalid value at parameter "Position" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the "position" value; restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#8401		Invalid value at parameter "Distance" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the "Distance" value; restart the command
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#8402		Invalid value at parameter "Velocity" of the Motion Control instruction	
	16#0002	Number format of value is invalid	Correct the "Velocity" value; restart the command
	16#0008	Velocity is greater than the maximum velocity	
	16#0009	Velocity is less than the start/stop velocity	
16#8403		Invalid value at parameter "Direction" of the Motion Control instruction	
	16#0011	Invalid selection value	Correct the selection value; restart the command
16#8404		Invalid value at parameter "Mode" of the Motion Control instruction	
	16#0011	Invalid selection value	Correct the selection value; restart the command
	16#0015	Active/passive homing is not configured	Correct the configuration and download it to the controller; enable the axis and restart the command
	16#0017	Axis reversal is activated at the HW limit switch, despite the fact that the hardware limit switches are disabled	<ul style="list-style-type: none"> • Activate the hardware limit switch using the tag <code><Axis>.Config.PositionLimits_HW.Active = TRUE</code>, restart the command • Correct the configuration and download it to the controller; enable the axis and restart the command
16#8405		Invalid value at parameter "StopMode" of the Motion Control instruction	
	16#0011	Invalid selection value	Correct the selection value; enable the axis again
16#8406		Simultaneous forward and backward jogging is not allowed	
	16#0001	-	Take steps to ensure that parameters "JogForward" and "JogBackward" do not have signal status TRUE simultaneously; restart the command.
16#8407		Switching the axis with Motion Control instruction "MC_Power" is only permitted after disabling the axis.	
	16#0001	-	Disable the active axis; it is then possible to switch to the other axis and enable it.

Configuration error

ErrorID	ErrorInfo	Description	Remedy
16#8600		Parameterization of pulse generator (PTO) is invalid	
	16#000B	Address is invalid	Correct the configuration of the PTO (Pulse Train Output) and download it to the controller
16#8601		Parameterization of the high-speed counter (HSC) is invalid	
	16#000B	Address is invalid	Correct the configuration of the HSC (High Speed Counter) and download it to the controller
16#8602		Invalid parameter assignment of "Enable output"	
	16#000D	Address is invalid	Correct the configuration and download it to the controller
16#8603		Invalid parameter assignment of "Ready input"	
	16#000D	Address is invalid	Correct the configuration and download it to the controller
16#8604		Invalid "Pulses per motor revolution" value	
	16#000A	The value is less than or equal to zero	Correct the configuration and download it to the controller
16#8605		Invalid "Load distance per motor revolution" value	
	16#0002	Number format of value is invalid	Correct the configuration and download it to the controller
	16#000A	The value is less than or equal to zero	
16#8606		Invalid "Start / stop velocity" value	
	16#0002	Number format of value is invalid	Correct the configuration and download it to the controller
	16#0003	Value exceeds the hardware limit	
	16#0004	Value is less than the hardware limit	
	16#0007	The start/stop velocity is greater than the maximum velocity	
16#8607		Value for "Maximum velocity" is invalid	
	16#0002	Number format of value is invalid	Correct the configuration and download it to the controller
	16#0003	Value exceeds the hardware limit	
	16#0004	Value is less than the hardware limit	
16#8608		Invalid "Acceleration" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; enable the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and restart the command, if necessary
	16#0003	Value exceeds the hardware limit	
	16#0004	Value is less than the hardware limit	

ErrorID	ErrorInfo	Description	Remedy
16#8609		Invalid "Deceleration" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; enable the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and restart the command, if necessary
	16#0003	Value exceeds the hardware limit	
	16#0004	Value is less than the hardware limit	
16#860A		Invalid "Emergency stop deceleration" value	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; enable the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and restart the command, if necessary
	16#0003	Value exceeds the hardware limit	
	16#0004	Value is less than the hardware limit	
16#860B		Value for position of the low SW limit switch is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; enable the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and restart the command, if necessary
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
	16#0007	The position value of the low SW limit switch is greater than that of the high SW limit switch	
16#860C		Value for position of the high SW limit switch is invalid	
	16#0002	Number format of value is invalid	<ul style="list-style-type: none"> Download error-free configuration to the controller; enable the axis again with instruction "MC_Power" Correct the incorrect value online; acknowledge error with instruction "MC_Reset" and restart the command, if necessary
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	
16#860D		Invalid address of the low HW limit switch	
	16#000C	Address of falling edge is invalid	Correct the configuration and download it to the controller
	16#000D	Address of rising edge is invalid	
16#860E		Invalid address of the high HW limit switch	
	16#000C	Address of falling edge is invalid	Correct the configuration and download it to the controller
	16#000D	Address of rising edge is invalid	
16#860F		Invalid "home position offset" value	
	16#0002	Number format of value is invalid	Correct the configuration and download it to the controller
	16#0005	The value is outside the number range (greater than $1e^{12}$)	
	16#0006	The value is outside the number range (less than $-1e^{12}$)	

ErrorID	ErrorInfo	Description	Remedy
16#8610		Invalid "approach velocity" value	
	16#0002	Number format of value is invalid	Correct the configuration and download it to the controller
	16#0008	Velocity is greater than the maximum velocity	
	16#0009	Velocity is less than the start/stop velocity	
16#8611		Invalid "Homing velocity" value	
	16#0002	Number format of value is invalid	Correct the configuration and download it to the controller
	16#0008	Velocity is greater than the maximum velocity	
	16#0009	Velocity is less than the start/stop velocity	
16#8612		Invalid address of the homing switch	
	16#000C	Address of falling edge is invalid	Correct the configuration and download it to the controller
	16#000D	Address of rising edge is invalid	
16#8613		During active homing, direction reversal at the hardware limit switch is activated although the hardware limit switches are not configured	
	16#0001	-	Correct the configuration and download it to the controller

Internal errors

ErrorID	ErrorInfo	Description	Remedy
16#8FFF		Internal error	
	16#F0**	-	POWER OFF and POWER ON the CPU If this does not work, contact Customer Support. Have the following information ready: <ul style="list-style-type: none"> • ErrorID • ErrorInfo • Diagnostic buffer entries

See also

Using multiple axes with the same PTO (Page 105)

S7-1200 Motion Control

2.1 MC_Power

2.1.1 MC_Power: Enable, disable axis

Description

The Motion Control instruction "MC_Power" releases or locks an axis.

Requirements

- The technology object "Axis" has been configured correctly.
- There is no pending enable-inhibiting error.

Override response

Execution of "MC_Power" cannot be aborted by a motion control command.

Disabling the axis (input parameter "Enable" = FALSE) aborts all motion control jobs for the associated technology object in accordance with the selected "StopMode".

Parameter

Parameter	Declaration	Data type	Default value	Description	
Axis	INPUT	TO_Axis_1	-	Axis technology object	
Enable	INPUT	BOOL	FALSE	TRUE	Motion Control attempts to enable the axis.
				FALSE	All active jobs are aborted according to the configured "StopMode" and the axis is stopped.
StopMode	INPUT	INT	0	0	Emergency stop If a request to disable the axis is pending, the axis brakes at the configured emergency stop deceleration. The axis is disabled after reaching standstill.
				1	Immediate stop If a request to disable the axis is pending, this axis is disabled without deceleration. The pulse output is stopped immediately.
				2	Emergency stop with jerk control If a request to disable the axis is pending, the axis brakes at the configured emergency stop deceleration. If the jerk control is activated, the configured jerk is taken into account. The axis is disabled after reaching standstill.
Status	OUTPUT	BOOL	FALSE	Status of axis enable	
				FALSE	The axis is disabled. The axis does not execute motion control jobs and does not accept any new jobs (exception: MC_Reset command). The axis is not homed. Upon disabling, the status does not change to FALSE until the axis reaches a standstill.
				TRUE	The axis is enabled. The axis is ready to execute motion control jobs. Upon axis enabling, the status does not change to TRUE until the signal "Drive ready" is pending. If the "Drive ready" drive interface was not configured in the axis configuration, the status changes to TRUE immediately.
Busy	OUTPUT	BOOL	FALSE	TRUE	MC Power is active
Error	OUTPUT	BOOL	FALSE	TRUE	An error occurred in motion control instruction "MC_Power" or in the associated technology object. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"	

NOTICE

If the axis is switched off due to an error, it will be enabled again automatically after the error has been eliminated and acknowledged. This requires that input parameter "Enable" has retained the value TRUE during this process.

