SIEMENS

SIMATIC

Component based Automation Creating PROFInet Components

Manual

Edition 12/2003

A5E00248719-01

Safety Guidelines

This manual contains notices intended to ensure personal safety, as well as to protect the products and connected equipment against damage. These notices are highlighted by the symbols shown below and graded according to severity by the following texts:

⚠ **Danger**
indicates that death, severe personal injury or substantial property damage will result if proper precautions are not taken.

⚠ **Warning**
indicates that death, severe personal injury or substantial property damage can result if proper precautions are not taken.

⚠ **Caution**
indicates that minor personal injury can result if proper precautions are not taken.

Caution
indicates that property damage can result if proper precautions are not taken.

Notice
draws your attention to particularly important information on the product, handling the product, or to a particular part of the documentation.

Qualified Personnel

Only **qualified personnel** should be allowed to install and work on this equipment. Qualified persons are defined as persons who are authorized to commission, to ground and to tag circuits, equipment, and systems in accordance with established safety practices and standards.

Correct Usage

Note the following:

⚠ **Warning**
This device and its components may only be used for the applications described in the catalog or the technical description, and only in connection with devices or components from other manufacturers which have been approved or recommended by Siemens.

This product can only function correctly and safely if it is transported, stored, set up, and installed correctly, and operated and maintained as recommended.

Trademarks

SIMATIC®, SIMATIC HMI® and SIMATIC NET® are registered trademarks of SIEMENS AG.

Third parties using for their own purposes any other names in this document which refer to trademarks might infringe upon the rights of the trademark owners.

Siemens AG
Bereich Automation and Drives
Geschaeftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nuernberg

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

Technical data subject to change.

Siemens Aktiengesellschaft
A5E00248719-01

Preface

Purpose of the manual

This manual provides support for creating PROFInet components with STEP 7.

It is directed at plant and machine constructors using SIMATIC automation systems in component based automation.

Required knowledge

General knowledge of automation technology and specialised knowledge the SIMATIC automation devices in use are prerequisites for understanding this documentation.

Since PROFInet components are generated with the basic STEP 7 software, you should have knowledge about operating this basis software as described in the manual, "Programming with STEP 7".

In addition, you are required to have a good working knowledge of computers or other equipment similar to PCs (e.g. programming devices) under the operating systems Windows 2000 or Windows XP.

Validity of the manual

This manual is valid for the software package SIMATIC iMap V2.0.

Your guide through the manual

The following manual is divided into the following topic areas:

- Chapter 1 describes how PROFInet components are created
- Chapter 2 describes the configuration option and special features of individual types of devices – control center modules, distributed I/O, etc. – that need to be taken into consideration when creating PROFInet components.
- Chapter 3 describes the characteristics and creation of Singleton components.
- The appendix contains reference information.
- The glossary contains explanations of important terms.
- The index helps you to quickly find textual information on important keywords.

Its place in the information environment

This manual is part of the SIMATIC iMap documentation package. The documentation is supplied with the software and includes the electronic manuals in PDF format:

- Getting started with SIMATIC iMap
  Brief introduction to working with component based automation and SIMATIC iMap.
- Configuring Plants with SIMATIC iMap
  Contains detailed information and instructions about configuration and commissioning of plants using SIMATIC iMap.
- Creating PROFInet Components – this manual
  Contains detailed information and instructions about creating PROFInet components with STEP 7 and the use of SIMATIC devices in component based automation.
- Commissioning the System, Tutorial
  Contains detailed information and step-by-step instructions using examples, from the creation of PROFInet components to commissioning of the complete plant.

In addition, the entire documentation is available as an HTML basic help in SIMATIC iMap.

References for Creating PROFInet Components with STEP 7

The following table provides a "thread" through all manuals about component based automation and SIMATIC iMap for creating PROFInet components. The chapters are listed in the order required for performing this task.

Manual	Chapter		Title
	Required	Optional	
Getting Started with SIMATIC iMap		1	Getting Started - Introduction
		2	Step 1: Defining PROFInet components
		3	Step 2: Creating PROFInet components with STEP 7
Creating PROFInet Components	1.1		Basic procedure
	1.3		Creating the project with STEP 7
	1.4.1		Properties of PROFInet interfaces
		1.4.2	Using the PROFInet interface editor
	1.4.3		Creating PROFInet interface
		1.4.4	Changing PROFInet interfaces
	1.6.1		Creating PROFInet components in SIMATIC Manager
		1.7	Importing PROFInet components to a library
		1.8	Modifying PROFInet components
		2	SIMATIC devices as PROFInet components
Commissioning Systems, Tutorial		2	Part 1: Creating PROFInet components

Conventions

Menu commands are written in bold letters, for example: **File > Save**.

Placeholders are set in angle brackets, for example <File name>.

Further support

Please contact your local SIEMENS partner if you have any further queries on the products described in this manual.

http://www.siemens.com/automation/partner

http://www.ad.siemens.de/cba/

Training Centers

We offer a range of courses to help you get started with xxxx and the S7 automation system. Please contact your regional training center or the central training center in D 90327 Nuremberg.

Phone: +49 (911) 895-3200.

Internet: http://www.sitrain.com

A&D Technical Support

Open round the clock, worldwide:

Worldwide (Nuremberg) **Technical Support** Local time: 0:00 to 24:00 / 365 days Phone: +49 (180) 5050-222 Fax: +49 (180) 5050-223 EMail: adsupport@ siemens.com GMT: +1:00		
Europe / Africa (Nuremberg) **Authorization** Local time: 8:00 to 17:00 Phone: +49 (180) 5050-222 Fax: +49 (180) 5050-223 EMail: adsupport@ siemens.com GMT: +1:00	**United States** (Johnson City) **Technical Support and Authorization** Local time: 8:00 to 17:00 Phone: +1 (423) 262 2522. +1-52520779 740 3699, Fax: EMail: simatic.hotline@ sea.siemens.com GMT: -5:00	**Asia / Australia** (Beijing) **Technical Support and Authorization** Local time: 8:00 to 17:00 Phone: +86 10 64 75 75 75 Fax: +86 10 64 74 74 74 E-Mail: adsupport.asia@ siemens.com GMT: +8:00
German and English spoken at the Technical Support and Authorization hotlines.		

Service & Support on the Internet

In addition to our documentation services, we also offer you our knowledge base on the Internet.

http://www.siemens.com/automation/service&support

Here, you will find:

- the newsletter which constantly provides you with up-to-date information about your products.
- your appropriate documentation via our Service & Support search engine
- a forum for the exchange of information between users and specialists worldwide
- your local Automation & Drives partner via our partner database.
- information on repairs, replacement parts and onsite service. You will find more information under "Services“.

Contents

Figures

Tables

Creating PROFInet components with STEP 7 1

1.1 Basic procedure for creating PROFInet components with STEP 7

Follow the steps described below to create PROFInet components using STEP 7:

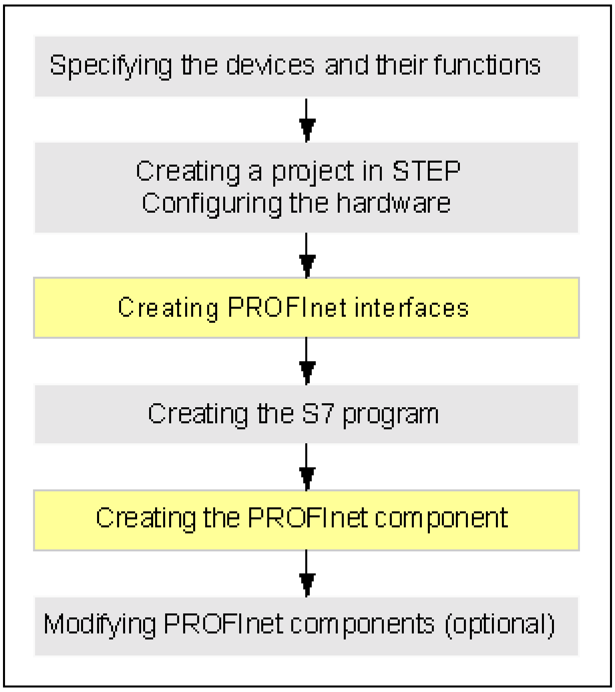

Specifying the devices and their functions
Creating a project in STEP
Configuring the hardware
Creating PROFInet interfaces
Creating the S7 program
Creating the PROFInet component
Modifying PROFInet components (optional)

Figure 1-1 Creating PROFInet components with STEP 7 - Basic procedure

Available tools

In SIMATIC Manager, use the following tools:

- Create the PROFInet interface with the PROFInet interface editor. In SIMATIC Manager, highlight the station, and then select the **Edit > Create PROFInet Interface** menu command.
- Create the PROFInet component in SIMATIC Manager using the **Edit > Create PROFInet component** menu command.
- For all other steps, use the normal tools in SIMATIC Manager, such as HW Config or program editor.

1.2 Defining devices and their functions

Recommendation

Create your own PROFInet components from plant components

- that occur more than once (reusable) or/and
- that create high communication traffic on the Ethernet or PROFIBUS and
- that form a discrete mechanical and electrical device.

PROFIBUS devices can only be used in conjunction with a PROFInet device that acts as a PROFIBUS master with proxy functionality.

Basic procedure

1. Break down the plant into individual, discrete technological modules so that each module forms a separate component within the overall project.
2. Define the programmable controllers that you need for the individual PROFInet components, including any peripheral devices.
3. Define the interfaces of the PROFInet components, i.e. the inputs and outputs for communication with other components.
4. Define the communication networks required – Ethernet or PROFIBUS.
5. Write the programs with the component functions.

1.3 Create the project using STEP 7

The STEP 7 basic project is the STEP 7 project from which the PROFInet component is created.

To create the STEP 7 basic project:

1. Create a STEP 7 project in SIMATIC Manager. Then create the PROFInet component from the station of the project.
2. Add the necessary subnets:
   - Industrial Ethernet for PROFInet devices
   - PROFIBUS for PROFInet devices with proxy functionality, PROFIBUS devices (DP_Slaves) or local DP slaves.
3. Configure the hardware, then configure the modules in HW Config, noting the possible configurations for SIMATIC devices in the "Configurations for PROFInet component" section.

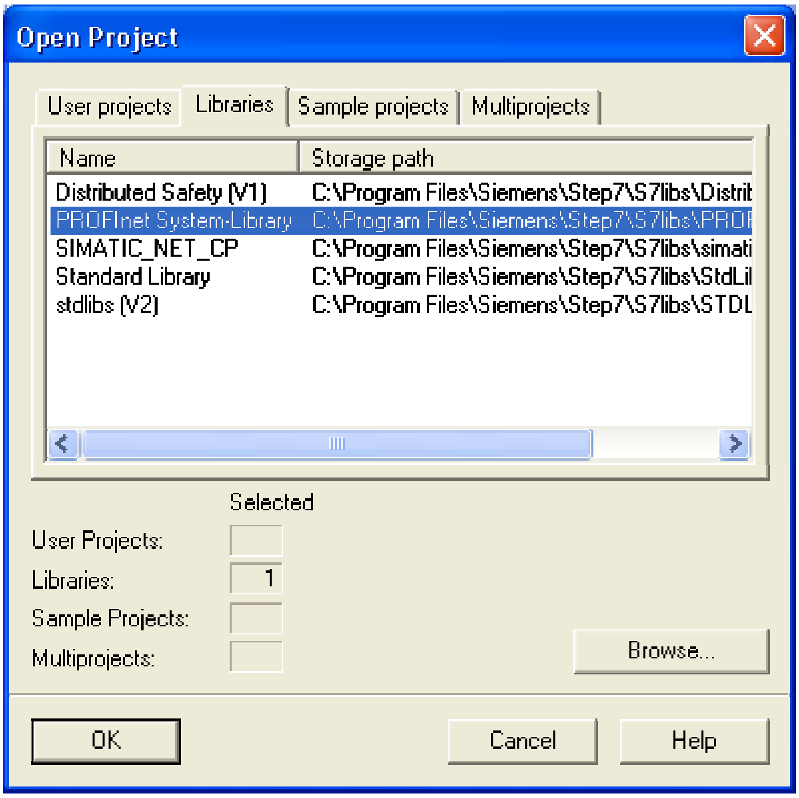
Note

Detailed descriptions can be found in the "Commissioning Systems, Tutorial" manual.

STEP 7 library with templates for PROFInet components

An S7 library containing all the blocks you need as templates for creating PROFInet components is installed in STEP 7.

The PROFInet system library can be found in the STEP 7 \s7libs\PROFInet directory.



Name	Storage path
Distributed Safety (V1)	C:\Program Files\Siemens\Step7\S7libs\Distrit
PROFInet System-Library	C:\Program Files\Siemens\Step7\S7libs\PROF
SIMATIC_NET_CP	C:\Program Files\Siemens\Step7\S7libs\simati
Standard Library	C:\Program Files\Siemens\Step7\S7libs\StdLil
stdlibs (V2)	C:\Program Files\Siemens\Step7\S7libs\STDL

Selected

User Projects:

Libraries: 1

Sample Projects:

Multiprojects:

Figure 1-2 Opening the PROFInet system library

Table 1-1 Content of the PROFInet system library

Folder	Content: Blocks for...	Device type
CP 300	CP 343-1 PN	PROFInet device
CPU 317-2 PN	CPU 317-2 PN/DP	PROFInet device
I-DP slave	Intelligent DP slaves	PROFIBUS device with programmable functionality
WinLC PN	Windows logic controller WinLC PN	PROFInet device

To copy the blocks to the STEP 7 basic project:

Only for PROFInet components with programmable functionality:

1. In SIMATIC Manager select the **File > Open** menu command from the "Libraries" tab to open the "PROFInet system library".
2. In the library, open the block folder corresponding to the device type of the PROFInet component to be created, and copy all the block templates to the block folder of the basic project.

Tip

Copy all the blocks for the each type of device (including OBs) to the S7 program folder since you will need them for creating the PROFInet components.

Configuring the assignment of message numbers

In SIMATIC Manager, menu **Options > Customize**, "Message numbers" tab, check whether the "Always assign unique message numbers within CPU" option is active. Activate the option if this is not the case.

This option must me activated when you create PROFInet components with STEP 7.

Next Steps

Create the interface DB in the PROFInet interface editor by converting the external inputs and outputs of the future PROFInet component into declarations for the interface DB.

1.4 Editing PROFInet interfaces

1.4.1 Properties of the PROFInet interfaces

1.4.1.1 Properties of the PROFInet interface

Definition

The PROFInet interface is the component interface of the PROFInet component. Every PROFInet component has an interface that you can use to communicate with other PROFInet components and the HMI system. There must be a description of this interface in the STEP 7 basic project when the PROFInet component is created. The interface description is stored in the STEP 7 basic project in a data block - the interface DB.

When you create the PROFInet component, the structure of the interface, i.e. the number and properties of the I/Os, is determined from this interface DB.

Optionally, the PROFInet interface can be extended with non-interconnectable connectors for HMI/MES.

PROFInet interface editor

There is a dedicated editor for creating and editing PROFInet interfaces. If a station has been selected in SIMATIC Manager, it can then be called up using the **Edit > Create PROFInet Interface** menu command. The edited blocks are automatically assigned PROFInet properties in the form of attributes in the PROFInet interface editor, so this editor is only able to create or open blocks of the PROFInet interface.

Blocks of the PROFInet interface

The PROFInet interface consists of the interface DB, which contains the interface definition for the PROFInet component. A variable declaration for the interface DB defines a connector and its properties. The interface DB contains all the interconnectable and non-interconnectable connectors that can be accessed via PROFInet protocols. Optionally, the interface DB may also contain non-interconnectable connectors that are accessible via S7 protocols for HMI/MES only. These are known as S7 variables. The S7 variable options depend on the type of device used.

Optionally, internal HMI interface DBs may be added as an HMI extension to the PROFInet interface. An HMI interface DB contains only S7 variables for HMI/MES.

For PROFInet components with programmable functionality, the interface DB is used as a storage area for data that is sent via the interface of the PROFInet component. Other STEP 7 blocks can then access this data directly via the interface DB at run time.

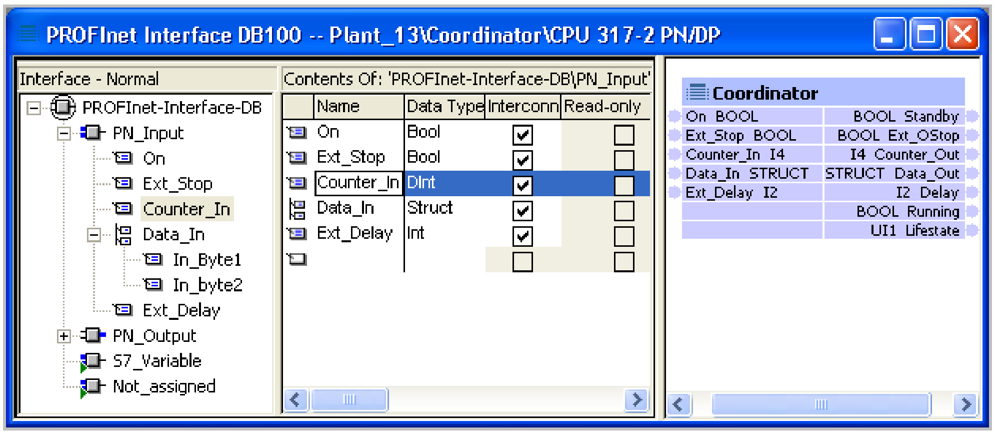
Structure of the PROFInet interface DB

The PROFInet interface DB consists of the following sections:

- PN_Input – contains the inputs that can be accessed via PROFInet protocols,
- PN_Output – contains the outputs that can be accessed via PROFInet protocols,
- S7_Variable – contains the non-interconnectable connectors for HMI/MES that can be accessed via S7 protocols. This section is only present in PROFInet components with programmable functionality.
- Unassigned – contains connectors that have not yet been assigned to the PROFInet interface. These connectors must be moved to one of the other three sections in order to become part of the PROFInet interface.

Every section contains variable declarations, where a variable declaration corresponds to a connector of the PROFInet component.

The following picture shows an interface DB in the PROFInet interface editor.



Name	Data Type	Interconn	Read-only
On	Bool	☑	□
Ext_Stop	Bool	☑	□
Counter_In	DInt	☑	□
Data_In	Struct	☑	□
Ext_Delay	Int	☑	□
		□	□

Coordinator	
On BOOL	BOOL Standby
Ext_Stop BOOL	BOOL Ext_OStop
Counter_In I4	I4 Counter_Out
Data_In STRUCT	STRUCT Data_Out
Ext_Delay I2	I2 Delay
	BOOL Running
	UI1 Lifestate

Figure 1-3 Example: Interface DB in the PROFInet interface editor

Structure of the HMI interface DB

The HMI interface DB contains only S7 variables, i.e. non-interconnectable connectors for HMI/MES that can be accessed via S7 protocols.

Assignment to the device, active and inactive blocks

The PROFInet interface DB is always permanently assigned to a device. The STEP 7 basic project may contain several PROFInet interface DBs, but only one of these may be active for a given device, i.e. taken into account when the PROFInet component is created.

This allows different PROFInet interfaces to be defined for a technological function. The PROFInet interface DB most recently saved in the interface editor is always the active block that is taken into account when the PROFInet component is created. All other PROFInet interface DBs are inactive in the STEP 7 basic project.

PROFInet properties

The following table shows the PROFInet properties of the blocks as they are displayed in the PROFInet interface editor:

Table 1-2 PROFInet properties

Type of block	Property	Explanation
PROFInet interface DB	PN interface (active)	Global data block - contains the interface description that is taken into account when the PROFInet component is created.
	PN interface (inactive)	Global data block - contains an interface description that is **not** taken into account when the PROFInet component is created.
internal HMI interface DB or FB	HMI (active)	Extension of the PROFInet interface to include non-interconnectable HMI/MES connectors that are taken into account when the PROFInet component is created.
	HMI (inactive)	Extension of the PROFInet interface to include non-interconnectable HMI/MES connectors that are **not** taken into account when the PROFInet component is created.
DB or FB	None	The block contains no PROFInet properties (attributes).

Note

If an HMI interface FB is identified as "HMI (inactive)", then the assigned instance DB also receives the property "HMI (inactive)", and is not taken into account when the PROFInet component is created.

1.4.1.2 Properties of the connectors

For every connector there must be a declaration line in one of the sections. The declaration lines are listed in the form of a table in which the columns contain the following connector properties:

Column	Meaning	To edit
Name	Name of the connector (see "Naming conventions").	Enter unique names for the connectors.
Data type	Data type of the connector (e.g. BOOL, WORD, STRING, see "Data types").	The default is BOOL. You can change the data type if necessary.
Address	Address of the variable in the DB (format BYTE.BIT).	Not possible, the address is assigned automatically.
Interconnecta ble	If this option is selected, then the connector can be seen in the SIMATIC iMap plant view, and may be interconnected. The interconnectable connectors of the PROFInet interface are displayed in the right-hand editor window.	Optional A PROFInet interface DB must contain at least one interconnectable connector.
HMI/MES	If this option is selected, then the variable is accessible for HMI or MES via OPC.	Optional If the "Interconnectable“ option is selected, then "HMI/MES“ is automatically selected as well.
Read-Only	If this option is selected, then only read access to the variable is permitted. This option is only of relevance to S7 variables (HMI/MES).	PN_Input: This option cannot be changed. PN_Output: This option is automatically assigned to all variables. It cannot be changed. S7 variable: This option can be set.
Initial value	Initial value of the connector that is accepted as the current value when the object is saved for the first time.	PROFInet devices: You can change the default setting, if necessary. The value must correspond to the data type. PROFIBUS devices: The default settings for the inputs (section PN_Input) cannot be changed.
Comment	Comment about the connector (up to 80 characters).	Optional

Connector names

A connector name must not exceed 24 alphanumeric characters. Always follow the naming conventions for connectors when assigning names (see "Naming conventions").

Data types

The declaration lines for the interface DB may be configured with S7 data types.

Only arrays and structures of simple data types may be declared. Arrays and structures of arrays or structures are not permitted, in contrast to STEP 7.

The S7 data types are mapped onto PROFInet data types as per Microsoft OLE 2.0 (see "Connector data types").

Data length of the declared connectors

The maximum data length of a connector and the maximum data length of all connectors can be seen in the object properties of the section concerned in the PROFInet interface editor.

Note

A connector of the data type array or struct always takes up at least 2 bytes.

A connector of the data type string always takes up at least 4 bytes.

For PROFIBUS devices with programmable functionality: For a connector of the data type String, the maximum length of the working data is just 30 bytes, which corresponds to 30 characters, i.e. STRING[30].

Number of connectors

The number of interconnectable connectors can be seen from the object properties of the section concerned in the PROFInet interface editor.

The PN_Input section of an interface DB may not have more than 300 interconnectable inputs and the PN_Output section may not have more than 300 interconnectable outputs, including the life-state output.

There is no limit to the number of non-interconnectable connectors.

Life-state output

Although it is not explicitly declared in the interface DB, every PROFInet component created in STEP 7 is automatically given a life-state output, which allows the status of the device to be monitored by the communication partners. Information about the life-state output can be found under "The "life-state" connector on SIMATIC devices".

Interconnectable and non-interconnectable connectors

- Connectable connectors relate to process values that are part of the component interface and can be transferred during PROFInet communication.

  The interconnectable connectors appear in the plant view and can be accessed via OPC (OLE for Process Control).
- Non-interconnectable connectors relate to data that is not part of the component interface, but may only be contained in the OPC symbol file if the HMI/MES option is activated. Such connectors do not appear in the plant view, and the data can only be accessed via OPC.

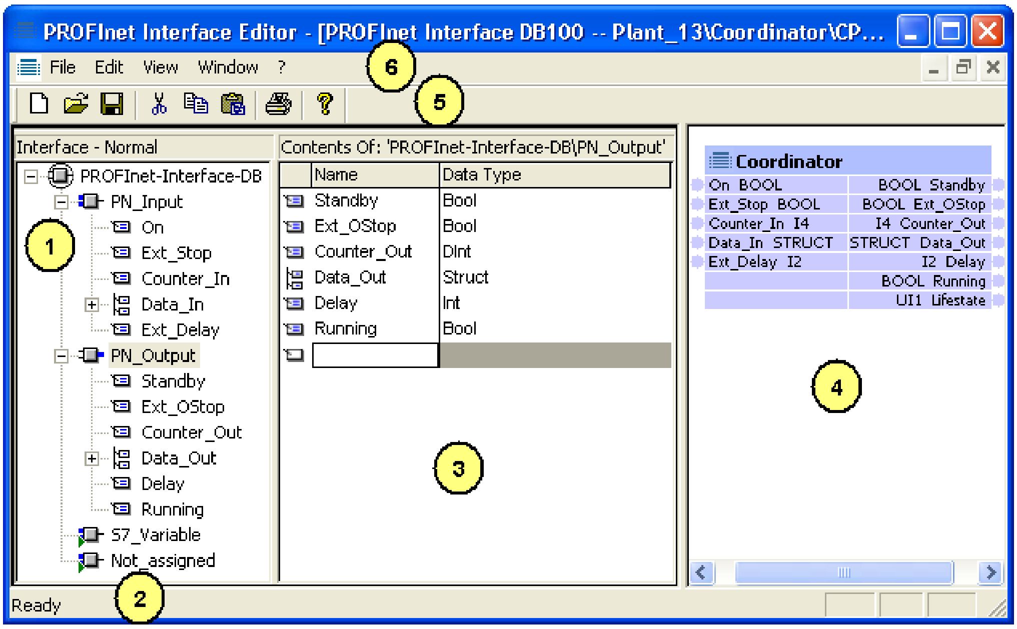
"HMI/MES" option

Connectors with this option are entered in the OPC symbol file and may be used via OPC applications for operator input and for monitoring, commissioning or diagnostics.

1.4.2 Using the PROFInet interface editor

1.4.2.1 User interface of the PROFInet interface editor

Structure of the user interface



Contents Of: 'PROFInet-Interface-DB\PN_Output'

Name	Data Type
Standby	Bool
Ext_OStop	Bool
Counter_Out	DInt
Data_Out	Struct
Delay	Int
Running	Bool

Figure 1-4 PROFInet interface editor - Structure of the user interface

Table 1-3 Key

No.	Meaning
1	Variable overview
2	Status Bar
3	Variable detailed view
4	View of the technological function in SIMATIC iMap
5	Toolbar
6	Menu bar

Call from SIMATIC Manager

The PROFInet interface editor is called from SIMATIC Manager by highlighting a station and then selecting:

- **Edit > Create PROFInet Interface** from the menu bar or
- **Create PROFInet Interface** from the context menu.

Working with the PROFInet interface editor

Features used in Windows applications are available here:

- Menu commands (menu bar, context menus or icons in the toolbar),
- Drag&Drop,
- Double-clicking on a selected object
- Keyboard input

Calling help

To call up help about a window in the user interface, click in the window and press function key F1.

1.4.2.2 Menu commands in the PROFInet interface editor

"File" menu

Command	Description
New	Creates a new block (interface DB)
Open	Opens an existing block
Close	Closes the current block
Save	Saves the current block
Print	Prints the current block
Page View	Displays a whole page
Printer Setup	Changes the printer and the printing options
Recently Opened Interfaces	Opens one of the recently opened interfaces
Exit	Exits the interface editor

"Edit" menu

Command	Description
Cut	Deletes the selection and transfers it to the clipboard
Copy	Copies the selection and transfers it to the clipboard
Paste	Inserts the clipboard contents

"View" menu

Command	Description
Toolbar	Shows or hides the toolbar
Status Bar	Shows or hides the status bar

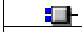
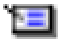
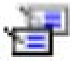
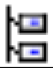
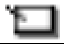
"Window" menu

Command	Description
Cascade	Cascades all open windows one behind the other
Horizontally	Tiles all open windows from top to bottom
Vertically	Tiles all open windows from left to right
Arrange Icons	Arranges the icons in the bottom part of the window
Active Window	Activates the selected window

"Help" menu

Command	Description
Help Topics	Calls up help about the PROFInet interface editor
About	Displays information about the program, version number and copyright

1.4.2.3 Icons in the PROFInet interface editor

Icon	Meaning
	Declaration section for inputs (PN_Input).
	Declaration section for outputs (PN_Output).
	Declaration section for S7 variables (S7_Variable) and Declaration section for unassigned connectors (Unassigned)
	DP_MasterInputSlot or DP_MasterOutputSlot
	Elementary data type (e.g. Bool, Byte, Word, Int).
	Array with elements of an elementary data type.
	Structured data type (STRUCT).
	Empty element (always the last element in a folder).

1.4.2.4 Setting the column width in the variable detail view

Procedure

Position the mouse pointer in the table header on the right-hand edge of the column, hold down the left mouse button and move the edge of the column in the desired direction.

Double-click on the right-hand edge of a column to automatically set it to the optimum width.

You can also use the F7 and F8 function keys to increase or reduce the size of columns.

Note

There is a minimum size for columns. It is not possible to create smaller columns. To hide a column, select **Show/Hide Columns** from the context menu.

1.4.2.5 Showing / hiding columns in the variable detail view

Show / Hide

1. Select **Show/Hide Columns** from the context menu or press the F11 key.
2. From the dialog box that appears, select the columns that you wish to show or hide.
3. Click on the "--->" or "<---" button.
4. Use the "Up" or "Down" button to change the order of the columns.

Hint: Save your new column layout

To save the column layout for the current PROFInet interface and all further interfaces, select the "Save current column view as user default" option.

You can save separate column settings for each section (PN_Input, PN_Output, etc). Every new object of the same type will then appear with your stored settings.

You can also modify these settings during a session.

You can also click on the "Default" button to restore the previously saved or system default column layouts.

Note

If you modify the column settings for an object type and save it as a user default, this will only take effect on objects of this type that have not yet been opened. The original column settings will be retained by any objects of this type that are already open. To apply your new column settings to these objects as well, select **Default > User**.

1.4.2.6 Keyboard commands in the variable overview

Function	Keys
One object up.	Up
One object down.	Down
Display the next hierarchical level	Right or +
Close a hierarchical level	Left or -
Show All Levels	x (multiplication key on the keypad)
Go to topmost object in the directory tree	Pos1
Go to bottom object in the directory tree	End
Go to higher level object	Home
Move up one page	Page Up
Move down one page	Page Down

1.4.2.7 Keyboard commands in the variable detail view

Function	Keys
Select all elements	Ctrl + A
Switch selected field to edit mode	F2
Highlight the entire text in a field	F3
Move one field to the right	Right
Move one field to the left	Left
Move one field up	Up
Move one field down	Down
To the first column	Ctrl + Left
To the last column	Ctrl + Right
Move up one page	Page Up
Move down one page	Page Down
Highlight several rows	Shift + Up / Shift + Down
Highlight all rows up to the start.	Shift + Ctrl + Up
Highlight all rows down to the end	Shift + Ctrl + Down
To next field of the table	Tab
To previous field of the table	Shift + Tab
Go to higher level object	Home
Reduce width of column	F7
Increase width of column	F8
Confirm input and move to next row	Enter
Run a defined standard operation for the highlighted object (e.g. Call object properties)	Shift + Enter
Activate / deactivate an option (check a check box).	Space bar
Insert a variable	Alt + Insert
Delete a variable	Delete
Show/hide columns	F11

1.4.3 Creating PROFInet interfaces

1.4.3.1 Creating a PROFInet interface DB - Basic procedure

Requirements

- The inputs and outputs of the component interface have been defined.
- The STEP 7 basic project has been created.
- The hardware has been configured.

Creating the PROFInet interface DB - Basic procedure

The following steps are needed to create the interface DB for the future PROFInet component:

1. Open the STEP 7 basic project for the PROFInet component in SIMATIC Manager.
2. Start the PROFInet interface editor.
3. Open the PROFInet interface DB. There are two options:
   - Open an existing interface DB. To do this,
   - select Create a New Interface DB.
4. In the "PN_Input" section, enter the connectors that are defined as inputs of the technological function of the PROFInet component, and assign the necessary properties to the entries: name, data type, interconnectable, etc.
5. In the "PN_Output" section, enter the connectors that are defined as outputs of the technological function of the PROFInet component, and assign the necessary properties to the entries: name, data type, interconnectable, etc.
6. For intelligent PROFIBUS devices, enter the HMI connectors in the "S7_Variable" section.
7. Save the new PROFInet interface DB.

Detailed information...

Detailed information about the individual steps can be found in the following sections.

How many PROFInet interface DBs are needed?

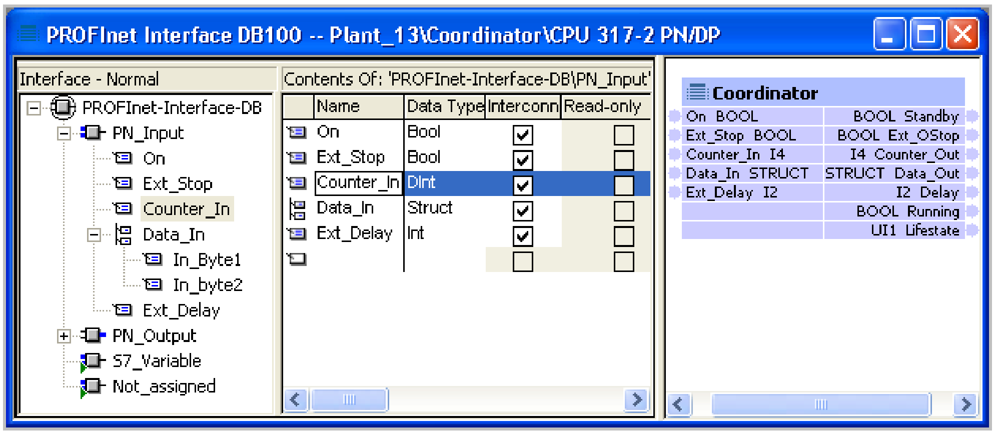
Just one PROFInet interface DB is needed to create a PROFInet component.

The following applies according to the type of device used:

- Create a basic project with a station for each PROFInet component with programmable functionality (PROFInet devices or intelligent PROFIBUS devices (I-slaves)). There must be only one active interface DB in the user program block folder.
- For PROFInet components with fixed functionality (PROFIBUS slaves), create a basic project with a SIMATIC 300 station that forms a DP master system for one or more DP slaves. In this case, the PROFInet component is only created from a DP slave, and the DP master is not part of the component.

  The user program block folder for such a station can thus contain several interface DBs, but just one active PROFInet interface DB for each device.