Enabling an axis with configured drive interface

To enable the axis, follow these steps:

1. Check the requirements indicated above.
2. Initialize input parameter "StopMode" with the desired value.
Set input parameter "Enable" to TRUE.

The enable output for "Drive enabled" changes to TRUE to enable the power to the drive. The CPU waits for the "Drive ready" signal of the drive.

When the "Drive ready" signal is available at the configured ready input of the CPU, the axis becomes enabled. Output parameter "Status" and tag of technology object <Axis name>.StatusBits.Enable indicate the value TRUE.

Enabling an axis without configured drive interface

To enable the axis, follow these steps:

1. Check the requirements indicated above.
2. Initialize input parameter "StopMode" with the desired value. Set input parameter "Enable" to TRUE. The axis is enabled. Output parameter "Status" and tag of technology object <Axis name>.StatusBits.Enable indicate the value TRUE.

Disabling an axis

To disable an axis, you can follow the steps described below:

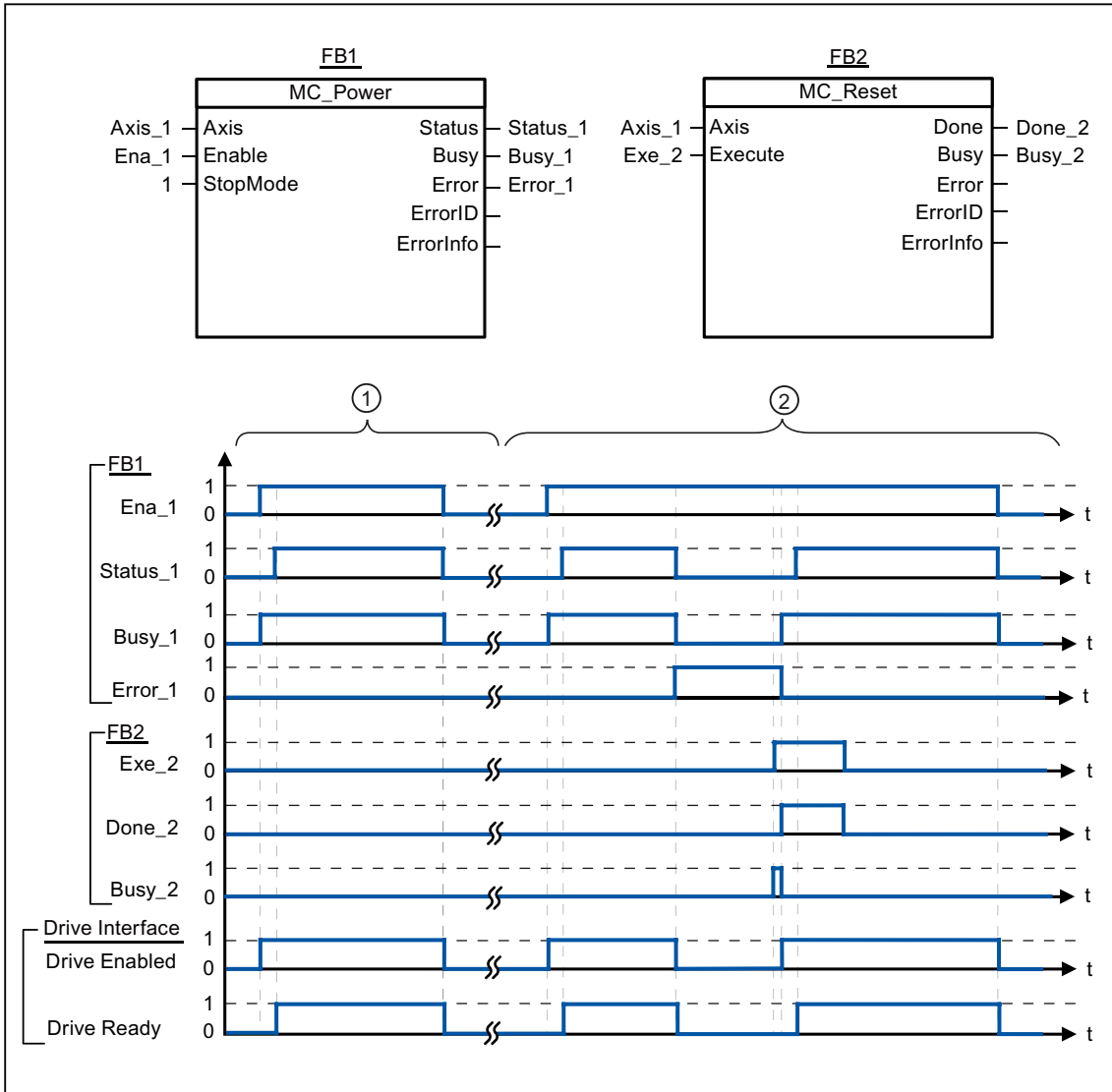
1. Bring the axis to a standstill.
You can identify when the axis is at a standstill in the tag of the technology object <Axis name>.StatusBits.StandStill.
2. Set input parameter "Enable" to FALSE after standstill is reached.
3. If output parameters "Busy" and "Status" and tag of technology object <Axis name>.StatusBits.Enable indicate the value FALSE, disabling of the axis is complete.

See also

- List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)
- MC_Power: Function chart (Page 165)
- MC_Reset: Acknowledge error (Page 166)
- MC_Home: Home axes, set home position (Page 168)
- MC_Halt: Halt axis (Page 172)
- MC_MoveAbsolute: Absolute positioning of axes (Page 176)
- MC_MoveRelative: Relative positioning of axes (Page 180)
- MC_MoveVelocity: Move axes at preset rotational speed (Page 184)
- MC_MoveJog: Move axes in jogging mode (Page 189)
- MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)
- MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0) (Page 192)

2.1.2 MC_Power: Function chart

Function chart



- | | |
|---|--|
| ① | An axis is enabled and then disabled again. When the drive has signaled "Drive ready" back to the CPU, the successful enable can be read out via "Status_1". |
| ② | Following an axis enable, an error has occurred that caused the axis to be disabled. The error is eliminated and acknowledged with "MC_Reset". The axis is then enabled again. |

See also

MC_Power: Enable, disable axis (Page 161)

2.2 MC_Reset

2.2.1 MC_Reset: Acknowledge error

Description

Motion Control instruction "MC_Reset" can be used to acknowledge "Operating error with axis stop" and "Configuration error". The errors that require acknowledgement can be found in the "List of ErrorIDs and ErrorInfos" under "Remedy".

From version V3.0, the axis configuration can be downloaded to the work memory in RUN operating mode.

Requirements

- The technology object "Axis" has been configured correctly.
- The cause of a pending configuration error requiring acknowledgement has been eliminated (for example, acceleration in "Axis" technology object has been changed to a valid value).

Override response

The MC_Reset command cannot be aborted by any other motion control command.

The new MC_Reset command does not abort any other active motion control jobs.

Parameter

Parameter	Declaration	Data type	Default value	Description	
Axis	INPUT	TO_Axis_1	-	Axis technology object	
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge	
Restart	INPUT	BOOL	FALSE	(From version V3.0)	
				TRUE	Download the axis configuration from the load memory to the work memory. The command can only be executed when the axis is disabled. Please refer to the notes on Download to the CPU (Page 74).
				FALSE	Acknowledges pending errors
Done	OUTPUT	BOOL	FALSE	TRUE Error has been acknowledged.	
Busy	OUTPUT	BOOL	FALSE	TRUE The command is being executed.	
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".	
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"	

Acknowledging an error requiring acknowledgement with MC_Reset

To acknowledge an error, follow these steps:

1. Check the requirements indicated above.
2. Start the acknowledgement of the error with a rising edge at input parameter "Execute".
3. If output parameter "Done" indicates the value TRUE and tag of technology object <Axis name>.StatusBits.Error the value FALSE, the error has been acknowledged.