Example 1: Interface DB for a PROFInet device



PROFInet Interface DB100 -- Plant_13\Coordinator\CPU 317-2 PN/DP

Interface - Normal

- PROFInet-Interface-DB
  - PN_Input
    - On
    - Ext_Stop
    - Counter_In
    - Data_In
      - In_Byte1
      - In_byte2
    - Ext_Delay
  - PN_Output
  - S7_Variable
  - Not_assigned

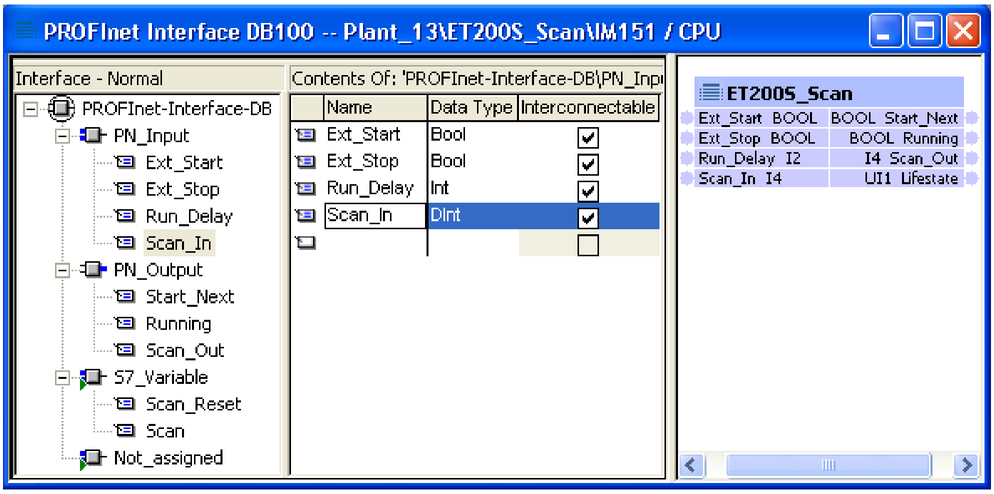
Contents Of: 'PROFInet-Interface-DB\PN_Input'

Name	Data Type	Interconn	Read-only
On	Bool	☑	□
Ext_Stop	Bool	☑	□
Counter_In	DInt	☑	□
Data_In	Struct	☑	□
Ext_Delay	Int	☑	□
		□	□

Coordinator	
On BOOL	BOOL Standby
Ext_Stop BOOL	BOOL Ext_OStop
Counter_In I4	I4 Counter_Out
Data_In STRUCT	STRUCT Data_Out
Ext_Delay I2	I2 Delay
	BOOL Running
	UI1 Lifestate

Figure 1-5 Interface DB for a SIMATIC 300 station with CPU 317-2PN/DP

Example 2: Interface DB for a PROFIBUS device, DP slave with programmable functionality



PROFInet Interface DB100 -- Plant_13\ET200S_Scan\IM151 / CPU

Interface - Normal

- PROFInet-Interface-DB
  - PN_Input
    - Ext_Start
    - Ext_Stop
    - Run_Delay
    - Scan_In
  - PN_Output
    - Start_Next
    - Running
    - Scan_Out
  - S7_Variable
    - Scan_Reset
    - Scan
  - Not_assigned

Contents Of: 'PROFInet-Interface-DB\PN_Inpu

Name	Data Type	Interconnectable
Ext_Start	Bool	☑
Ext_Stop	Bool	☑
Run_Delay	Int	☑
Scan_In	DInt	☑
		□

ET200S_Scan	
Ext_Start BOOL	BOOL Start_Next
Ext_Stop BOOL	BOOL Running
Run_Delay I2	I4 Scan_Out
Scan_In I4	UI1 Lifestate

Figure 1-6 Interface DB for ET 200S with IM 151/CPU

1.4.3.2 Open the interface DB in the PROFInet interface editor

In the PROFInet interface editor, you can create a new interface DB or open an existing PROFInet interface DB.

Start the PROFInet interface editor

1. Open the STEP 7 basic project for the PROFInet component in SIMATIC Manager.
2. Highlight the station for the basic project and select:
   - **Edit > Create PROFInet Interface** from the menu bar or
   - **Create PROFInet Interface** from the context menu.

The interface editor starts and the "New/Open PROFInet interface" dialog opens.

The procedure for creating a new interface DB is as follows:

1. There are two options:
   - If the PROFInet interface editor has not yet started, start it from SIMATIC Manager (see above).
   - If the PROFInet interface editor has started, select the **File > New...** menu command.

     The "New/Open PROFInet interface" dialog opens in both cases.
2. In the left-hand window of the "New/Open PROFInet interface" dialog box, select the station and the device to which the PROFInet interface is to be assigned. If necessary, click on the "Browse" button to look up the path of the required STEP 7 project.
3. Make sure that the "New" check box is checked.
4. Confirm by clicking on the "OK" button.

   Result: The properties dialog of the newly created block opens.
5. In the "Name and type" box, enter the required block number and select the "Global DB" block type.
6. You can also enter other block properties, such as a symbolic name.
7. Confirm by clicking on the "OK" button. Result: The block is opened in the PROFInet interface editor.
8. In the "PN_Input" section, enter the connectors that are defined as inputs of the technological function of the PROFInet component, and assign the necessary properties to the entries: name, data type, interconnectable, etc.

   Result: The interconnectable connectors appear in the graphical representation of the technological function (in the right-hand window of the interface editor).
9. In the "PN_Output" section, enter the connectors that are defined as outputs of the technological function of the PROFInet component, and assign the necessary properties to the entries: name, data type, interconnectable, etc.

   Result: The interconnectable connectors appear in the graphical representation of the technological function (in the right-hand window of the interface editor).
10. For intelligent DP slaves (optional): Enter the HMI connectors in the "S7_Variable" section.
11. Save your new PROFInet interface DB using the **File > Save** menu command.

To open an existing interface DB:

1. There are two options:
   - If the PROFInet interface editor has not yet started, start it from SIMATIC Manager (see above).
   - If the PROFInet interface editor has started, select the **File > Open...** menu command.

     The "New/Open PROFInet interface" dialog opens in both cases.
2. In the left-hand window, select the station and the device to which the PROFInet interface is to be assigned. If necessary, click on the "Browse" button to look up the path of the required STEP 7 project.

   Once you have highlighted a device, the blocks that can be opened for this device are displayed in the right-hand window. The block that already contains a PROFInet interface for this device is identified as "active" in the "PROFInet property" column.
3. Make sure that the "Open" check box is checked. If the "New" box is checked, you will not be able to select a block.
4. Highlight a block and click on the "OK" button. Result: The block is opened in the PROFInet interface editor.
5. In the "PN_Input" section, enter the connectors that are defined as inputs of the technological function of the PROFInet component, and assign the necessary properties to the entries: name, data type, interconnectable, etc.

   Result: The interconnectable connectors appear in the graphical representation of the technological function (in the right-hand window of the interface editor).
6. In the "PN_Output" section, enter the connectors that are defined as outputs of the technological function of the PROFInet component, and assign the necessary properties to the entries: name, data type, interconnectable, etc.

   Result: The interconnectable connectors appear in the graphical representation of the technological function (in the right-hand window of the interface editor).
7. For intelligent DP slaves (optional): Enter the HMI connectors in the "S7_Variable" section.
8. Save the PROFInet interface DB using the menu command **File > Save**.

Hint: Displaying blocks

Check the "Display all interface-compliant DBs and FBs" check box to determine which blocks from the S7 program container should appear for selection in the "Open" dialog.

- If the option is deactivated, only the blocks with PROFInet properties are displayed. These may be:
  - active PROFInet interface DB
  - inactive PROFInet interface DB
  - HMI interface FB or global HMI interface DB
- If the option is activated, all the blocks (DBs and FBs) that could be used to create a PROFInet interface are displayed, i.e. the global data blocks and function blocks without PROFInet properties are also displayed.

The procedure for opening an existing global DB is as follows:

To edit a global DB that has not yet been saved in the PROFInet interface editor:

1. There are two options:
   - If the PROFInet interface editor has not yet started, start it from SIMATIC Manager (see above).
   - If the PROFInet interface editor has started, select the **File > Open...** menu command.

     The "New/Open PROFInet interface" dialog opens in both cases.
2. In the left-hand window, select the station and the device to which the PROFInet interface is to be assigned. If necessary, click on the "Browse" button to look up the path of the required STEP 7 project.
3. Check the "Display all interface-compliant DBs and FBs" check box. The global DBs without PROFInet properties now also appear in the right-hand part of the dialog box.
4. Make sure that the "Open" check box is checked. If the "New" box is checked, you will not be able to select a block.
5. Highlight the block and click on the "OK" button.

   If the "Define PROFInet interface type" dialog box opens, check the "PROFInet interface DB" check box and click on the "OK" button. Result: The block is opened in the PROFInet interface editor.

   Any existing entries can be found in the "Unassigned" section, and should be moved to one of the other sections "PN_Input", PN_Output" or "S7_Variable" as required.
6. Edit the block.
7. Save the block using the **File > Save** menu command.

Note

When it is saved in the PROFInet interface editor, the global DB is assigned the property PROFInet interface (active), and thus becomes the interface DB for the future PROFInet component.

Possible errors while opening interface DBs

If the interface DB was first edited in another editor, errors may occur when the block is opened in the PROFInet interface editor, particularly if block attributes were modified, added or deleted.

Incorrect entries cannot be assigned to a section while opening. Such entries are handled as follows:

- The incorrect entry is removed, and all subsequent entries for the data block are moved to the "Unassigned" section.
- The incorrect entry is moved to the "Unassigned" section, together with all subsequent entries for the data block.

In both cases, an error message is displayed and you will have to move the connectors from the "Unassigned" section to the relevant sections.

1.4.3.3 Declaring simple data types

Requirement

The block must have been opened in the PROFInet interface editor.

Procedure

1. Select a section from the variable overview.
2. Enter the name of the connector in the "Name" column of the variable detail view.
3. Press Enter to confirm your input and insert another blank line into the variable declaration.

You can now enter another name on the new line.

The default settings in the "Data type", "Address" and "Initial value" columns may be changed if necessary.

Hint: Rapid entry

Press Alt + Insert to quickly insert several variables one after another.

Syntax check

A syntax check is carried out after every entry. Any errors are displayed in red. For example, the initial value of a variable must correspond to the specified data type. You do not have to correct these errors immediately you can continue editing and make any corrections later.

1.4.3.4 Declaring connectors with the data type ARRAY

Requirement

The block must have been opened in the PROFInet interface editor.

Procedure

1. In the "Data type" column, click on the selection box and select the data type ARRAY.
2. Then enter the required dimensions in the same column, with upper and lower limit, plus the type of element, e.g. ARRAY [1..10] of Bool.
   Attention: There must be a space before and after "of".
3. If you wish to preassign certain values to the individual elements in the box, enter the desired values in the "Initial value" column.

Examples for entering initial values

Data type	Initial value	Explanation
ARRAY[1..14] of Int	1234	The initial value 1234 is only assigned to the first ARRAY element. All other elements in the field receive the initial value 0 (default for Int)
ARRAY[1..14] of Int	1234, 56, 78, 90	The initial values 1234, 56, 78, 90 are assigned in this order to the first four ARRAY elements. The remaining elements are given the value 0 (default for Int).
ARRAY[1..14] of Int	14 (9876)	The initial value 9876 is assigned to all 14 ARRAY elements.
ARRAY[1..14] of Int	15 (4711)	Error: More initial values were assigned than there are elements.

Rules

Only one-dimensional arrays may be declared.

Only arrays of simple data types may be declared. Arrays of arrays or structures are not permitted.

1.4.3.5 Declaring connectors with the data type STRUCT

Requirement

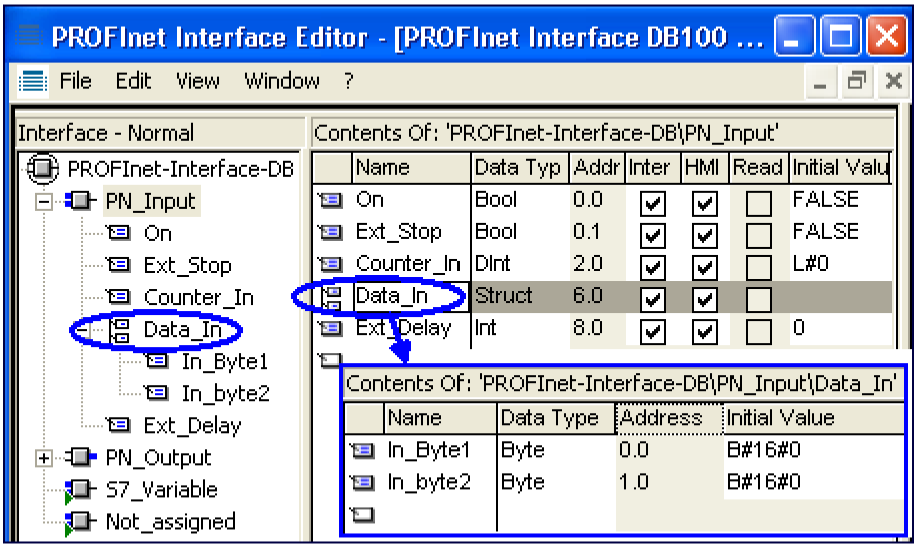
The block must have been opened in the PROFInet interface editor.

Procedure

1. In the "Data type" column, click on the selection box and select the STRUCT entry. An element of the type STRUCT is created.
2. Select the element from the variable overview and double-click. The structure is opened and its content appears in the detail window.
3. You can now insert any number of variables into the structure.

Example

The interface DB in the diagram below has an input of the type Struct.



Contents Of: 'PROFInet-Interface-DB\PN_Input'

Name	Data Typ	Addr	Inter	HMI	Read	Initial Valu
On	Bool	0.0	☑	☑	□	FALSE
Ext_Stop	Bool	0.1	☑	☑	□	FALSE
Counter_In	DInt	2.0	☑	☑	□	L#0
Data_In	Struct	6.0	☑	☑	□	
Ext_Delay	Int	8.0	☑	☑	□	0

Contents Of: 'PROFInet-Interface-DB\PN_Input\Data_In'

Name	Data Type	Address	Initial Value
In_Byte1	Byte	0.0	B#16#0
In_byte2	Byte	1.0	B#16#0

Figure 1-7 Interface DB with connectors of the type STRUCT

Rule

Only structures of simple data types may be declared. Structures of arrays or structures are not permitted.

1.4.4 Changing PROFInet interfaces

1.4.4.1 Modifying connectors

This section describes the possible changes that you can make to the connectors of an open PROFInet interface DB.

Notice

When you modify connectors, it is also possible to change the addresses of variables in the data block, so you will then have to adapt the program accordingly.

Recommendation:

Use symbolic names for the variables in the interface DB. In this way, you will make sure that the interface DB is independent of the physical addresses.

To do this, make the following setting in SIMATIC Manager: In the object properties for the block folder, "Address Priority" tab, check the "Symbol has priority for all accesses" check box.

The procedure for copying and moving connectors is as follows:

You can copy or move connectors anywhere, both within a section and between two sections.

Using the mouse:

- To move the connector, highlight it, hold down the left mouse button and drag the connector to the desired destination.
- To copy the connector, highlight it, hold down the left mouse button, press the CTRL button and drag the connector to the desired destination.

Using menu commands:

1. Click on the variable that you wish to copy.
2. Select the **Edit > Copy** or **Edit > Cut** menu command.
3. Open the declaration section in which you wish to save the copied or cut variable.
4. Select the **Edit > Paste** menu command.

Note

When you copy a connector, a number is automatically appended to the name, thus "variable" will become "variable_1", for example.

To delete a connector:

1. Highlight the connector that you wish to delete.
2. Select the **Edit > Delete** menu command or press the Delete key.

To change the data type

1. Click on the selection box in the "Data type" column. The selection depends on the current block type and which section is selected.
2. Select one of the displayed data types.
3. Overwrite the default initial value, if necessary.

Response in the "Initial value"column

If the "Initial value" column still contained the default setting for the old data type, the default for the new data type is automatically entered here.

If you have already entered an initial value that differs from the default, then the value you entered is retained. This is displayed in red if it is unsuitable for the new data type.

To change the initial value

Select the "Initial value" column from the variable detail view and enter a value. The initial value must conform to the specified data type.

Note

There are no specific format rules for your input. If your input is clear, then the program will correct it to conform to the IEC standard.

Invalid initial values are identified in red.

To enter a comment

Highlight a variable and enter a descriptive text in the "Comment" column. The comment may be up to 80 characters long.

1.4.4.2 Displaying properties

You can display the properties of the following object types:

- Entire PROFInet interface
- Section
- Slot
- Connector

Procedure

1. Click on the required icon in the tree structure in the variable overview.
2. Click the right mouse button and select the **Object Properties** menu command from the context menu.

1.4.4.3 Looking up and resetting PROFInet properties

Looking up PROFInet properties

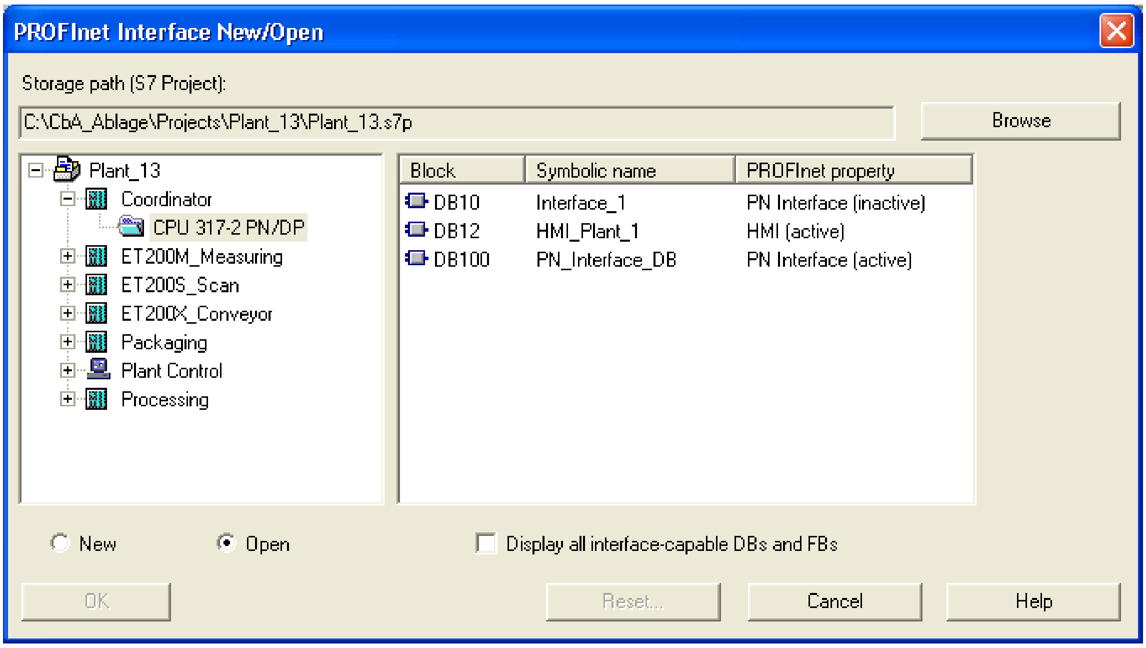
The block folder for the STEP 7 basic project may contain several interface DBs:

- just one with the property "PN interface (active)" that is taken into account when the PROFInet component is created and
- optionally, one or more with the property "PN interface (inactive)". These blocks also contain declarations for connectors, but they are ignored when the PROFInet component is created.

The PROFInet property of the PROFInet interface can be identified from the name that appears in the first line of the variable overview. You can also look up the PROFInet property from the object properties of the PROFInet interface.

The PROFInet properties are assigned to the blocks in the form of attributes.

Example: Displaying PROFInet properties



PROFInet Interface New/Open

Storage path (S7 Project):

C:\CbA_Ablage\Projects\Plant_13\Plant_13.s7p

- Plant_13
  - Coordinator
    - CPU 317-2 PN/DP
  - ET200M_Measuring
  - ET200S_Scan
  - ET200X_Conveyor
  - Packaging
  - Plant Control
  - Processing

Block	Symbolic name	PROFInet property
DB10	Interface_1	PN Interface (inactive)
DB12	HMI_Plant_1	HMI (active)
DB100	PN_Interface_DB	PN Interface (active)

○ New ● Open □ Display all interface-capable DBs and FBs

Figure 1-8 Displaying PROFInet properties in the "Open / New" dialog box

Automatic assignment of PROFInet properties

When you save a global DB in the PROFInet interface editor, the DB automatically receives the PROFInet property "PN interface (active).

Resetting PROFInet properties

There are two options for resetting PROFInet properties:

- Deactivate – the structure as a PROFInet interface with the connectors assigned to the PN_Input, PN_Output and S7_Variable sections is retained, but the block is ignored when the PROFInet component is created.

  The PROFInet properties can be reactivated by opening and saving the block in the PROFInet interface editor.

- Delete – the assignment of connectors to the PN_Input, PN_Output and S7_Variable sections is lost.

Requirement

The block whose PROFInet properties you wish to activate or deactivate must not be open in the PROFInet interface editor.

To deactivate or delete the PROFInet properties of an interface DB:

1. Select the **File > Open** menu command in the PROFInet interface editor. The "New/Open PROFInet interface" dialog box opens.
2. In the left-hand window of the dialog box, select the station and the device to which the block is assigned.

   Once you have highlighted a device, the blocks that can be opened for this device are displayed in the right-hand window.
3. Make sure that the "Open" box is checked. If the "New" box is checked, you will not be able to select a block.
4. Highlight the required block and click on the "Reset" button.
5. In the "Reset PROFInet Properties" dialog box:
   - Check the "Deactivate PROFInet interface block" check box to deactivate the PROFInet properties of the block or
   - Check the "Delete all PROFInet properties of the block" check box to delete all the PROFInet properties of the block.
6. Confirm by clicking on the "OK" button.

The change to the block appears in the "PROFInet property" column in the right-hand window of the "New/Open PROFInet interface" dialog box. If you have deleted all the PROFInet properties of the block, the block will only be displayed if you have checked the "Display all interface-compliant DBs and FBs" check box.

1.4.5 Special PROFInet interfaces

1.4.5.1 PROFInet DB for DP slaves with fixed functionality

Introduction

DP slaves with fixed functionality do not have their own user program. Thus, the interface DB only contains the interface definition. For the PROFInet component, you must create an interface DB for the DP slave and save it to the block folder of the S7 program assigned to the DP master station.

Assignment to the device

Several DP slaves may be configured in a STEP 7 basic project for PROFIBUS devices with fixed functionality. A PROFInet interface DB may be assigned to every DP slave from which a PROFInet component is to be created.

Structure of the interface DB for DP slaves with fixed functionality (e.g. standard slaves)

For interface DBs for PROFIBUS devices with fixed functionality, the PN_Input and PN_Output sections are divided into slots. A slot corresponds to the slot in an input module or output module of the DP slave.

The following applies to the structure of the interface DB:

- The output signals from the DP interface of the DP master are mapped onto slots in the PN_Input section. The inputs of the technological function are defined in this section.
- The input signals for the DP interface of the DP master are mapped onto slots in the PN_Output section. The outputs of the technological function are defined in this section.

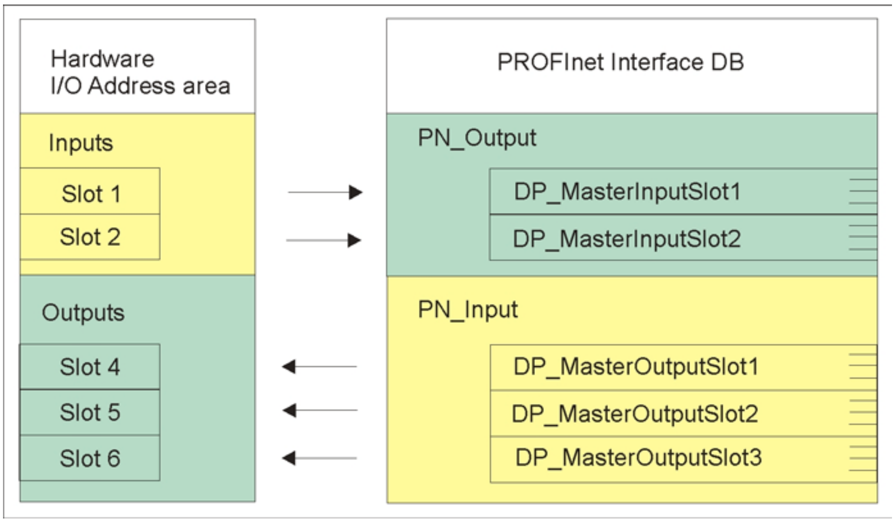


Figure 1-9 Structure of the interface DB for DP slaves with fixed functionality

Note

A connector may not be declared beyond slot limits.

Data length of the connectors

For DP slaves with fixed functionality, the maximum permitted data length for a connector is exactly the same as the consistent length of the slot.

Several connectors may be declared within a slot, although the data length of a connector cannot be greater than the data length of the slot for which it is defined.

The maximum permitted data lengths and consistent lengths are specified in the object properties for the slot.

Note

We recommend that you do not use off-the-shelf data blocks for the PROFInet interface for a DP slave with fixed functionality. Instead you should create a new block for the device in the interface editor. The input and output modules of the DP slave are automatically mapped onto slots in the structure of the interface DB.

When you save the interface DB, the permitted data lengths of the connectors are checked, thus preventing run-time errors within the plant because consistent lengths are exceeded.

Consistent length of the transferred data

The consistent length of the transferred data is displayed in the slot properties in the PROFInet interface editor. Depending on the type of module, the consistent length may be configured in HW Config.

Unless otherwise specified, the consistent length is generally:

- byte (8 bit) per digital input or output
- word (16 bit) per analog input or output

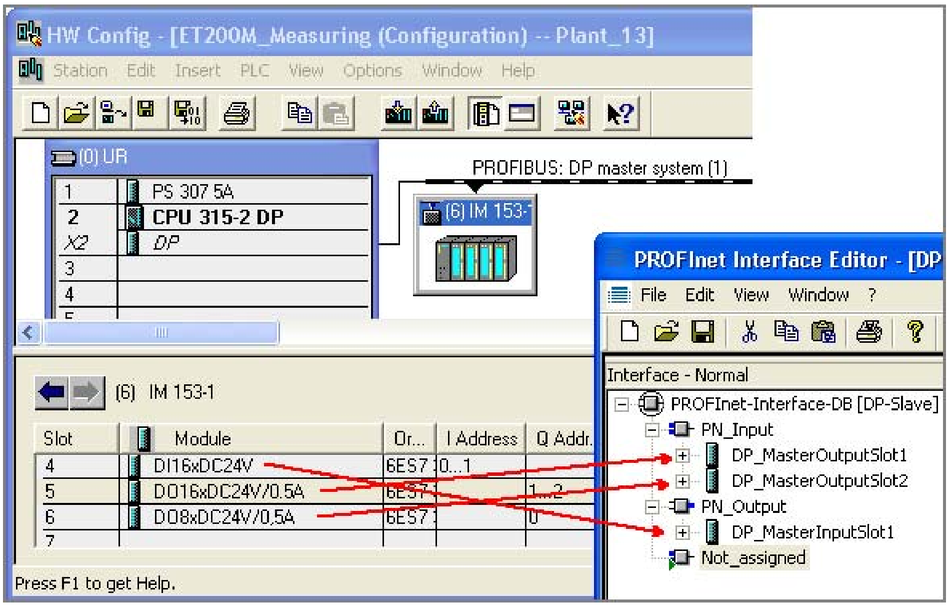
This may be different for some modules. Detailed information can be found in the technical data for the module concerned.

Unique connector names

The extension that is automatically added to the connector name, e.g. when you copy declarations, only ensures that the name is unique within a slot. The connector name is not checked for uniqueness within the entire interface DB until you save the interface DB.

Example: Interface DB for a PROFIBUS device, DP slave with fixed functionality, e.g. standard slave

The PROFInet component is created from just one ET 200M with IM 153-1. The following diagram illustrates the hardware configuration and how it is mapped onto the structure of the PROFInet interface.


HW Config - [ET200M_Measuring (Configuration) -- Plant_13]
Station Edit Insert PLC View Options Window Help
(0) UR
PROFIBUS: DP master system (1)
(6) IM 153-
(6) IM 153-1
Press F1 to get Help.
PROFInet Interface Editor - [DP
File Edit View Window ?
Interface - Normal
PROFInet-Interface-DB [DP-Slave]
PN_Input
DP_MasterOutputSlot1
DP_MasterOutputSlot2
PN_Output
DP_MasterInputSlot1
Not_assigned

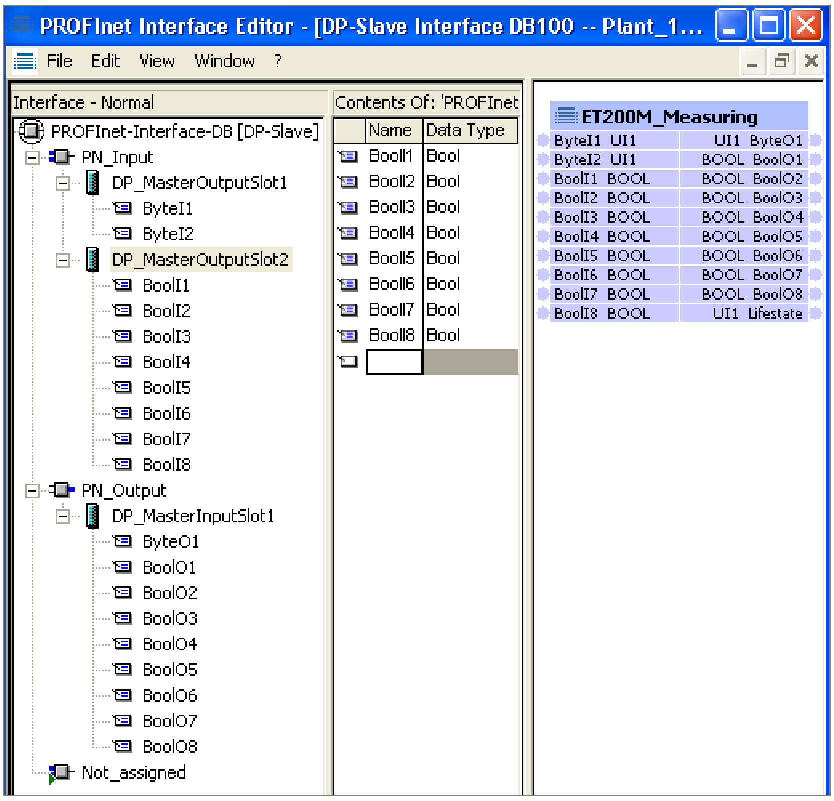
1	PS 307 5A
2	CPU 315-2 DP
X2	DP
3	
4	

Slot	Module	Or...	I Address	Q Addr.
4	DI16xDC24V	6ES7	0...1	
5	DO16xDC24V/0.5A	6ES7		1...2
6	DO8xDC24V/0.5A	6ES7		0
7				

Figure 1-10 DP slave ET 200M with IM 153, hardware configuration

Interface DB of the PROFInet component for IM153-1:

- the input module is mapped onto DPMasterInputSlot1 in the "PN_Output" section,
- the 2 output modules are mapped onto DPMasterOutputSlot1 and 2 in the "PN_Input" section.



PROFInet Interface Editor - [DP-Slave Interface DB100 -- Plant_1...

Interface - Normal

```
PROFInet-Interface-DB [DP-Slave]
  PN_Input
    DP_MasterOutputSlot1
      ByteI1
      ByteI2
    DP_MasterOutputSlot2
      BoolI1
      BoolI2
      BoolI3
      BoolI4
      BoolI5
      BoolI6
      BoolI7
      BoolI8
  PN_Output
    DP_MasterInputSlot1
      ByteO1
      BoolO1
      BoolO2
      BoolO3
      BoolO4
      BoolO5
      BoolO6
      BoolO7
      BoolO8
  Not_assigned
```

Contents Of: 'PROFInet

Name	Data Type
BoolI1	Bool
BoolI2	Bool
BoolI3	Bool
BoolI4	Bool
BoolI5	Bool
BoolI6	Bool
BoolI7	Bool
BoolI8	Bool

ET200M_Measuring

ByteI1	UI1	UI1	ByteO1
ByteI2	UI1	BOOL	BoolO1
BoolI1	BOOL	BOOL	BoolO2
BoolI2	BOOL	BOOL	BoolO3
BoolI3	BOOL	BOOL	BoolO4
BoolI4	BOOL	BOOL	BoolO5
BoolI5	BOOL	BOOL	BoolO6
BoolI6	BOOL	BOOL	BoolO7
BoolI7	BOOL	BOOL	BoolO8
BoolI8	BOOL	UI1	Lifestate

Figure 1-11 PROFInet interface for IM 153-1

In this example, the technological function contains 10 inputs and 10 outputs (including the lifestate output).

1.4.5.2 Creating internal HMI interface DBs

Additional internal HMI interface DBs are possible in addition to the interface DB. These contain only non-interconnectable connectors that can be accessed via OPC using S7 protocols. An HMI interface DB may be assigned to either a PROFInet device or to a programmable PROFIBUS device.

Types of block used

You can create an HMI interface DB from the following block types:

- Instance data block of a function block
- Global data block (global DB)

You will always have to edit an existing block in the PROFInet interface editor. In the first case this is an FM, in the second it is an existing global data block. The declarations must have been added using a different editor. In the PROFInet interface editor, only the necessary PROFInet properties are added to the existing declarations.

Requirements

The global DB or FB must have already been created using an editor, e.g. ladder diagram/sequential function chart/statement list.

The variables declared in the block may only have the S7 data types permitted for PROFInet, otherwise the block cannot be saved in the PROFInet interface editor.

To create an HMI interface DB:

1. Select the **File > Open** menu command in the PROFInet interface editor.

   The "New/Open PROFInet interface" dialog box opens.
2. In the left-hand window, select the station and the device to which the HMI interface is to be assigned. If necessary, click on the "Browse" button to find the STEP 7 project.

   Once you have highlighted a device, the blocks that can be opened for this device are displayed in the right-hand window.
3. Make sure that the "Open" box is checked. If the "New" box is checked, you will not be able to select a block.
4. Make sure that you have activated the "Display all interface-compliant DBs and FBs" option. This will ensure that all DBs and FBs in the block folder needed to create an HMI interface will be displayed.
5. Highlight the required block and click on the "OK" button.
6. Only if you selected a global DB: Select the "HMI interface DB" check box from the "Define PROFInet interface type" dialog box and click on the "OK" button.

   Result: The block is opened in the interface editor.
7. Change the properties of the connectors as required (see below)
8. Save the block using the **File > Save** menu command.
9. Only if the edited block is a function block:
   Create an instance DB for this FB.

Result: The global or instance DB is created to extend the PROFInet interface and is taken into account the next time a PROFInet component is created.

Note

When you save a global DB or FB in the PROFInet interface editor, the block automatically receives the PROFInet property "HMI(active)". The instance DB of such an FB inherits the PROFInet property when it is created.

Editing the HMI interface DB or FB in the PROFInet interface editor

You cannot delete, add, copy or move connectors. You can only change the following properties of the connectors:

- HMI/MES – If this property is active, the variable is written to the OPC symbol file in SIMATIC iMap and may be used for HMI/MES via OPC.
- Read-Only – If this property is active, the variable is identified as read only in the OPC symbol file. If the property is not active, the variable is identified as read and write in the OPC symbol file.

  The "Read-Only" property can only be activated if "HMI/MES" is also active. The two options are always activated together.

  Deactivate the option in the "Read-Only" column for the variables with read/write access.

Deactivating and activating PROFInet properties

When you deactivate the PROFInet properties of an HMI interface FB, all the associated instance DBs are automatically deactivated as well, and are thus ignored when the PROFInet component is created.

When you reactivate the FB (by saving it in the PROFInet interface editor), an existing instance DB will not automatically be activated. You will have to create a new instance DB for the activated FB that inherits the activated PROFInet properties and is taken into account when the PROFInet component is created.