See also

Download to CPU (Page 74)

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

MC_Power: Enable, disable axis (Page 161)

MC_Home: Home axes, set home position (Page 168)

MC_Halt: Halt axis (Page 172)

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

MC_MoveRelative: Relative positioning of axes (Page 180)

MC_MoveVelocity: Move axes at preset rotational speed (Page 184)

MC_MoveJog: Move axes in jogging mode (Page 189)

MC_CommandTable: Run axis jobs as movement sequence
(as of technology object "Axis" V2.0) (Page 192)

MC_ChangeDynamic: Changing the dynamic settings for the axis
(as of technology object "Axis" V2.0) (Page 194)

2.3 MC_Home

2.3.1 MC_Home: Home axes, set home position

Description

Use the motion control instruction "MC_Home" to match the axis coordinates to the real, physical drive position. Homing is required for absolute positioning of the axis.

The following types of homing can be executed:

- Active homing (Mode = 3)

The homing procedure is executed automatically.

- Passive homing (Mode = 2)

During passive homing, the motion control instruction "MC_Home" does not carry out any homing motion. The traversing motion required for this step must be implemented by the user via other motion control instructions. When the homing switch is detected, the axis is homed.

- Direct homing absolute (Mode = 0)

The current axis position is set to the value of parameter "Position".

- Direct homing relative (Mode = 1)

The current axis position is offset by the value of parameter "Position".

Requirements

- The technology object "Axis" has been configured correctly.
- The axis is enabled.
- No MC_CommandTable command may be active upon start with Mode = 0, 1 or 2.

Override response

The override response is dependent on the selected mode:

Mode = 0, 1

The MC_Home command cannot be aborted by any other motion control command.

The MC_Home command does not abort any active motion control jobs. Position-related motion jobs are resumed after homing according to the new homing position (value at input parameter: "Position").

Mode = 2

The MC_Home command can be aborted by the following motion control jobs:

- MC_Home command Mode = 2, 3

The new MC_Home command aborts the following active motion control command.

- MC_Home command Mode = 2

Position-related motion jobs are resumed after homing according to the new homing position (value at input parameter: "Position").

Mode = 3

The MC_Home command can be aborted by the following motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

The new MC_Home command aborts the following active motion control jobs:

- MC_Home command Mode = 2, 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

Parameter

Parameter	Declaration	Data type	Default value	Description	
Axis	INPUT	TO_Axis_1	-	Axis technology object	
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge	
Position	INPUT	REAL	0.0	<ul style="list-style-type: none"> Mode = 0, 2, and 3 Absolute position of axis after completion of the homing operation Mode = 1 Correction value for the current axis position Limit values: $-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$	
Mode	INPUT	INT	0	Homing mode	
				0	Direct homing absolute New axis position is the position value of parameter "Position".
				1	Direct homing relative New axis position is the current axis position + position value of parameter "Position".
				2	Passive homing Homing according to the axis configuration. Following homing, the value of parameter "Position" is set as the new axis position.
3	Active homing Reference point approach in accordance with the axis configuration. Following homing, the value of parameter "Position" is set as the new axis position.				
Done	OUTPUT	BOOL	FALSE	TRUE	Command completed
Busy	OUTPUT	BOOL	FALSE	TRUE	The command is being executed.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	During execution the command was aborted by another command.
Error	OUTPUT	BOOL	FALSE	TRUE	An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000		Error ID for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000		Error info ID for parameter "ErrorID"

Note

Axis homing is lost under the following conditions:

- Disabling of axis by motion control instruction "MC_Power"
 - Changeover between automatic mode and manual control
 - Upon start of active homing. After successful completion of the homing operation, axis homing is again available.
 - After POWER OFF -> POWER ON of the CPU
 - After CPU restart (RUN-STOP -> STOP-RUN)
-

Homing an axis

To home the axis, follow these steps:

1. Check the requirements indicated above.
2. Initialize the necessary input parameters with values, and start the homing operation with a rising edge at input parameter "Execute"
3. If output parameter "Done" and technology object tag <Axis name>.StatusBits.HomingDone indicate the value TRUE, homing is complete.

See also

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

MC_Power: Enable, disable axis (Page 161)

MC_Reset: Acknowledge error (Page 166)

MC_Halt: Halt axis (Page 172)

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

MC_MoveRelative: Relative positioning of axes (Page 180)

MC_MoveVelocity: Move axes at preset rotational speed (Page 184)

MC_MoveJog: Move axes in jogging mode (Page 189)

MC_CommandTable: Run axis jobs as movement sequence
(as of technology object "Axis" V2.0) (Page 192)

MC_ChangeDynamic: Changing the dynamic settings for the axis
(as of technology object "Axis" V2.0) (Page 194)

2.4 MC_Halt

2.4.1 MC_Halt: Halt axis

Description

The "MC_Halt" motion control instruction stops all movements and brings the axis to a standstill with the configured deceleration. The standstill position is not defined.

Requirements

- The technology object "Axis" has been configured correctly.
- The axis is enabled.

Override response

The MC_Halt command can be aborted by the following motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

The new MC_Halt command aborts the following active motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

Parameter

Parameter	Declaration	Data type	Default value	Description
Axis	INPUT	TO_Axis_1	-	Axis technology object
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge
Done	OUTPUT	BOOL	FALSE	TRUE Zero velocity reached
Busy	OUTPUT	BOOL	FALSE	TRUE The command is being executed.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE During execution the command was aborted by another command.
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"

See also

MC_Halt: Function chart (Page 174)

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

MC_Power: Enable, disable axis (Page 161)

MC_Reset: Acknowledge error (Page 166)

MC_Home: Home axes, set home position (Page 168)

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

MC_MoveRelative: Relative positioning of axes (Page 180)

MC_MoveVelocity: Move axes at preset rotational speed (Page 184)

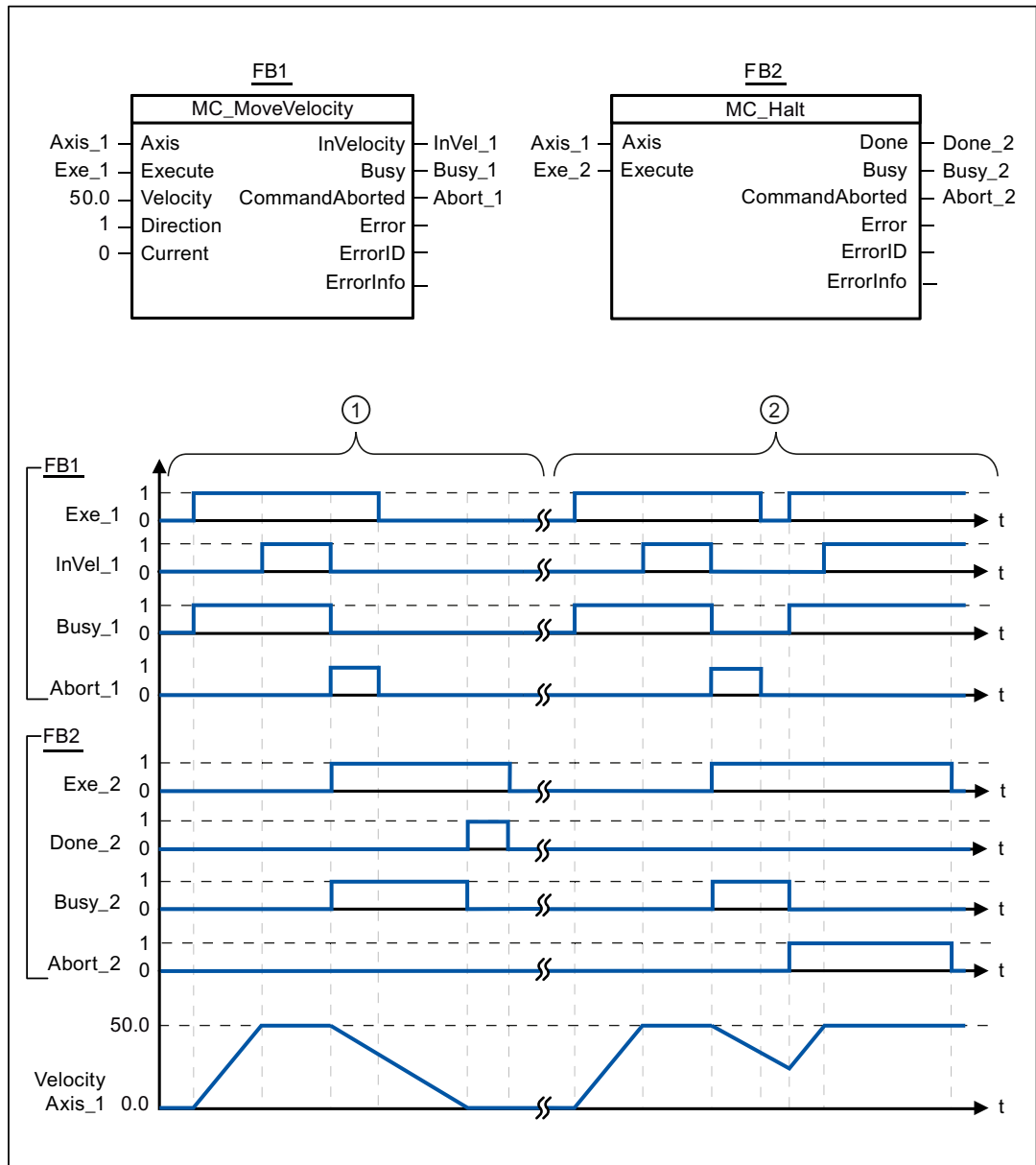
MC_MoveJog: Move axes in jogging mode (Page 189)