1.5 Creating an S7 program

Points to note when creating the S7 program...

The S7 program for the future PROFInet component with programmable functionality is created in STEP 7 as usual. Take the following point into consideration:

- Access to the inputs and outputs declared in the interface DB must be programmed in the S7 program.
- The necessary PROFInet copy blocks (from the PROFInet system library) must be located in the block folder for the S7 program. Make sure that these blocks are called correctly in the S7 program.

  Instructions can be found in the online help for the blocks (function key F1).
- The necessary interrupt and error OBs must also be programmed.

Detailed descriptions...

Note

Detailed descriptions of how to program PROFInet components can be found in the "Commissioning Systems, Tutorial" manual.

1.6 Creating PROFInet components

1.6.1 Creating PROFInet components in SIMATIC Manager

Introduction

If you wish to create PROFInet components from SIMATIC programmable controllers, you must do this using the STEP 7 SIMATIC Manager.

Requirement

The STEP 7 basic project must be complete, i.e.:

- The hardware configuration and parameter settings for the modules are complete.
- The S7 program and the PROFInet interface DB have been created.
- The STEP 7 basic project has been tested.
- You have the documentation for the future PROFInet component.
- You have the files with the icons for representing the entire PROFInet component, the technological function and the associated device. If you do not have your own icon files for the component to be created, you can use the files supplied with the software, which are provided as defaults.

Rule

You can create a PROFInet component from the station in the STEP 7 basic project as follows:

- from the entire station (CPU, intelligent DP slave) or
- from one of the DP slaves with fixed functionality.

To create a PROFInet component in SIMATIC Manager

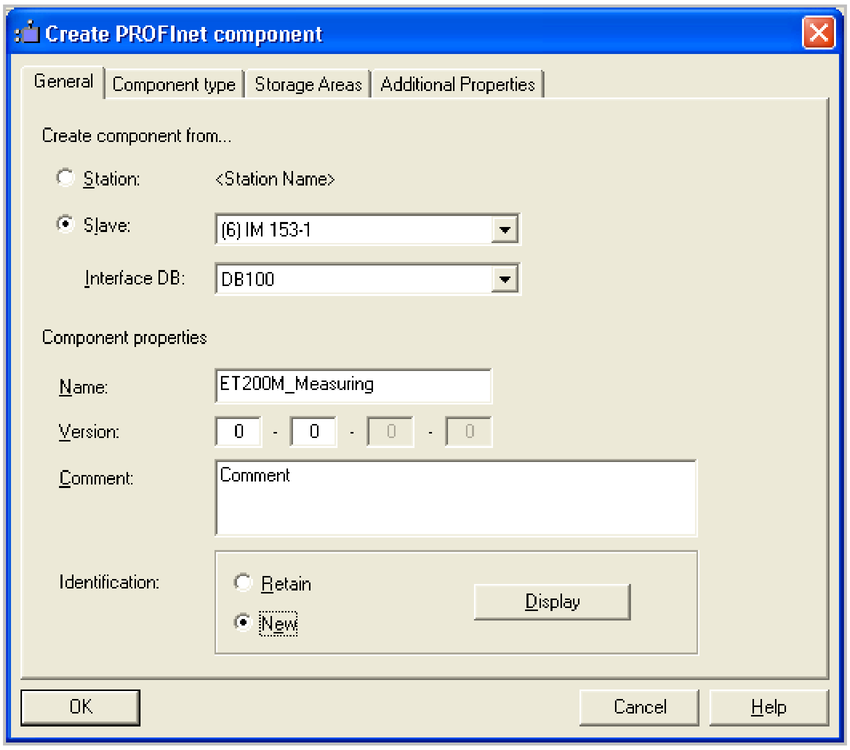
1. In your STEP 7 basic project, select the station from which the component is to be created.
2. Select the Create PROFInet component menu command. There are two options:
   - **Edit > Create PROFInet component** menu command or
   - Context menu (right mouse button) **Create PROFInet component**.

   Result: The "Create PROFInet component" dialog box opens.
3. Enter the properties of the PROFInet component to be created (see below).
4. Click on "OK" or "Cancel" to end your entries. If you click on "OK", the PROFInet component is created. If you click on "Cancel", the dialog box closes without applying the changes in STEP 7.

"General" tab

This tab is used to define:

- which elements should be used to create the PROFInet component, e.g. the entire station (a DP master and one or more local DP slaves, if applicable) or an individual DP slave,
- Properties, such as name and version, and identifications of the component.



Create PROFInet component

General | Component type | Storage Areas | Additional Properties

Create component from...

○ Station: <Station Name>

● Slave: (6) IM 153-1

Interface DB: DB100

Component properties

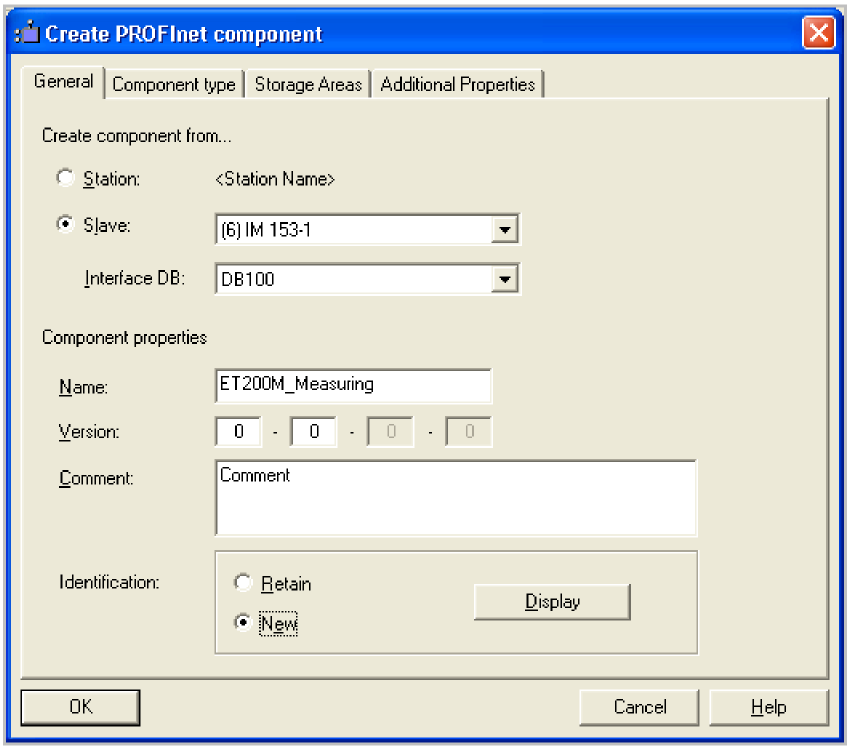
Name: ET200M_Measuring

Version: 0 - 0 - 0 - 0

Comment: Comment

Identification: ○ Retain ● New

Figure 1-12 Creating the PROFInet component from a station



Create component from...

○ Station: <Station Name>

● Slave: (6) IM 153-1

Interface DB: DB100

Component properties

Name: ET200M_Measuring

Version: 0 . 0 . 0 . 0

Comment: Comment

Identification: ○ Retain ● New

Figure 1-13 Creating the PROFInet component from a DP slave

Name

The name must not exceed 24 characters and must start with a letter.

Version number

The version contains four series of numbers, separated by dots.

The first two places of the version number may be any numbers (up to 3 digits). The last two places are increased automatically.

When you enter a version number, it must always be higher than the previous number.

Identification of the PROFInet component

The identification of the PROFInet component is a unique identifier conforming to the COM standard. It consists of the class ID, the interface name and the interface ID. In SIMATIC iMap, PROFInet components are identified by their class ID and version number.

The options are as follows:

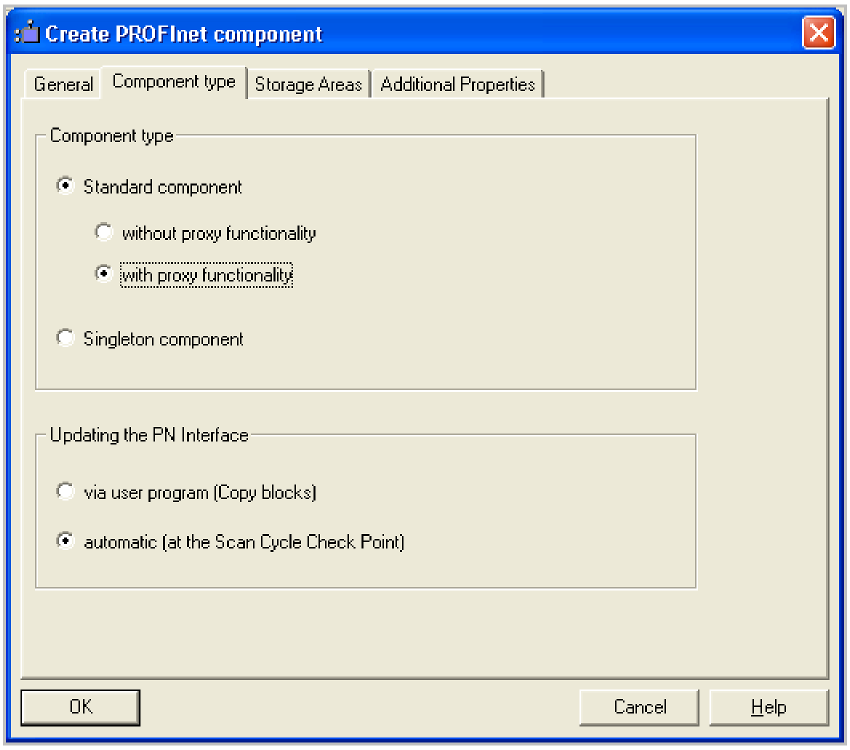
- If you select "New", a new identification is generated automatically. You will have to assign a new identification if:
  - you are creating the PROFInet component for the first time, or
  - you have changed the name of the PROFInet component.
- If you select the "Retain" option, the default identification is retained, and the PROFInet component is only given a new version number.

We recommend that you accept the default values ("Retain" option).

Note

The interface ID is automatically changed whenever the component interface is changed.

"Component type" tab



Create PROFInet component

General | Component type | Storage Areas | Additional Properties

Component type

● Standard component

○ without proxy functionality

● with proxy functionality

○ Singleton component

Updating the PN Interface

○ via user program (Copy blocks)

● automatic (at the Scan Cycle Check Point)

Figure 1-14 Component type

This tab is used to define the following:

- Which component type is to be created from the hardware configuration.
- how the PROFInet interface is to be updated at runtime.

Selecting the component type

The options are as follows:

- Select the "Standard component" option to create a simple PROFInet component. We distinguish between PROFInet components without or with proxy functionality.
- Select the "Singleton" option to create a Singleton component. This component type can be created for any hardware configurations with an Ethernet connection. Additional information can be found under "Singleton components".

Standard component without proxy functionality

Check this check box to create the PROFInet component from one of the following hardware configurations:

- From a SIMATIC station with a PROFInet-compliant device without proxy functionality, e.g. a CPU 31x with a CP 343-1 PN. The created PROFInet component has one PROFInet device.
- From a SIMATIC station with a CPU that is configured as a DP slave and contains the necessary blocks in the S7 program. The created PROFInet component has one PROFIBUS device with programmable functionality (intelligent DP slave).
- From a DP slave only, and not from the entire station. The created PROFInet component has one PROFIBUS device (DP slave) with fixed functionality.

Standard component with proxy functionality

Check this check box if the created PROFInet component is to contain a PROFInet device with proxy functionality. This component type can only be created for hardware configurations with the following features:

- The station contains a PROFInet-compliant device with proxy functionality, e.g. a CPU 317-2PN/DP or WinLC PN.
- The station has a configured DP master system.

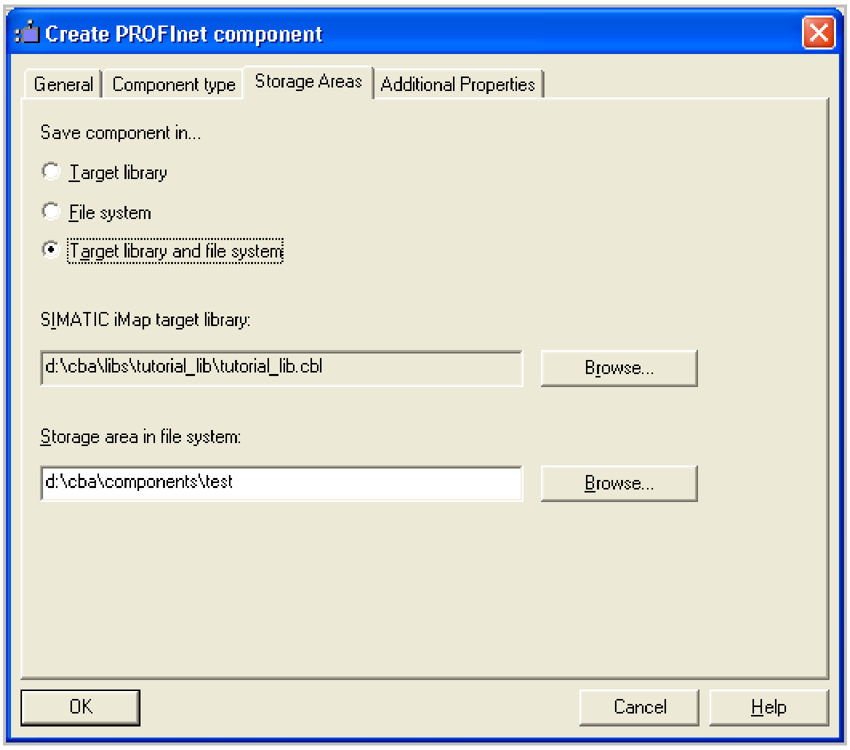
Updating the PROFInet interface

The PROFInet interface is updated at runtime as follows:

- "Using the user program (copy blocks)"– This option can only be selected for programmable PROFInet devices.

  The copy blocks must be copied from the PROFInet system library to the block folder for the STEP 7 basic project. Information on the copy blocks can be found in the descriptions of the individual devices.
- "Automatically (at the scan cycle checkpoint)"– (i.e. at the time the process image is updated). This is the default setting that is supported for most PROFInet devices and for all PROFIBUS devices (DP slaves) with fixed functionality.

"Storage Areas" tab



Save component in...

○ Target library

○ File system

● Target library and file system

SIMATIC iMap target library:

d:\cba\libs\tutorial_lib\tutorial_lib.cbl

Storage area in file system:

d:\cba\components\test

Figure 1-15 Storage areas

With this tab, you define whether component created is to be saved to the file system and/or imported directly into a library.

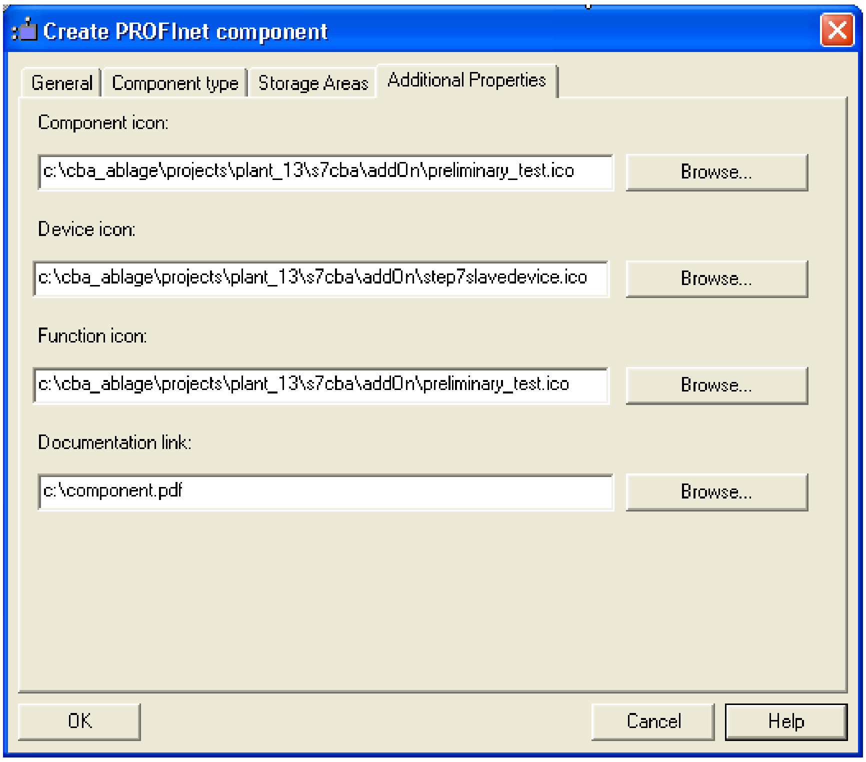
Note

If the PROFInet component is to be imported into a library, the library must have been created using SIMATIC iMap.

"Additional Properties" tab

This tab is used to define:

- which icons of the technological function should be assigned to the PROFInet device and the PROFInet component and
- the path name or Internet address under which the documentation for the PROFInet component can be found.



Component icon: c:\cba_ablage\projects\plant_13\s7cba\addOn\preliminary_test.ico

Device icon: c:\cba_ablage\projects\plant_13\s7cba\addOn\step7slavedevice.ico

Function icon: c:\cba_ablage\projects\plant_13\s7cba\addOn\preliminary_test.ico

Documentation link: c:\component.pdf

Figure 1-16 Additional Properties

Note

Documentation link

If you enter the documentation link as the URL, you must make sure that the documentation is actually accessible in the plant.

A documentation link can only reference a single file.

If an HTM file containing cross-references to other files is specified when the PROFInet component is created, only the specified HTM file is copied to the PROFInet component folder - not the referenced files.

Remedy:

We recommend that you use a different format, e.g. PDF or CHM, if you require complex documents.

Result

The PROFInet component is stored as a directory at the specified storage area - in the file system or directly in the library, or at both locations. The directory name contains the names, the identification (class ID) and the version number of the PROFInet component. The directory contains the following:

- The XML file containing the description of the PROFInet component and, in particular, the technological function. The system searches for this XML file when importing to a library.
- The icon files
- The archived STEP 7 component project that contains a copy of the station or the DP slave from the basic project.
- The file for the documentation link, if a path name in the file system was specified as the documentation link.

1.6.2 Version numbers of PROFInet components

Application

Every PROFInet component has a version number that forms a unique identifier for the component together with the identification (class ID). This ensures that a newly created PROFInet component does not accidentally overwrite an earlier issue of the component.

Structure of the Version Number

The version contains four series of numbers, separated by dots.

- The first two places of the version number may be assigned as required (up to 3 digits) e.g. 01.04
- The last two places of the build version cannot be modified and are incremented automatically.

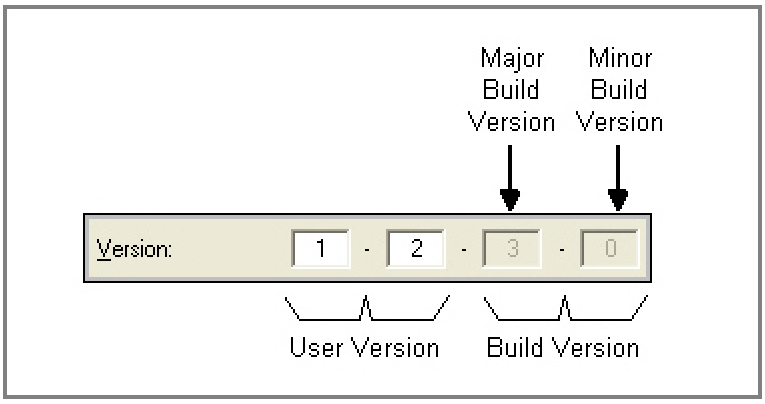


Figure 1-17 Structure of the Version Number

Rules

When you enter a version number, it must always be higher than the previous number.

You can change the first two parts or leave them unchanged. If you change them, you are asked whether the last two parts should be reset to 0 or incremented.

Automatically incrementing the version number

The last two parts of the version number are automatically incremented as follows:

- The last part (minor build version) is incremented by one whenever a new version of the PROFInet component is successfully created.
- The penultimate part (major build version) is incremented by one whenever the interface DB is changed. The last part is automatically set to 0.

This applies to both the PROFInet interface DB and to any internal HMI interface DB.

If changes are made to the PROFInet interface DB, a new interface ID is also assigned automatically.

Note

These changes are only applied if the PROFInet component was created successfully. If creation was cancelled, the version number and the interface ID (if applicable) remain unchanged.

1.7 Importing PROFInet components to a Library

Basic procedure

A PROFInet component must be located in a library before it can be inserted into SIMATIC iMap. You can insert a PROFInet component into a library in the following ways:

- When you create the PROFInet component in the device-specific programming and configuration tool, e.g. SIMATIC Manager
- In SIMATIC iMap, by importing to a library (described in the SIMATIC iMap Help topics under "Configuring a plant, Importing PROFInet components“).

Requirement

If you create a PROFInet component and wish to import it, then the library must have been created using SIMATIC iMap.

Importing the PROFInet component into a Library after Creation

Proceed as follows:

1. Create and test your basic project.
2. In SIMATIC Manager, select the station for the component to be created.
3. From the context menu, select **Create PROFInet component**.
4. In the "Create PROFInet component" dialog box, "Storage areas" tab, select one of the following options:
   - "Target library" or
   - "Target library and file system“.
5. Accept the default library in the "SIMATIC iMap target library" box or click on the "Browse" button to look for another library in the file system.

   In the "Select" box, navigate to the library folder, highlight the file with the extension .cbl and click on OK to confirm.
6. On the other tabs of the dialog box, enter any settings you wish to make and click on OK to confirm.

Result: The PROFInet component is created in the specified folder in the library (and in the file system, if necessary).

1.8 Modifying PROFInet components

Modification options

The options for modifying existing PROFInet components are as follows:

- Edit the original STEP 7 basic project in SIMATIC Manager.

  or

- Revise a PROFInet component directly in the library in SIMATIC iMap. This operation is described in the SIMATIC iMap documentation under "Configuring plants, Working with modified PROFInet components".

In both cases, you will have to modify the associated STEP 7 project in SIMATIC Manager and then recreate the PROFInet component.

Procedure for editing a STEP 7 basic project

To modify an existing PROFInet component:

1. Open the STEP 7 basic project in SIMATIC Manager.
2. Make your changes to the program or hardware configuration.
3. If necessary, compile the program changes and save the STEP 7 project (and hardware configuration, if necessary).
4. Create a new PROFInet component from the modified station or DP slave. Select the **Edit > Create PROFInet component** menu command.

   Note: The dialog box contain the default properties that were used when the last component was created.
5. If necessary, modify the properties of the PROFInet component, such as the name, version or storage area.
6. Click on "OK" to confirm.
7. If you have only specified the file system (and no library) as the storage area, import the created PROFInet component into a SIMATIC iMap library.

You can then use the modified PROFInet component in a SIMATIC iMap project.

SIMATIC Devices as PROFInet components 2

2.1 SIMATIC devices - Overview

SIMATIC devices that can be used

You can create PROFInet components from the following SIMATIC devices:

- Central processing units with central and distributed peripherals as PROFInet devices or PROFIBUS devices with programmable functionality
- PCs with Windows Logic Controller for PROFInet (WinLC PN) as a PROFInet or PROFIBUS device with programmable functionality
- "Intelligent" DP slaves (I slaves) as PROFIBUS devices with programmable functionality
- DP slaves as PROFIBUS devices with fixed functionality

Bus ports

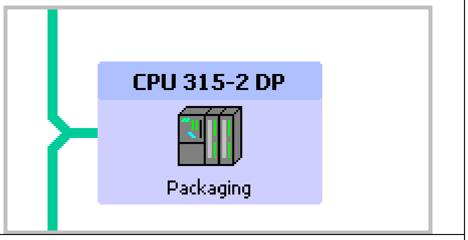
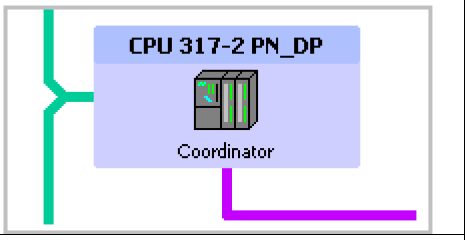
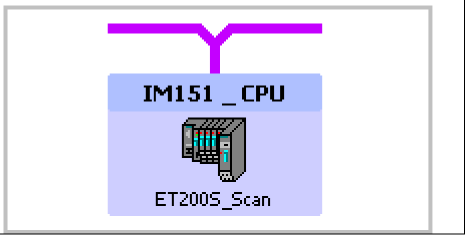
The device's bus ports are shown in the SIMATIC iMap net view.

PROFInet devices can have the following bus ports:

- Industrial Ethernet – always
- PROFIBUS DP as a DP master with proxy functionality for PROFIBUS devices – for some PROFInet devices only

PROFIBUS devices used as DP slaves have only one PROFIBUS connector.

Table 2-1 Representation of devices in SIMATIC iMap

Bus ports	Representation in SIMATIC iMap
PROFInet device on the Industrial Ethernet	
PROFInet device on the Industrial Ethernet and DP master on the PROFIBUS (proxy functionality)	
PROFIBUS device DP slave on the PROFIBUS	

The internal networking of the devices via a local PROFIBUS or MPI bus, for example, cannot be seen in SIMATIC iMap.

Representation: Devices and functions

A PROFInet component is represented as follows in SIMATIC iMap:

- As a PROFInet component – unit consisting of device and technological function – in the library,
- As a PROFInet or PROFIBUS device in the net view and
- As a technological function (if present) in the plant view.

Functionality

We differentiate between PROFInet components:

- with programmable functionality

  These components have their own S7 program which can be downloaded from SIMATIC iMap to the module (e.g. to a CPU or WinLC PN).
- with fixed functionality

  These components do not have their own S7 program, e.g. ET 200M or IE/PB Link.

Features of STEP 7 projects

PROFInet components for SIMATIC devices are created from STEP 7 projects in SIMATIC Manager. The following features are important:

- Station

  The device from which the PROFInet component is created may be located in a separate station in SIMATIC Manager (e.g. a CPU or WinLC PN), or it may be configured as a DP slave with fixed functionality (e.g. ET 200M) on any DP master system. In the second case, the DP master is not part of the PROFInet component to be created.

- Local PROFIBUS

  Programmable controllers, such as CPUs or WinLC PN, may be configured as DP masters with a local PROFIBUS DP master system. In this case, the entire local DP master system is part of the PROFInet component, but the local DP slaves still do not appear in SIMATIC iMap.

- Local MPI bus

  CPUs may be configured with a local MPI bus in order to integrate an HMI device. In this case, the HMI device is part of the PROFInet component, but neither the HMI device nor the MPI bus can be seen in SIMATIC iMap.

- Blocks required

  The block folder of the S7 program must contain the following blocks which, in turn, contain the interface declaration for the technological function.

  - Interface DB - for all PROFInet components that contain a technological function
  - Additional blocks, e.g. copy blocks for intelligent DP slaves

Information on the blocks can be found in the descriptions of the individual device types.

Note

A PROFInet component can have either a local PROFIBUS or a local MPI connector. Both connectors are not permitted in a single component.

Summary

Table 2-2 Features of SIMATIC devices as PROFInet components in SIMATIC iMap

Device type	Bus ports	Representation in SIMATIC iMap	Functionality
CPU 317-2 PN/DP	Ethernet PROFIBUS master	1 PROFInet device 1 technological function	Programmable
Win LC PN	Ethernet PROFIBUS master (optional)	1 PROFInet device 1 technological function	Programmable
S7-300 CPU with CP 343-1 PN	Ethernet	1 PROFInet device 1 technological function	Programmable
S7-300 CPU as an intelligent DP slave	PROFIBUS slave	1 PROFIBUS device 1 technological function	Programmable
ET 200X , BM 147 /CPU ET 200S, IM 151 /CPU	PROFIBUS slave	1 PROFIBUS device 1 technological function	Programmable
Standard slaves and S7 slaves as DP slaves with fixed functionality	PROFIBUS slave	1 PROFIBUS device 1 technological function	Fixed

Table 2-3 Features of PROFInet components in STEP 7

Device type	Station	Local PROFIBUS is possible	Local MPI bus is possible	Blocks required
CPU 317-2 PN/DP	Separate station	Yes	Yes	Interface DB
Win LC PN	Separate station	Yes	Yes	Interface DB
S7-300 CPU with CP 343-1 PN	Separate station	Yes	Yes	interface DB additional FB
S7-300 CPU as an intelligent DP slave	Separate station	Yes	Yes	Interface DB additional DB, FCs
ET 200X , BM 147 /CPU ET 200S, IM 151 /CPU	Separate station	Yes	No	Interface DB, additional DBs, FCs
Standard slaves and S7 slaves as DP slaves with fixed functionality	DP slave on a station with any DP master	No	No	Interface DB

2.2 Configurations for PROFInet components

Device types in Component based Automation

We differentiate between the following types of device downstream of the bus connectors:

- PROFInet devices
- PROFIBUS devices

PROFInet device

A PROFInet device always has an Ethernet connector. A PROFInet device may also have a PROFIBUS connector as a proxy master for PROFIBUS devices.

Possible configurations for a PROFInet device

The PROFInet device may be one of the following SIMATIC devices:

- CPU 31x-2PN/DP

  Central processing unit with integral PROFInet interface. The CPU 31x-2PN/DP is a PROFInet device with proxy functionality. Local DP slaves and/or a local HMI device may be connected to the PROFIBUS.

- S7-300 CPU with a PROFInet-compliant communication processor (PROFInet CP), such as the CP 343-1 PN.

  The PROFInet device of an S7-300 central processing unit does not have proxy functionality, but it can run a local PROFIBUS as the master or a local MPI bus.

- Windows logic controller WinLC PN

  The WinLC PN is a PROFInet device with proxy functionality. Local DP slaves and/or a local HMI device may be connected to the PROFIBUS.

- Network transition IE/PB Link

  The IE/PB Link is a PROFInet device with proxy functionality, but no local DP slaves may be connected to it.

- Further hardware configurations with Ethernet connectors as singleton components (see "Singleton components").

Locally connected devices of a PROFInet device

The locally connected devices of a PROFInet device cannot be seen in SIMATIC iMap and are not involved in the PROFInet communication. The PROFIBUS addresses of the local DP slaves are regarded as assigned in SIMATIC iMap.

The following devices may be connected to the local PROFIBUS:

- Intelligent DP slaves
- DP standard slaves
- DP-S7 slaves
- DP-ASI links
- DP-PA links
- an HMI device, e.g. OP

The intelligent DP slaves on the local PROFIBUS can, in turn, each run a local PROFIBUS as a master. Only the following devices may be connected to this local PROFIBUS, however:

- DP standard slaves
- DP-S7 slaves
- DP-ASI links
- DP-PA links

Note

When a program is downloaded to the PROFInet device, the programs for the associated local intelligent DP slaves are automatically downloaded as well, if the local DP slaves are connected.

Only one HMI device may be connected to the local MPI bus of a PROFInet device. The program can only be downloaded to a local HMI device via ProTool/Pro CS (see HMI devices in PROFInet components).

Connecting S7 slaves

Note

Not every DP master can run S7 slaves. This information can be found in the relevant manuals.

Example: Maximum configuration for a PROFInet device with a local PROFIBUS

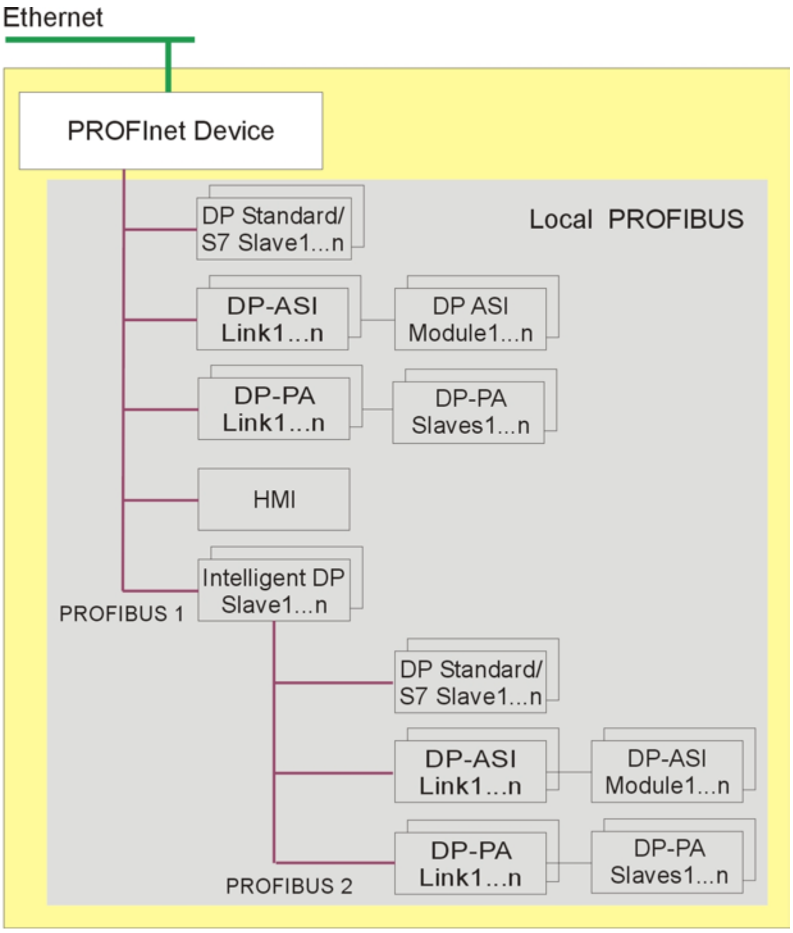


Figure 2-1 Maximum configuration for a PROFInet device with a local PROFIBUS

PROFIBUS device

A PROFIBUS device has just one PROFIBUS connector and is always a slave. It does not participate directly in PROFInet communication and is integrated via a PROFInet device with proxy functionality.

PROFIBUS devices are subdivided into:

- PROFIBUS devices with programmable functionality running their own S7 programs, i.e. "intelligent" DP slaves (I slaves).
- PROFIBUS devices with fixed functionality - DP standard slaves or S7 slaves without their own program.

Possible configuration for a PROFIBUS device

- A **PROFIBUS device with programmable functionality** may be one of the following SIMATIC devices:
  - ET 200X with BM 147/CPU
  - ET 200S with IM 151CPU
  - CPU 31x-2DP, CPU 318-2 or CPU 31xC-2DP as an intelligent DP slave
  - Complete C7 6xy DP devices as intelligent DP slaves

    A PROFIBUS device with programmable functionality always has an integral PROFIBUS interface that is used to couple it to the PROFIBUS master with proxy functionality in SIMATIC iMap.
- A **PROFIBUS device with fixed functionality** may be one of the following devices:
  - DP standard slave
  - DP-S7 slave
  - DP-ASI link

Locally connected devices of a PROFIBUS device

Central processing units as intelligent DP slaves may have up to one further interface:

- to a local PROFIBUS or
- to a local MPI bus.

The locally connected devices of a PROFInet device cannot be seen in SIMATIC iMap.

Only the following devices may be connected to the local PROFIBUS:

- DP standard slaves
- DP-S7 slaves
- DP-ASI links
- DP-PA links
- an HMI device, e.g. OP

Only one HMI device may be connected to the local MPI bus of a PROFIBUS device.