MC_CommandTable: Run axis jobs as movement sequence
(as of technology object "Axis" V2.0) (Page 192)

MC_ChangeDynamic: Changing the dynamic settings for the axis
(as of technology object "Axis" V2.0) (Page 194)

2.4.2 MC_Halt: Function chart

Function chart



The following values were configured in the configuration window **Dynamics > General**:

- Acceleration: 10.0
- Deceleration: 5.0

①	The axis is braked by an MC_Halt job until it comes to a standstill. The axis standstill is signaled via "Done_2".
②	While an MC_Halt job is braking the axis, this job is aborted by another motion job. The abort is signaled via "Abort_2".

See also

MC_Halt: Halt axis (Page 172)

2.5 MC_MoveAbsolute

2.5.1 MC_MoveAbsolute: Absolute positioning of axes

Description

The "MC_MoveAbsolute" motion control instruction starts an axis positioning motion to move it to an absolute position.

Requirements

- The technology object "Axis" has been configured correctly.
- The axis is enabled.
- The axis is homed.

Override response

The MC_MoveAbsolute command can be aborted by the following motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

The new MC_MoveAbsolute command aborts the following active motion control jobs:

- MC_Home command Mode = 3
- MC_Halt-Auftrag
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

Parameter

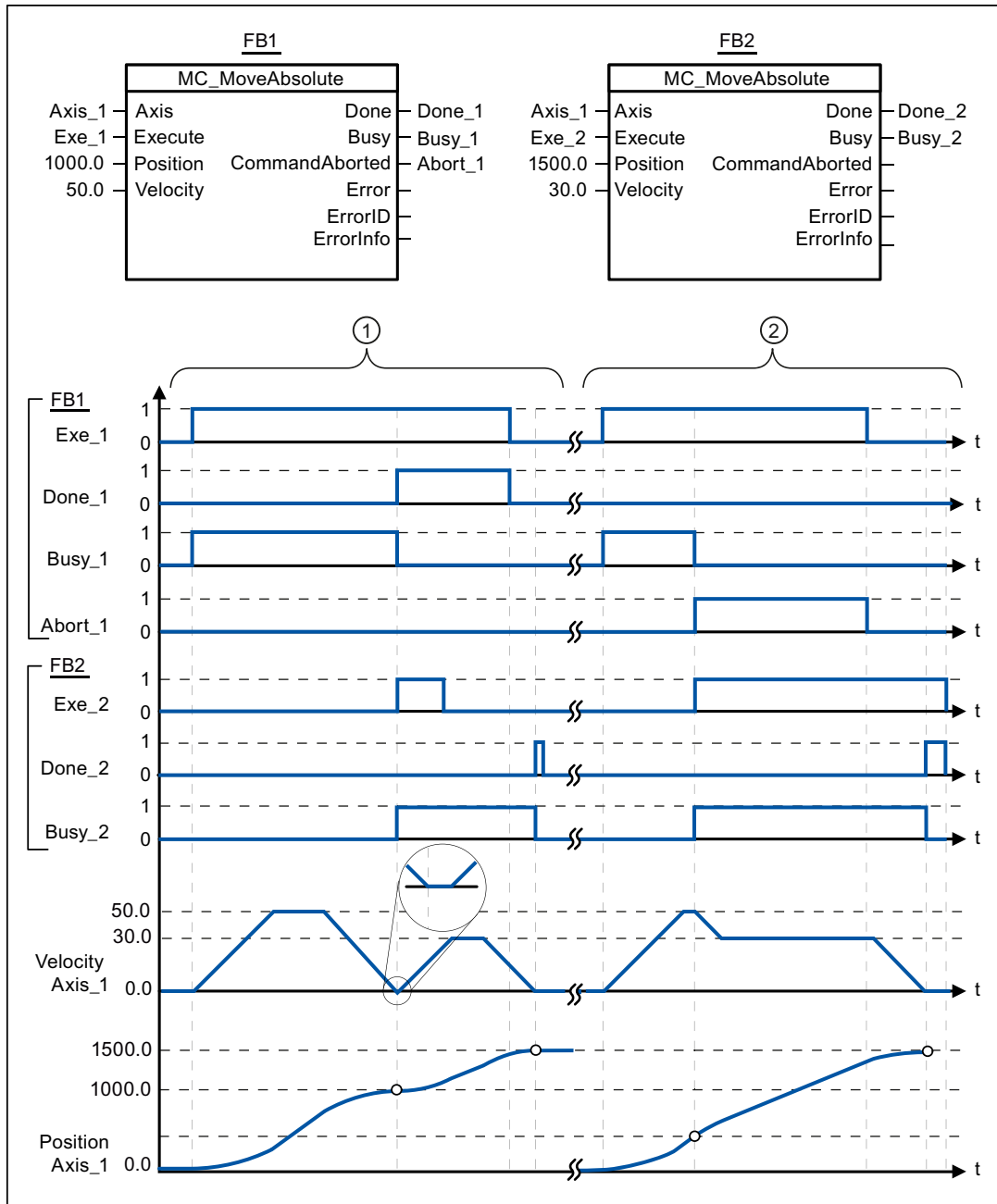
Parameter	Declaration	Data type	Default value	Description
Axis	INPUT	TO_Axis_1	-	Axis technology object
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge
Position	INPUT	REAL	0.0	Absolute target position Limit values: $-1.0e^{12} \leq \text{Position} \leq 1.0e^{12}$
Velocity	INPUT	REAL	10.0	Velocity of axis This velocity is not always reached on account of the configured acceleration and deceleration and the target position to be approached. Limit values: $\text{Start/stop velocity} \leq \text{Velocity} \leq \text{maximum velocity}$
Done	OUTPUT	BOOL	FALSE	TRUE Absolute target position reached
Busy	OUTPUT	BOOL	FALSE	TRUE The command is being executed.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE During execution the command was aborted by another command.
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"

See also

- MC_MoveAbsolute: Function chart (Page 178)
- List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)
- MC_Power: Enable, disable axis (Page 161)
- MC_Reset: Acknowledge error (Page 166)
- MC_Home: Home axes, set home position (Page 168)
- MC_Halt: Halt axis (Page 172)
- MC_MoveRelative: Relative positioning of axes (Page 180)
- MC_MoveVelocity: Move axes at preset rotational speed (Page 184)
- MC_MoveJog: Move axes in jogging mode (Page 189)
- MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0) (Page 192)
- MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)

2.5.2 MC_MoveAbsolute: Function chart

Function chart



The following values were configured in the configuration window **Dynamics > General**:

- Acceleration: 10.0
- Deceleration: 10.0

①	An axis is moved to absolute position 1000.0 with a MC_MoveAbsolute job. When the axis reaches the target position, this is signaled via "Done_1". When "Done_1" = TRUE, another MC_MoveAbsolute job, with target position 1500.0, is started. Because of the response times (e.g., cycle time of user program, etc.), the axis comes to a standstill briefly (see zoomed-in detail). When the axis reaches the new target position, this is signaled via "Done_2".
②	An active MC_MoveAbsolute job is aborted by another MC_MoveAbsolute job. The abort is signaled via "Abort_1". The axis is then moved at the new velocity to the new target position 1500.0. When the new target position is reached, this is signaled via "Done_2".

See also

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

2.6 MC_MoveRelative

2.6.1 MC_MoveRelative: Relative positioning of axes

Description

The "MC_MoveRelative" motion control instruction starts a positioning motion relative to the start position.

Requirements

- The technology object "Axis" has been configured correctly.
- The axis is enabled.

Override response

The MC_MoveRelative command can be aborted by the following motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

The new MC_MoveRelative command aborts the following active motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

Parameter

Parameter	Declaration	Data type	Default value	Description
Axis	INPUT	TO_Axis_1	-	Axis technology object
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge
Distance	INPUT	REAL	0.0	Travel distance for the positioning operation Limit values: $-1.0e^{12} \leq \text{Distance} \leq 1.0e^{12}$
Velocity	INPUT	REAL	10.0	Velocity of axis This velocity is not always reached on account of the configured acceleration and deceleration and the distance to be traveled. Limit values: $\text{Start/stop velocity} \leq \text{Velocity} \leq \text{maximum velocity}$
Done	OUTPUT	BOOL	FALSE	TRUE Target position reached
Busy	OUTPUT	BOOL	FALSE	TRUE The command is being executed.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE During execution the command was aborted by another command.
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"

See also

MC_MoveRelative: Function chart (Page 182)

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

MC_Power: Enable, disable axis (Page 161)

MC_Reset: Acknowledge error (Page 166)

MC_Home: Home axes, set home position (Page 168)

MC_Halt: Halt axis (Page 172)

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

MC_MoveVelocity: Move axes at preset rotational speed (Page 184)

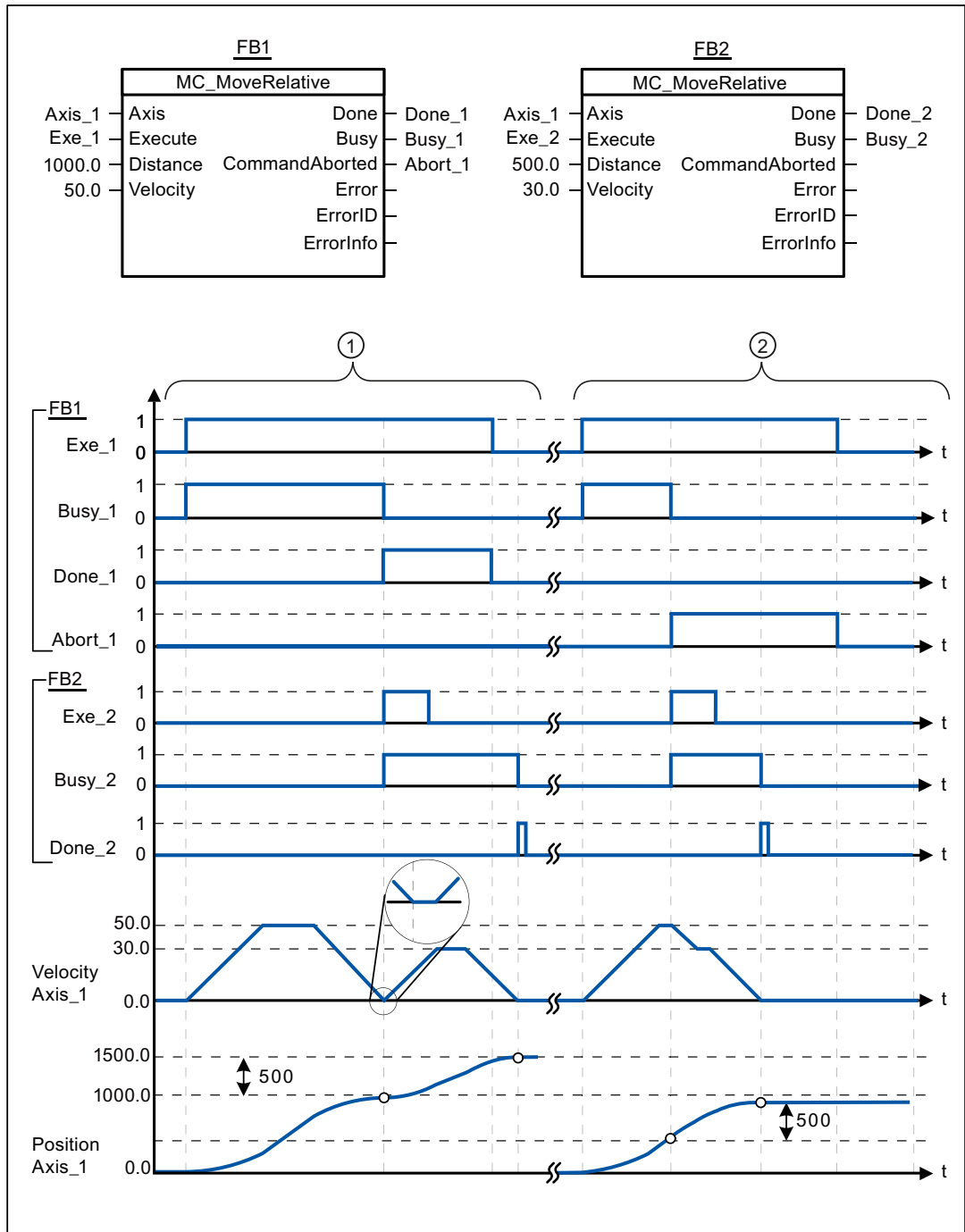
MC_MoveJog: Move axes in jogging mode (Page 189)

MC_CommandTable: Run axis jobs as movement sequence
(as of technology object "Axis" V2.0) (Page 192)

MC_ChangeDynamic: Changing the dynamic settings for the axis
(as of technology object "Axis" V2.0) (Page 194)

2.6.2 MC_MoveRelative: Function chart

Function chart



The following values were configured in the configuration window **Dynamics > General**:

- Acceleration: 10.0
- Deceleration: 10.0

①	The axis is moved by an MC_MoveRelative job by the distance ("Distance") 1000.0. When the axis reaches the target position, this is signaled via "Done_1". When "Done_1" = TRUE, another MC_MoveRelative job, with travel distance 500.0, is started. Because of the response times (e.g., cycle time of user program, etc.), the axis comes to a standstill briefly (see zoomed-in detail). When the axis reaches the new target position, this is signaled via "Done_2".
②	An active MC_MoveRelative job is aborted by another MC_MoveRelative job. The abort is signaled via "Abort_1". The axis is then moved at the new velocity by the new distance ("Distance") 500.0. When the new target position is reached, this is signaled via "Done_2".

See also

MC_MoveRelative: Relative positioning of axes (Page 180)

2.7 MC_MoveVelocity

2.7.1 MC_MoveVelocity: Move axes at preset rotational speed

Description

Motion control instruction "MC_MoveVelocity" moves the axis constantly at the specified velocity.

Requirements

- The technology object "Axis" has been configured correctly.
- The axis is enabled.

Override response

MC_MoveVelocity can be aborted by the following motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

The new MC_MoveVelocity command aborts the following active motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

Parameter

Parameter	Declaration	Data type	Default value	Description	
Axis	INPUT	TO_Axis_1	-	Axis technology object	
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge	
Velocity	INPUT	REAL	10.0	Velocity specification for axis motion Limit values: Start/stop velocity \leq Velocity \leq maximum velocity (Velocity = 0.0 is permitted)	
Direction	INPUT	INT	0	Direction specification	
				0	Direction of rotation corresponds to the sign of the value in parameter "Velocity"
				1	Positive direction of rotation (The sign of the value in parameter "Velocity" is ignored)
2	Negative direction of rotation (The sign of the value in parameter "Velocity" is ignored)				
Current	INPUT	BOOL	FALSE	Maintain current velocity	
				FALSE	"Maintain current velocity" is deactivated. The values of parameters "Velocity" and "Direction" are used.
				TRUE	"Maintain current velocity" is activated. The values in parameters "Velocity" and "Direction" are not taken into account. When the axis resumes motion at the current velocity, the "InVelocity" parameter returns the value TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	<ul style="list-style-type: none"> "Current" = FALSE: The velocity specified in parameter "Velocity" was reached. "Current" = TRUE: The axis travels at the current velocity at the start time.
Busy	OUTPUT	BOOL	FALSE	TRUE	The command is being executed.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	During execution the command was aborted by another command.
Error	OUTPUT	BOOL	FALSE	TRUE	An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000		Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000		Error info ID (Page 120) for parameter "ErrorID"

Behavior with zero set velocity (Velocity = 0.0)

An MC_MoveVelocity command with "Velocity" = 0.0 (such as an MC_Halt command) aborts active motion jobs and stops the axis with the configured deceleration.