Examples: Configurations for PROFIBUS devices

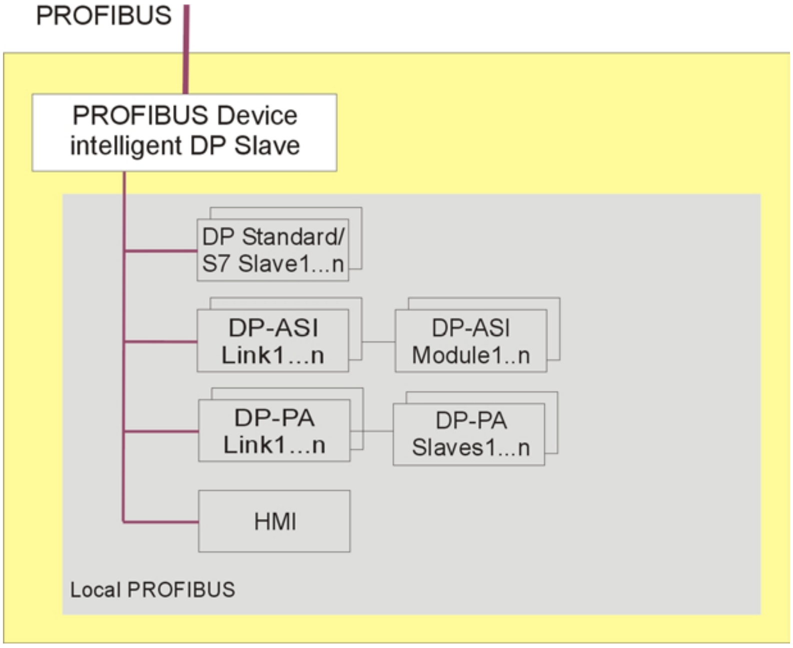


Figure 2-2 Maximum configuration for an intelligent PROFIBUS device with local PROFIBUS

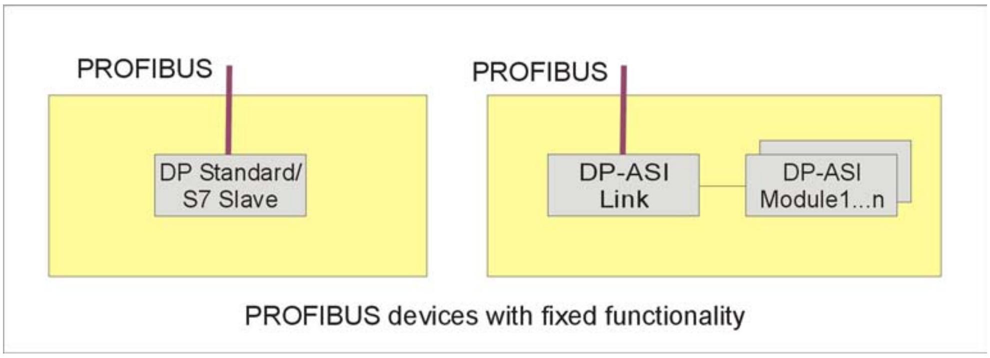


Figure 2-3 PROFIBUS devices with fixed functionality, hardware configurations

Rule: No more than 2 network interfaces per device

The CPU in the station of the PROFInet component may have up to 2 configured network interfaces, only one of which is the DP master. The following combinations are possible:

- For PROFInet devices - always one Ethernet connector and
  - one PROFIBUS connector as the master with proxy functionality (optionally with local DP slaves on this DP master system)
  - one local PROFIBUS connector as the master
  - one local MPI connector
- For PROFIBUS devices - always one PROFIBUS connector as the slave and
  - one local PROFIBUS connector as the master
  - one local MPI connector

Illegal configurations

The following configurations are not permitted in PROFInet components:

- Intelligent DP slaves with local PROFIBUS with further intelligent DP slaves connected.
- Intelligent DP slaves on the local PROFIBUS via PROFIBUS-CP

  If the local PROFIBUS of a CPU is structured via a PROFIBUS CP (e.g. CP 342-5), no local intelligent DP slaves may be connected to it.
- Combination of several stations or DP slaves without a DP master system for a PROFInet component
- Configuration of more than one DP master system within a station
- C7 with an integrated HMI device on the MPI bus as a local intelligent DP slave of a PROFInet device.
- More than one integrated HMI device.

2.3 Central Processing Units

2.3.1 Central processing units - Configuration options

Central processing units as PROFInet and PROFIBUS devices

A CPU from the SIMATIC S7-300 product family always creates a PROFInet component with programmable functionality. CPUs may be used as:

- PROFInet devices (with an Ethernet connector)
- Intelligent PROFIBUS devices (with a PROFIBUS connector as a DP slave)

The station forms a unit with a defined function, such as a machining station or coordinator. The PROFInet component generally consists of a technological function and the associated device.

Local bus connection: PROFIBUS or MPI

A local (internal) connector is possible:

- a local PROFIBUS for connecting local DP slaves, e.g. via the integral DP interface on the CPU 31x-2DP or
- one local MPI connector for an HMI device.

The locally connected devices, such as DP slaves or an HMI device, cannot be seen on the PROFInet component in SIMATIC iMap.

2.3.2 Network configurations for CPUs

The possible network configurations for different CPU types are described below:

- as a PROFInet device and
- as a PROFIBUS device (intelligent DP slave)

CPU 31x-2PN/DP

PROFInet components of the following device types may be created from a SIMATIC-300 station with a CPU 31x-2PN/DP:

- PROFInet device, optionally with proxy functionality. "Ethernet" operating mode must be configured for network interface X1.

  Internal networking:

  - Optionally either a local PROFIBUS if the "DP master" operating mode is configured for the integral network interface X2, or
  - a local MPI bus just for the connector of an HMI device if the "MPI" operating mode is configured for the integral network interface X2.
- PROFIBUS device (intelligent DP slave), if no operating mode is configured for the integral network interface X1 and the "DP slave" operating mode is configured for the integral network interface X2.

  Internal networking:

  - Either a local PROFIBUS with the CP 342-5DP or
  - a local MPI bus for connecting an HMI device only.

CPU 31x, CPU 31xC and CPU31x-2PtP

The components of a PROFInet device may be created from a SIMATIC-300 station with a CPU 31x or CPU 31xC and the CP 343-1 PN.

Internal networking:

- Either a local PROFIBUS with the CP 342-5DP or
- a local MPI bus for connecting an HMI device only.

Configuration as a PROFIBUS device (intelligent DP slave) is not possible.

CPU 31x-2DP and CPU 31xC-2DP

PROFInet components of the following device types may be created from a SIMATIC-300 station with a CPU 31x-2DP or CPU 31xC-2DP:

- PROFInet device - the CP 343-1 PN must be suitably configured.

  Internal networking:

  - Either a local PROFIBUS if the "DP master" operating mode is configured for the integral network interface X2, or
  - a local MPI bus for connecting an HMI device only.
- PROFIBUS device (intelligent DP slave), the "DP slave" operating mode is configured for the integral network interface X2.

  Internal networking:

  - Either a local PROFIBUS with the CP 342-5DP or
  - a local MPI bus for connecting an HMI device only.

CPU 31x-2

PROFInet components of the following device types may be created from a SIMATIC-300 station with a CPU 31x-2:

- CPU 31x-2 as a PROFInet device - the CP 343-1 PN must be suitably configured.

  Internal networking:

  - A local PROFIBUS is possible if the "DP master" operating mode is configured for the integral network interface X2. The integral network interface X1 may not then be networked.
  - The integral network interface X1 may also be programmed for MPI in order to configure an HMI device as part of the PROFInet component. The integral network interface X2 may not then be networked.
- CPU 31x-2 as a PROFIBUS device

  PROFIBUS device (intelligent DP slave), the "DP slave" operating mode is configured for the integral network interface X2. (The network interface X1 as a "DP slave" is not permitted).

  Internal networking:

  - A local PROFIBUS may be configured via the integral network interface X1. A PROFIBUS interface via a PROFIBUS-CP (e.g. CP 342-5DP) is not permitted.
  - A local MPI bus may be configured via the integral network interface X1. This can then be used to connect a local HMI device.

Complete C7-6xy or C7-6xy DP device

PROFInet components of the following device types may be created from a SIMATIC-300 station with a complete C7-6xy or C7-6xy DP device:

- PROFInet device - the CP 343-1 PN must be suitably configured.
- PROFIBUS device (intelligent DP slave) from the C7-6xy DP only, provided that the "DP slave" operating mode is configured for the integral network interface X2.

Internal networking:

- An HMI device must be configured as part of the PROFInet component via MPI.
- A local PROFIBUS **CANNOT** be configured.

2.3.3 Central processing units as PROFInet devices

Central processing units of the type CPU 31x-2PN/DP have an integral PROFInet interface and are thus PROFInet-compliant.

Other types of CPU have to be combined with a PROFInet-compliant communication processor (PROFInet-CP), such as the CP 343-1 PN.

Optionally, the SIMATIC 300 station may contain the associated central and distributed peripherals, see "Configurations for PROFInet components".

Configuration options

PROFInet components with an Ethernet connector may be created from the following S7-300 CPUs:

- CPU 31x-2PN/DP
- CPU 31x
- CPU 31x IFM
- CPU 31x-2DP
- CPU 31x-2
- CPU 614
- CPU 31x-C and CPU 31xC-2DP
- Complete C7-6xy or C7-6xy DP device

The CPU 31x-2PN/DP may optionally be configured as a PROFInet device with proxy functionality.

Updating the PROFInet interface for the CPU 317-2 PN/DP

When you create the PROFInet component for a CPU 317-2 PN/DP, you have two ways to update the PROFInet interface at runtime:

- Automatically at the scan cycle checkpoint, i.e. at the time the process image is refreshed.
- Via a user program using copy blocks (SFCs 112, 113 and 114).

  You will have to copy the copy blocks from the PROFInet system library to the block folder for the STEP 7 basic project and call them in the S7 program.

Block	Description
SFC 112	PN_IN – Updates the inputs of the interface DB of the PROFInet component
SFC 113	PN_OUT – Copies the data from the outputs of the interface DB to the CPU memory area
SFC 114	PN_DP – Updates the DP interconnections in the proxy system

In STEP 7 in, information on the blocks can be found the help on system functions / function blocks.

If a PROFInet CP is used...

please note the following points:

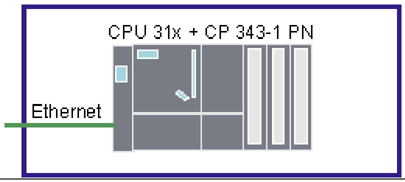
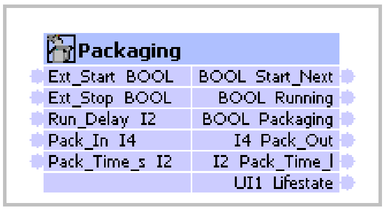
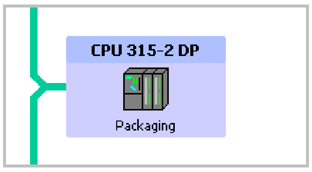
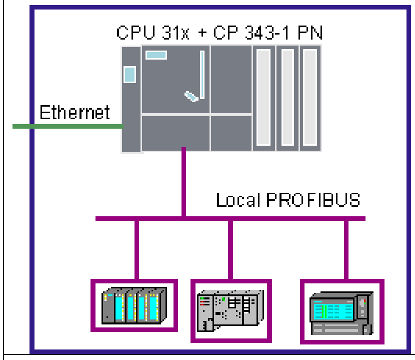
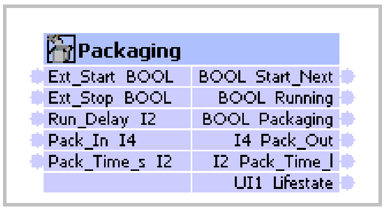
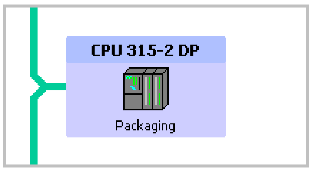
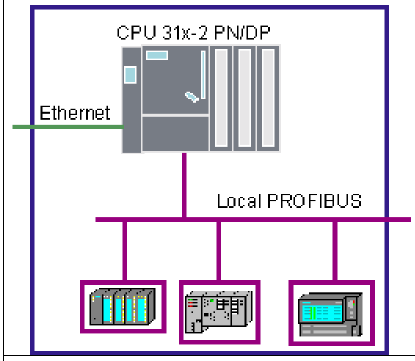
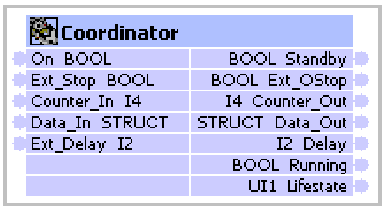
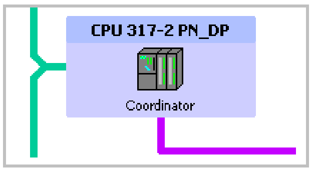
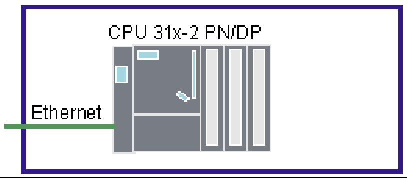
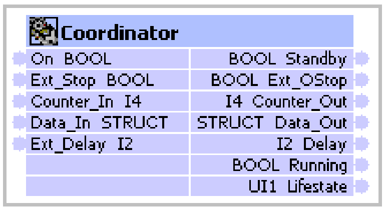
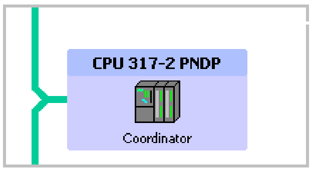
- **Hardware configuration:** Only one PROFInet CP may be configured in the central rack in a SIMATIC 300 station.
- **Configuration:** When you configure the Ethernet port, you must:
  - select the "IP address is used" option
  - deselect the "Set MAC address / Use ISO protocol" option.
- **Copy block in the S7 program:**

  The PN_InOut function block (FB88) must be called in the CPU program. This function block is a PROFInet copy block. It transfers the data from the interface DB to the PROFInet CP and from the PROFInet CP to the interface DB.

  The FB88 must be copied from the PROFInet system library to the block folder for the STEP 7 basic project.

Examples: Central processing units as PROFInet devices

Table 2-4 Configurations: Central processing units as PROFInet devices

Hardware configuration	Technological function	Device
CPU 315-2 DP with CP 343-1 PN  	 Packaging Ext_Start BOOL — BOOL Start_Next Ext_Stop BOOL — BOOL Running Run_Delay I2 — BOOL Packaging Pack_In I4 — I4 Pack_Out Pack_Time_s I2 — I2 Pack_Time_l UI1 Lifestate	
CPU 315-2 DP with CP 343-1 PN and with local PROFIBUS  	 Packaging Ext_Start BOOL — BOOL Start_Next Ext_Stop BOOL — BOOL Running Run_Delay I2 — BOOL Packaging Pack_In I4 — I4 Pack_Out Pack_Time_s I2 — I2 Pack_Time_l UI1 Lifestate	
CPU 317-2 PN/DP with proxy functionality and local PROFIBUS  	 Coordinator On BOOL — BOOL Standby Ext_Stop BOOL — BOOL Ext_OStop Counter_In I4 — I4 Counter_Out Data_In STRUCT — STRUCT Data_Out Ext_Delay I2 — I2 Delay BOOL Running UI1 Lifestate	
CPU 317-2 PN/DP without proxy functionality  	 Coordinator On BOOL — BOOL Standby Ext_Stop BOOL — BOOL Ext_OStop Counter_In I4 — I4 Counter_Out Data_In STRUCT — STRUCT Data_Out Ext_Delay I2 — I2 Delay BOOL Running UI1 Lifestate	

2.3.4 Central processing units as PROFIBUS devices

Configuration options

PROFInet components with a PROFIBUS connector as the DP slave may be created from the following S7-300 CPUs:

- CPU 31x-2DP
- CPU 31xC-2DP
- CPU 31x-2
- C7-6xy DP

Optionally, the SIMATIC 300 station may contain the associated central and distributed peripherals, see "Configurations for PROFInet components".

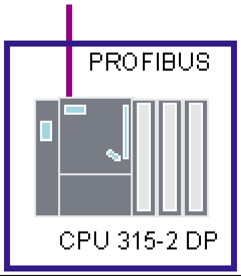
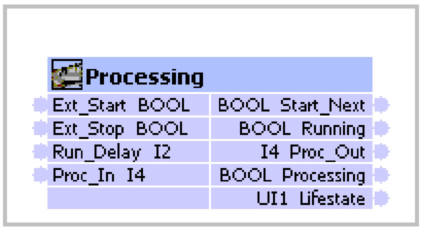
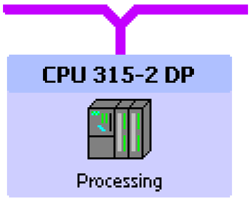
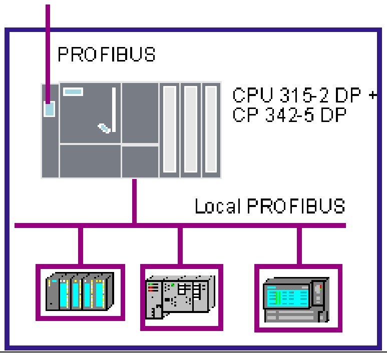
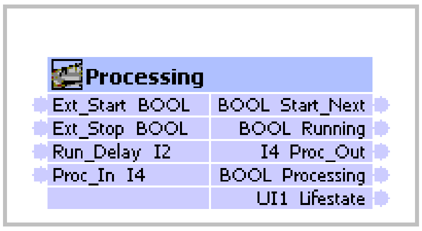
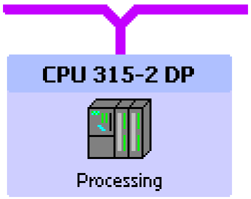
These modules all have at least one integral network connector that must be configured as a DP slave.