When the axis comes to a standstill, output parameter "InVelocity" indicates TRUE for at least one program cycle.

"Busy" indicates the value TRUE during the deceleration operation and changes to FALSE together with "InVelocity". If parameter "Execute" = TRUE is set, "InVelocity" and "Busy" are latched.

When the MC_MoveVelocity command is started, status bit "SpeedCommand" is set in the technology object. Status bit "ConstantVelocity" is set upon axis standstill. Both bits are adapted to the new situation when a new motion command is started.

See also

MC_MoveVelocity: Function chart (Page 187)

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

MC_Power: Enable, disable axis (Page 161)

MC_Reset: Acknowledge error (Page 166)

MC_Home: Home axes, set home position (Page 168)

MC_Halt: Halt axis (Page 172)

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

MC_MoveRelative: Relative positioning of axes (Page 180)

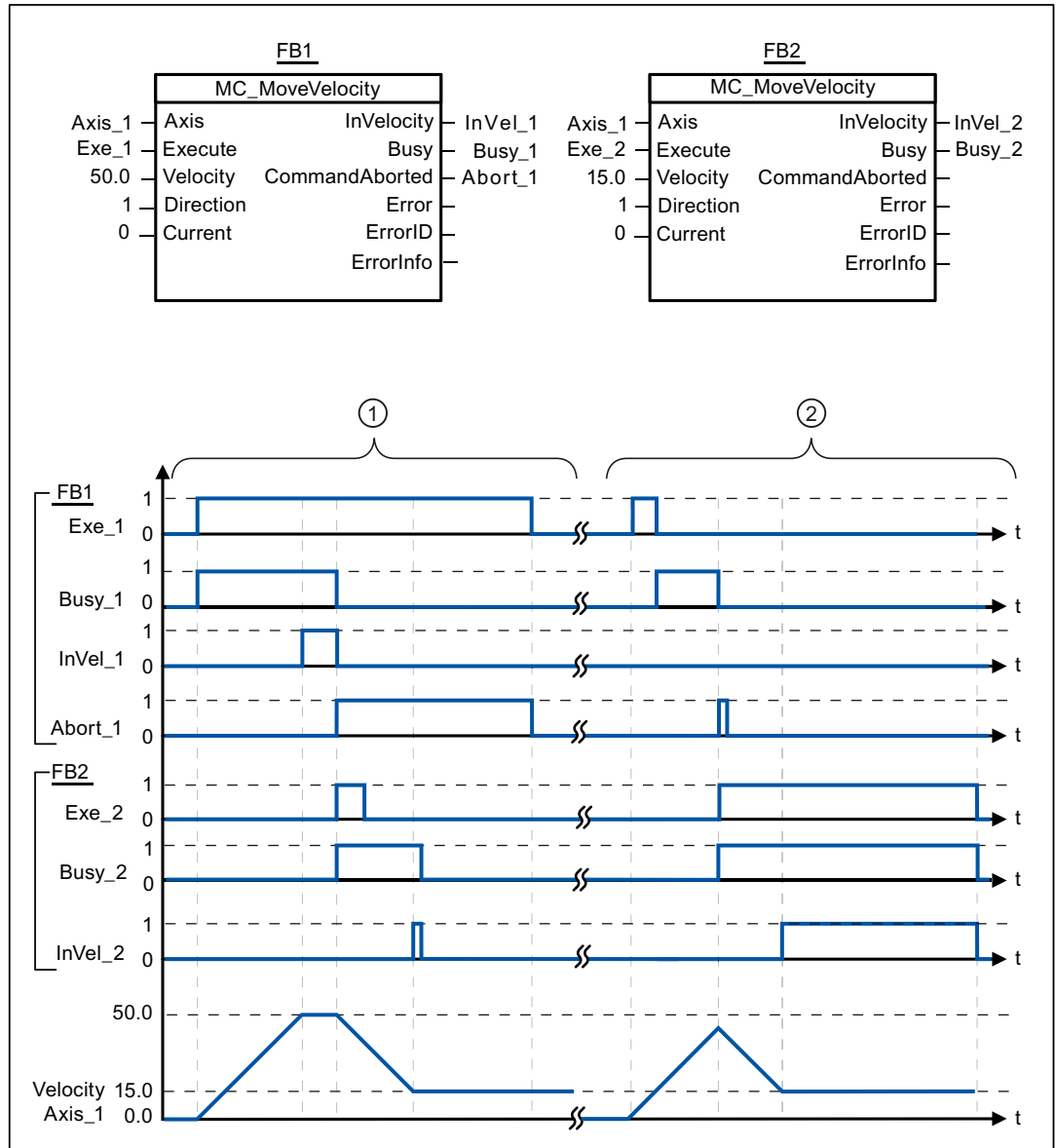
MC_MoveJog: Move axes in jogging mode (Page 189)

MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0) (Page 192)

MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)

2.7.2 MC_MoveVelocity: Function chart

Function chart



The following values were configured in the configuration window **Dynamics > General**:

- Acceleration: 10.0
- Deceleration: 10.0

①	An active MC_MoveVelocity job signals via "InVel_1" that its target velocity has been reached. It is then aborted by another MC_MoveVelocity job. The abort is signaled via "Abort_1". When the new target velocity 15.0 is reached, this is signaled via "InVel_2". The axis then continues moving at the new constant velocity.
②	An active MC_MoveVelocity job is aborted by another MC_MoveVelocity job prior to reaching its target velocity. The abort is signaled via "Abort_1". When the new target velocity 15.0 is reached, this is signaled via "InVel_2". The axis then continues moving at the new constant velocity.

See also

MC_MoveVelocity: Move axes at preset rotational speed (Page 184)

2.8 MC_MoveJog

2.8.1 MC_MoveJog: Move axes in jogging mode

Description

Motion control instruction "MC_MoveJog" moves the axis constantly at the specified velocity in jog mode. You use this motion control instruction, for example, for testing and commissioning purposes.

Requirements

- The technology object "Axis" has been configured correctly.
- The axis is enabled.

Override response

The MC_MoveJog command can be aborted by the following motion control jobs:

- MC_Home command Mode = 3
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

The new MC_MoveJog command aborts the following active motion control jobs:

- MC_Home command (Mode = 3)
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command

Parameter

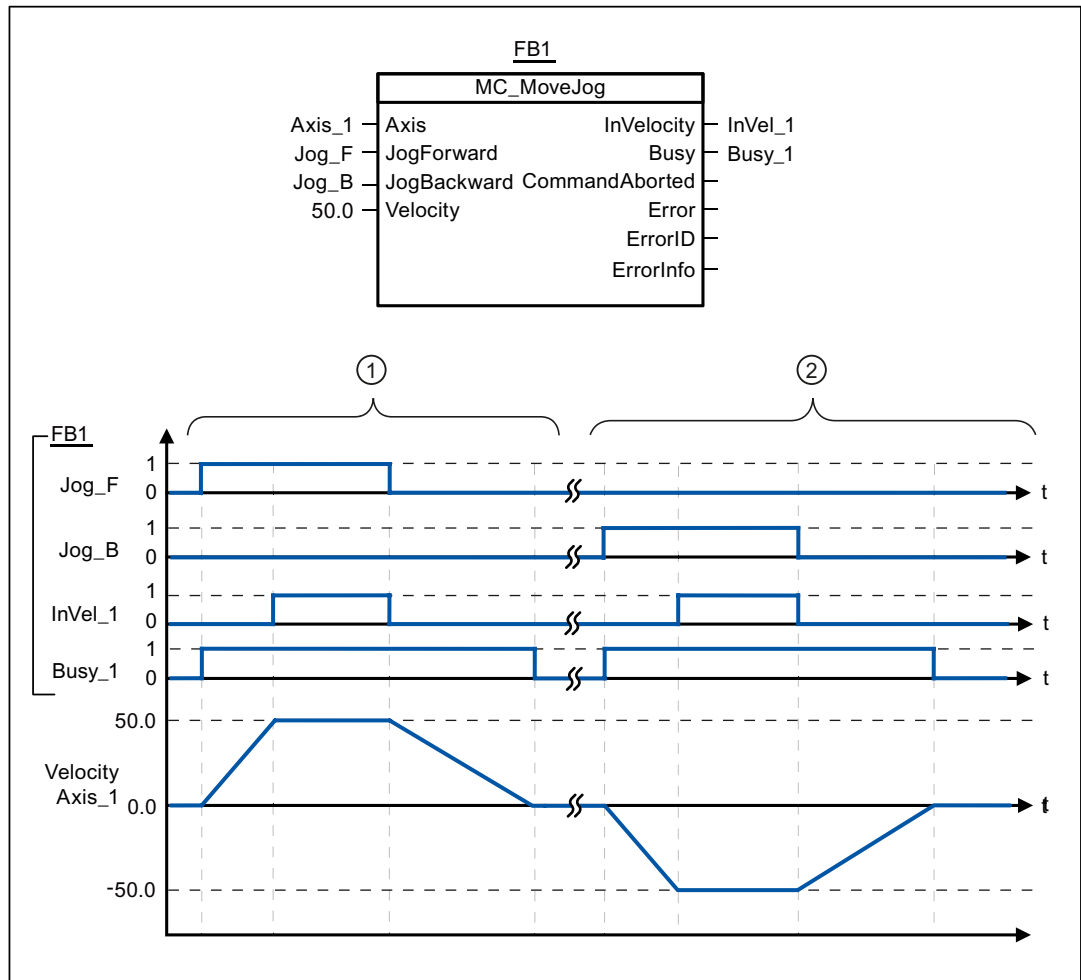
Parameter	Declaration	Data type	Default value	Description
Axis	INPUT	TO_Axis_1	-	Axis technology object
JogForward	INPUT	BOOL	FALSE	As long as the parameter is TRUE, the axis moves in the positive direction at the velocity specified in parameter "Velocity".
JogBackward	INPUT	BOOL	FALSE	As long as the parameter is TRUE, the axis moves in the negative direction at the velocity specified in parameter "Velocity".
If both parameters are simultaneously TRUE, the axis stops with the configured deceleration. An error is indicated in parameters "Error", "ErrorID", and "ErrorInfo".				
Velocity	INPUT	REAL	10.0	Preset velocity for jog mode
				Limit values, instruction version V1.0: Start/stop velocity ≤ Velocity ≤ maximum velocity
				Limits, instruction version V2.0: Start/stop velocity ≤ velocity ≤ maximum velocity
InVelocity	OUTPUT	BOOL	FALSE	TRUE The velocity specified in parameter "Velocity" was reached.
Busy	OUTPUT	BOOL	FALSE	TRUE The command is being executed.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE During execution the command was aborted by another command.
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"

See also

- MC_MoveJog: Function chart (Page 191)
- List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)
- MC_Power: Enable, disable axis (Page 161)
- MC_Reset: Acknowledge error (Page 166)
- MC_Home: Home axes, set home position (Page 168)
- MC_Halt: Halt axis (Page 172)
- MC_MoveAbsolute: Absolute positioning of axes (Page 176)
- MC_MoveRelative: Relative positioning of axes (Page 180)
- MC_MoveVelocity: Move axes at preset rotational speed (Page 184)
- MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0) (Page 192)
- MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)

2.8.2 MC_MoveJog: Function chart

Function chart



The following values were configured in the configuration window **Dynamics > General**:

- Acceleration: 10.0
- Deceleration: 5.0

①	The axis is moved in the positive direction in jog mode via "Jog_F". When the target velocity 50.0 is reached, this is signaled via "InVelo_1". The axis brakes to a standstill again after Jog_F is reset.
②	The axis is moved in the negative direction in jog mode via "Jog_B". When the target velocity 50.0 is reached, this is signaled via "InVelo_1". The axis brakes to a standstill again after Jog_B is reset.

See also

MC_MoveJog: Move axes in jogging mode (Page 189)

2.9 MC_CommandTable

2.9.1 MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0)

Description

The Motion Control instruction "MC_CommandTable" combines multiple individual axis control jobs in one movement sequence.

Requirements

- The technology object "Axis" has been added to Version V2.0 and correctly configured.
- The technology object "Command table" has been added and correctly configured.
- The axis is enabled.

Override response

The MC_CommandTable command can be aborted by the following motion control jobs:

- MC_Home command (Mode = 3)
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command
- MC_CommandTable command

The new MC_CommandTable command aborts the following active motion control jobs:

- MC_Home command (Mode = 3)
- MC_Halt command
- MC_MoveAbsolute command
- MC_MoveRelative command
- MC_MoveVelocity command
- MC_MoveJog command
- MC_CommandTable command

The active motion control command is cancelled by the start of the first "Positioning Relative", "Positioning Absolute", "Velocity set point" or "Halt" command.

Parameter

Parameter	Declaration	Data type	Default value	Description
Axis	INPUT	TO_Axis_1	-	Axis technology object
Command Table	INPUT	TO_Command Table_1	-	Command table technology object
Execute	INPUT	BOOL	FALSE	Command table start with positive edge
StartStep	INPUT	INT	1	Defines the step at which the execution of the command table should begin Limit values: $1 \leq \text{StartStep} \leq \text{EndStep}$
EndStep	INPUT	INT	32	Defines the step up to which the execution of command table should take place Limit values: $\text{StartStep} \leq \text{EndStep} \leq 32$
Done	OUTPUT	BOOL	FALSE	TRUE Command table has been successfully executed
Busy	OUTPUT	BOOL	FALSE	TRUE The command table is being executed.
Command Aborted	OUTPUT	BOOL	FALSE	TRUE The command table was cancelled by another command.
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command table. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"
CurrentStep	OUTPUT	INT	0	Step in command table currently being executed
StepCode	OUTPUT	WORD	16#0000	User-defined numerical value / bit pattern of the step currently being executed

See also

- List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)
- Overview of the Motion Control statements (Page 79)
- MC_Power: Enable, disable axis (Page 161)
- MC_Reset: Acknowledge error (Page 166)
- MC_Home: Home axes, set home position (Page 168)
- MC_Halt: Halt axis (Page 172)
- MC_MoveAbsolute: Absolute positioning of axes (Page 176)
- MC_MoveRelative: Relative positioning of axes (Page 180)
- MC_MoveVelocity: Move axes at preset rotational speed (Page 184)
- MC_MoveJog: Move axes in jogging mode (Page 189)
- MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0) (Page 194)

2.10 MC_ChangeDynamic

2.10.1 MC_ChangeDynamic: Changing the dynamic settings for the axis (as of technology object "Axis" V2.0)

Description

Motion Control instruction "MC_ChangeDynamic" allows you to change the following settings of the axis:

- Change the ramp-up time (acceleration) value
- Change the ramp-down time (deceleration) value
- Change the emergency stop ramp-down time (emergency stop deceleration) value
- Change the smoothing time (jerk) value

The effectiveness of the change is shown in the description of the tag (Page 130).

Requirements

- The technology object "Axis" has been added to Version V2.0.
- The technology object "Axis" has been configured correctly.

Override response

A MC_ChangeDynamic command cannot be aborted by any other Motion Control command.

A new MC_ChangeDynamic command does not abort any active Motion Control jobs.

Parameter

Parameter	Declaration	Data type	Default value	Description
Axis	INPUT	TO_Axis_1	-	Axis technology object
Execute	INPUT	BOOL	FALSE	Start of the command with a positive edge
Change RampUp	INPUT	BOOL	FALSE	TRUE Change ramp-up time in line with input parameter "RampUpTime"
RampUp Time	INPUT	REAL	5.00	Time (in seconds) to accelerate axis from standstill to configured maximum velocity without jerk limit. The change will influence the tag <Axis name>. Config.DynamicDefaults.Acceleration. The effectiveness of the change is shown in the description of this tag.
Change RampDown	INPUT	BOOL	FALSE	TRUE Change ramp-down time in line with input parameter "RampDownTime"
RampDown Time	INPUT	REAL	5.00	Time (in seconds) to decelerate axis from the configured maximum velocity to standstill without jerk limiter. The change will influence the tag <Axis name>. Config.DynamicDefaults.Deceleration . The effectiveness of the change is shown in the description of this tag.
Change Emergency	INPUT	BOOL	FALSE	TRUE Change emergency stop ramp-down time in line with input parameter "EmergencyRampTime"
Emergency RampTime	INPUT	REAL	2.00	Time (in seconds) to decelerate the axis from configured maximum velocity to standstill without jerk limiter in emergency stop mode. The change will influence the tag <Axis name>. Config.DynamicDefaults.EmergencyDeceleration . The effectiveness of the change is shown in the description of this tag.
ChangeJerk Time	INPUT	BOOL	FALSE	TRUE Change smoothing time according to the input parameter "JerkTime"
JerkTime	INPUT	REAL	0.25	Smoothing time (in seconds) used for the axis acceleration and deceleration ramps The change will influence the tag <Axis name>. Config.DynamicDefaults.Jerk . The effectiveness of the change is shown in the description of this tag.
Done	OUTPUT	BOOL	FALSE	TRUE The changed values have been written to the technology data block. The description of the tags will show when the change becomes effective.
Error	OUTPUT	BOOL	FALSE	TRUE An error occurred during execution of the command. The cause of the error can be found in parameters "ErrorID" and "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Error ID (Page 120) for parameter "Error"
ErrorInfo	OUTPUT	WORD	16#0000	Error info ID (Page 120) for parameter "ErrorID"

Note

Values can be entered at the input parameters "RampUpTime", "RampDownTime", "EmergencyRampTime" and "JerkTime" which exceed the admissible limits of the resulting parameters: "Acceleration", "Deceleration", "Emergency stop deceleration" and "Jerk".

Please note the equations and limit values in "Axis technology object" -> "Configuring the technology object" -> "Dynamics" and ensure that the values you input are within the valid range.

See also

List of ErrorIDs and ErrorInfos (technology objects as of V2.0) (Page 120)

Overview of the Motion Control statements (Page 79)

Changing the configuration of dynamics in the user program (Page 46)

Changing the homing configuration in the user program (Page 52)

MC_Power: Enable, disable axis (Page 161)

MC_Reset: Acknowledge error (Page 166)

MC_Home: Home axes, set home position (Page 168)

MC_Halt: Halt axis (Page 172)

MC_MoveAbsolute: Absolute positioning of axes (Page 176)

MC_MoveRelative: Relative positioning of axes (Page 180)

MC_MoveVelocity: Move axes at preset rotational speed (Page 184)

MC_MoveJog: Move axes in jogging mode (Page 189)

MC_CommandTable: Run axis jobs as movement sequence (as of technology object "Axis" V2.0) (Page 192)

Tag of the Axis technology object (Page 130)

Index

A

- Add empty line, 60
- Add separator line, 60
- Axis technology object:Active homing, 51
- Axis technology object:Add new object, 29
- Axis technology object:Axis name configuration, 31
- Axis technology object:Basic parameters, 30
- Axis technology object:Changing the configuration parameters for dynamics in the user program, 46
- Axis technology object:Changing the configuration parameters for homing in the user program, 52
- Axis technology object:Commissioning overview, 28
- Axis technology object:Config.DriveInterface. tag, 131
- Axis technology object:
 - Config.DynamicDefaults tag, 134
 - Axis technology object:Config.DynamicLimits tag, 133
 - Axis technology object:Config.General. tag, 130
 - Axis technology object:Config.Homing tag, 139
 - Axis technology object:Config.Mechanics tag, 132
- Axis technology object:
 - Config.PositionLimits_HW tag, 137
 - Axis technology object:
 - Config.PositionLimits_SW tag, 136
- Axis technology object:Configuration overview, 28
- Axis technology object:Configuration window icons, 30
- Axis technology object:Configuring acceleration, 41
- Axis technology object:Configuring active homing, 48
- Axis technology object:
 - Configuring approach/homing direction, 49
 - Axis technology object:Configuring deceleration, 41
- Axis technology object:
 - Configuring distance per motor revolution, 34
 - Axis technology object:
 - Configuring home position offset, 49
 - Axis technology object:
 - Configuring invert direction signal, 34
 - Axis technology object:Configuring jerk limiter, 41
- Axis technology object:
 - Configuring maximum velocity / start/stop velocity, 40
 - Axis technology object:Configuring passive homing, 47
 - Axis technology object:Configuring permit auto reverse at the hardware limit switch, 48
- Axis technology object:
 - Configuring pulses per motor revolution, 34
 - Axis technology object:Configuring ramp-down time, 41

- Axis technology object:Configuring ramp-up time, 41
- Axis technology object:
 - Configuring reference point position, 48, 49
 - Axis technology object:Configuring smoothing time, 42
- Axis technology object:
 - Configuring the homing speed, 49
 - Axis technology object:
 - Configuring the reference point switch input, 47
 - Axis technology object:Configuring the side of the reference point switch, 47, 49
- Axis technology object:
 - Configuring the velocity limiting unit configuration, 40
- Axis technology object:Diagnosis overview, 28
- Axis technology object:Drive enable configuration, 33
- Axis technology object:Drive ready configuration, 33
- Axis technology object:Drive signal configuration, 33
- Axis technology object:
 - Emergency stop deceleration configuration, 43
 - Axis technology object:ErrorBits tag, 146
 - Axis technology object:Extended parameters, 30
- Axis technology object:
 - General dynamics configuration, 40
 - Axis technology object:
 - Hardware and software components, 24
 - Axis technology object:
 - Hardware interface configuration, 31
 - Axis technology object:Mechanics configuration, 34
 - Axis technology object:MotionStatus tag, 142
 - Axis technology object:Passive homing, 50
 - Axis technology object:PTO and HSC configuration, 31
- Axis technology object:
 - Response when jerk limiter is activated, 44
 - Axis technology object:StatusBits tag, 143
 - Axis technology object:Tools, 27
 - Axis technology object:Updating the tags, 148
 - Axis technology object:User unit configuration, 33

B

- Basic knowledge
 - Required, 3

D

- Direction output and travel direction:relation, 16
- Drive signals, 14

H

Hardware and software limit switches:Function, 18
Hardware configuration for Motion Control S7-1200, 10
Homing:Homing modes, 20

I

Insert empty line, 60
Insert separator line, 60

J

Jerk limiter:Function, 19

L

Limit frequencies of pulse outputs, 11

M

Manual
 Purpose, 3
 Scope, 3
Maximum number of controllable drives, 11
MC_ChangeDynamic:Instruction, 194
MC_ChangeDynamic:Parameter, 195
MC_CommandTable:Instruction, 192
MC_CommandTable:Parameter, 193
MC_Halt:Function chart, 174
MC_Halt:Instruction, 172
MC_Halt:Parameter, 173
MC_Home:Instruction, 168
MC_Home:Parameter, 170
MC_MoveAbsolute:Function chart, 178
MC_MoveAbsolute:Instruction, 176
MC_MoveAbsolute:Parameter, 177
MC_MoveJog:Function chart, 191
MC_MoveJog:Instruction, 189
MC_MoveJog:Parameter, 190
MC_MoveRelative:Function chart, 182
MC_MoveRelative:Instruction, 180
MC_MoveRelative:Parameter, 181
MC_MoveVelocity:Function chart, 187
MC_MoveVelocity:Instruction, 184
MC_MoveVelocity:Parameter, 185
MC_Power:Function chart, 165
MC_Power:Instruction, 161
MC_Power:Parameter, 162
MC_Reset:Instruction, 166
MC_Reset:Parameter, 166

Motion Control CPU S7-1200
 Guidelines, 21

P

Pulse and direction output, 13
Pulse interface:Principle, 15
Purpose
 Manual, 3

R

Required basic knowledge, 3

S

Scope
 Manual, 3
Servo motor, 10
Stepper motor, 10

T

Technology object command table:Add new object, 54
Technology object command table:
 Basic parameters, 55
Technology object command table:
 Command table configuration, 56
Technology object command table:
 Configuration window icons, 55
Technology object command table:
 Configuring activate warnings, 56
Technology object command table:
 Configuring duration, 59
Technology object command table:
 Configuring position / travel path, 58
Technology object command table:
 Configuring the command type, 57
Technology object command table:
 Configuring the next step, 59
Technology object command table:
 Configuring the step code, 59
Technology object command table:
 Configuring use axis parameters of, 56
Technology object command table:
 Configuring velocity, 58
Technology object command table:
 Extended parameters, 55
Technology object command table:
 General configuration, 56

Technology object command table:
Shortcut menu commands, 60
Technology object command table:Tools, 53
Technology object command table:Usage, 53
Technology object, axis and command table:
List of ErrorIDs and ErrorInfo, 120
TO_Axis_PTO, 29