A local DP master system may also be configured as follows:

- For the CPU 31-2P, via the integral network connector X1. Network connector X2 must be configured as a DP slave. A PROFIBUS CP may not be used as a DP master.
- For a CPU 31x-2DP or CPU 31xC-2DP, via the CP 342-5 in the central rack.

Examples: Central processing units as PROFIBUS devices

Table 2-5 Configurations: Central processing units as PROFIBUS devices

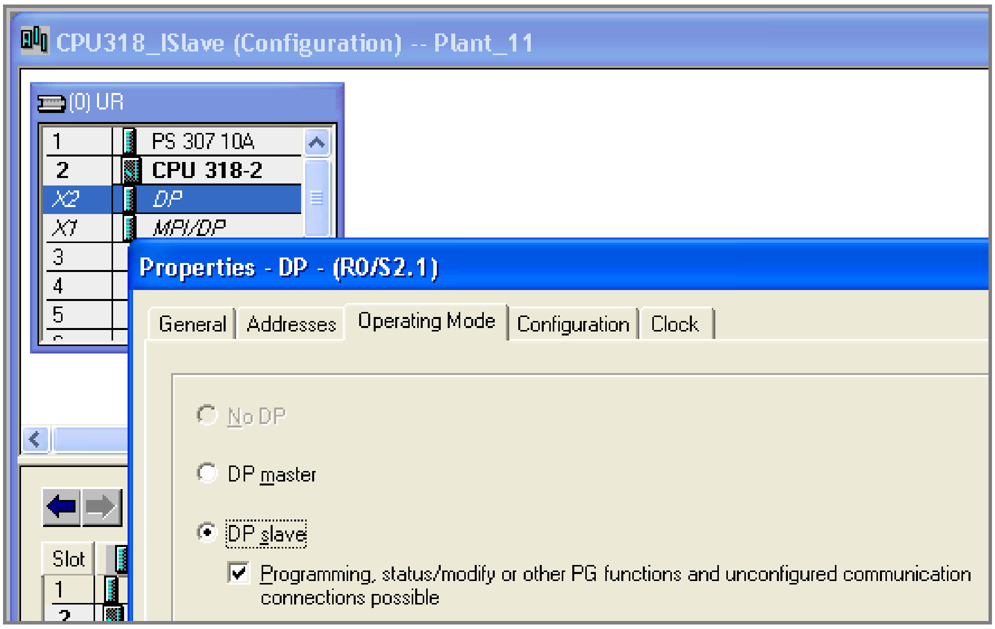
Hardware configuration	Technological function	Device
CPU 315-2DP  	 Processing Ext_Start BOOL · BOOL Start_Next Ext_Stop BOOL · BOOL Running Run_Delay I2 · I4 Proc_Out Proc_In I4 · BOOL Processing UI1 Lifestate	
CPU 315-2 DP with local PROFIBUS  	 Processing Ext_Start BOOL · BOOL Start_Next Ext_Stop BOOL · BOOL Running Run_Delay I2 · I4 Proc_Out Proc_In I4 · BOOL Processing UI1 Lifestate	

Special features

The CP 342-5 is not permitted in the expansion rack.

No intelligent DP slaves may be configured in the local DP master system.

When you configure a CPU as a DP intelligent slave, you must set "DP slave" mode in the DP interface properties in HW Config and select the "Programming... " option, if available (see diagram).


CPU318_ISlave (Configuration) -- Plant_11
(0) UR
Properties - DP - (R0/S2.1)
General
Addresses
Operating Mode
Configuration
Clock
Slot

1	PS 307 10A
2	CPU 318-2
X2	DP
X1	MPI/DP
3	
4	
5	

○ No DP

○ DP master

● DP slave

☑ Programming, status/modify or other PG functions and unconfigured communication connections possible

Figure 2-4 Configuring a CPU 318x-2 as an intelligent DP slave

2.4 WinLC PN

Introduction

The WinLC PN (Windows Logic Controller with PROFInet functionality) is a PROFInet-compliant device, and always forms a PROFInet component with programmable functionality.

Possible configurations

You can create a PROFInet component from a SIMATIC PC station that contains the following hardware:

- A WinLC PN with an Industrial Ethernet CP
- Local distributed peripherals, e.g. via CP 5613 or CP 5611
- An integral HMI device on the local PROFIBUS or MPI bus

The local DP slaves and the local HMI device cannot be seen on the PROFInet component in SIMATIC iMap.

The station forms a unit that performs a defined function, generally a central control function.

PROFInet functionality

The WinLC PN is a PROFInet device with proxy functionality (optional).

Bus ports

A PROFInet component created from a WinLC PN with the above setup has two bus ports:

- For an Industrial Ethernet and
- optionally on a PROFIBUS DP as a DP master with proxy functionality via CP 5613 or CP 5611.

Blocks required

The S7 program on the WinLC PN must contain the following organizational blocks in order to be able to respond to events:

- OB 82 diagnostic interrupt – e.g. when a DP slave is switched to STOP or RUN.
- OB 85 program sequence error - for errors that occur while updating the process image when a DP slave fails
- OB 86 rack failure - when the connector to a DP slave is interrupted

If these blocks are not present, the WinLC PN switches to STOP when an error occurs and can no longer provide its proxy functionality for the interconnected PROFIBUS devices. If the error OBs are present (even with no content), the WinLC PN continues to run in the event of an error. If necessary, a suitable response to the error must be programmed in each error OB.

Sample plant configuration with WinLC PN

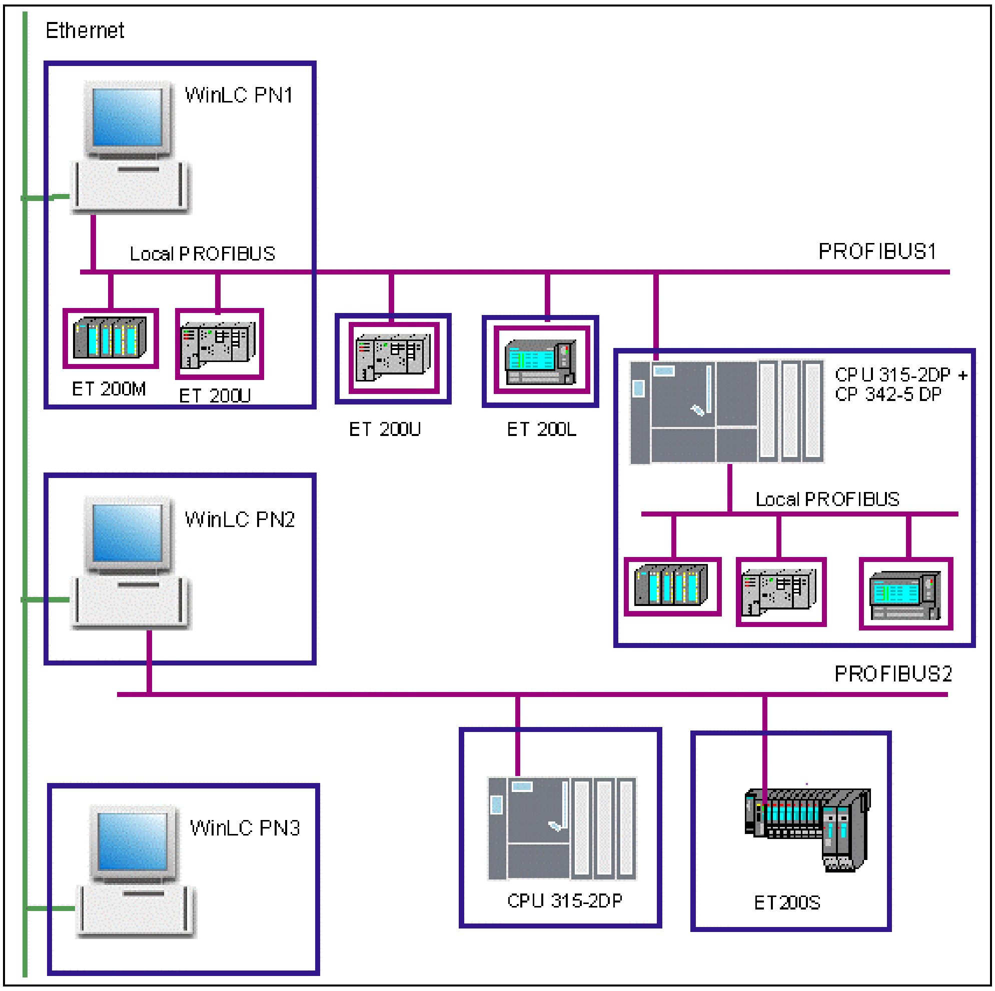
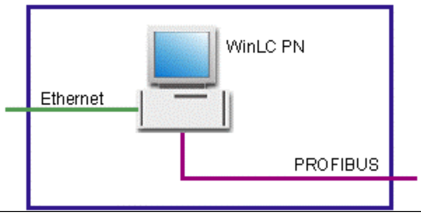
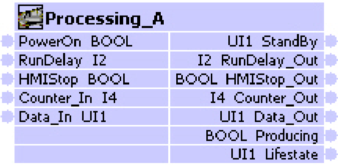
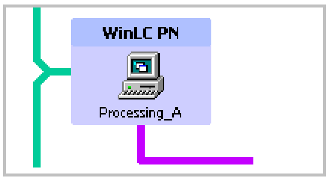
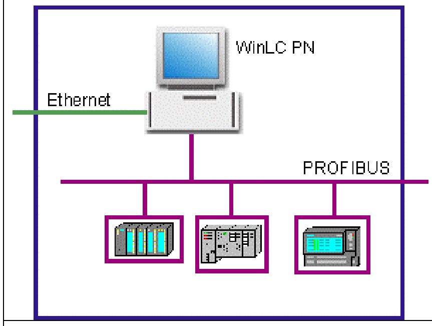
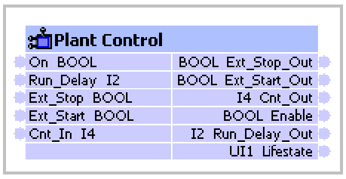
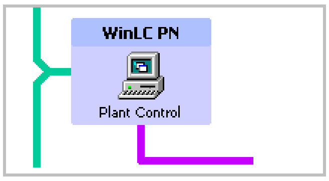
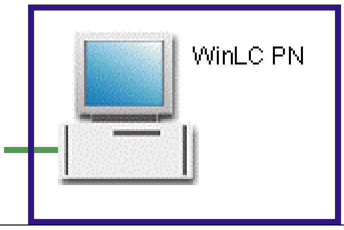
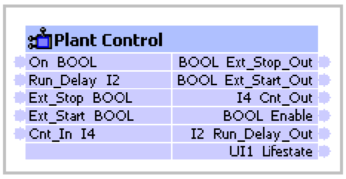
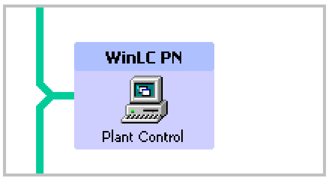


Figure 2-5 Sample plant with WinLC PN

Examples of PROFInet components

Table 2-6 Possible configurations with the WinLC PN

Hardware configuration	Technological function	Device
WinLC PN with proxy functionality  	 Processing_A PowerOn BOOL UI1 StandBy RunDelay I2 I2 RunDelay_Out HMIStop BOOL BOOL HMIStop_Out Counter_In I4 I4 Counter_Out Data_In UI1 UI1 Data_Out BOOL Producing UI1 Lifestate	
WinLC PN with proxy functionality and local DP master system  	 Plant Control On BOOL BOOL Ext_Stop_Out Run_Delay I2 BOOL Ext_Start_Out Ext_Stop BOOL I4 Cnt_Out Ext_Start BOOL BOOL Enable Cnt_In I4 I2 Run_Delay_Out UI1 Lifestate	
WinLC PN without proxy functionality  	 Plant Control On BOOL BOOL Ext_Stop_Out Run_Delay I2 BOOL Ext_Start_Out Ext_Stop BOOL I4 Cnt_Out Ext_Start BOOL BOOL Enable Cnt_In I4 I2 Run_Delay_Out UI1 Lifestate	

2.5 Distributed peripherals

2.5.1 Overview: Distributed peripheral devices as PROFInet components

Depending on the structure of the S7 program, we differentiate between:

- PROFIBUS devices with programmable functionality - intelligent DP slaves (I-slaves) with a CPU running a separate S7 program, such as the IM 151/CPU for ET 200S or BM 147/CPU for ET 200X. This type of device also includes CPUs that can be configured as DP slaves, e.g. CPU 315-2DP (see also the chapter entitled "Central Processing Units"). The program for such a device may be downloaded from SIMATIC iMap.
- PROFIBUS devices with fixed functionality - DP standard slaves or S7 slaves without their own program.

The representation in the SIMATIC iMap net view is the same for both types of device - they are shown as PROFIBUS devices with a PROFIBUS connector and acting as a DP slave. The plant view shows the technological function of the PROFInet component.

2.5.2 Intelligent DP slaves as PROFInet components

Hardware configurations

You can create PROFInet components from the following programmable distributed peripherals:

- ET 200X with BM 147/CPU
- ET 200S with IM 151/CPU
- CPU 31x-2DP, C7 6xy DP, CPU 31xC-2DP (see also the section entitled "Central Processing Units").

Optionally, the SIMATIC 300 station may contain the associated central and distributed peripherals, see "Configurations for PROFInet components".

Note

Direct exchange of data (cross-traffic) between PROFIBUS devices is not supported.

Exchange of data between the DP master and intelligent DP slaves

For an intelligent DP slave, the DP master accesses a transfer area in the address space of the CPU, rather than directly accessing the connected inputs/outputs of the intelligent DP slave. To do this, the input and output data for the PROFInet component, i.e. the data of the interface DB, must be transferred from the transfer area and to the transfer area of the intelligent DP slave at the user program runtime.

To exchange data between the DP master and an intelligent DP slave, the copy blocks PN_IN (FC10) and PN_OUT (FC11) are needed in the user program.

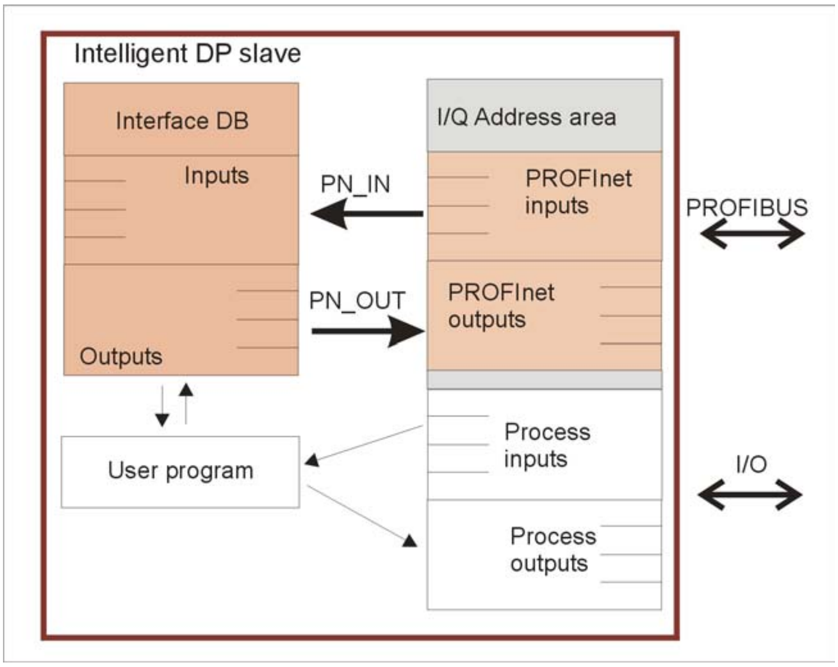


Figure 2-6 Exchange of data between the interface DB and the I/O area of the intelligent DP slave

Blocks required in the S7 program of an intelligent DP slave

The following off-the-shelf blocks are available in the PROFInet system library:

Block	Description
FC10	PNPN_IN - Copies the data from the transfer area of the DP master with proxy functionality to the input of the interface DB Attribute: CBA_islave_copy_fc and value: in The block PN_IN (FC10) should be called at the start of OB1 to ensure that it is the current data that is transferred to the interface DB.
FC11	PN_OUT – Copies the outputs of the interface DB to the transfer area of the DP master with proxy functionality. Attribute: CBA_islave_copy_fc and value: out The block PN_OUT (FC11) should be called at the end of OB1 to ensure that the data in the transfer area is not overwritten.
DB2	Data block with the attribute: CBA_islave_copy_db and value: true The DB contains data for the PN_IN and PN_OUT functions and must not be modified.

The numbers of these blocks can be modified as required.

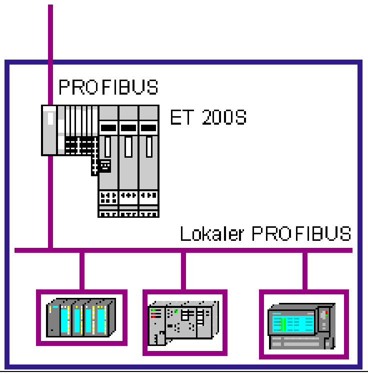
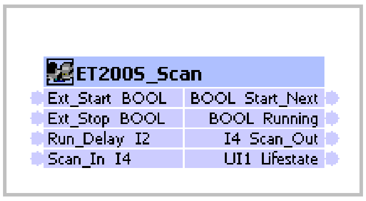
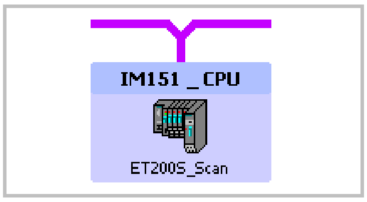
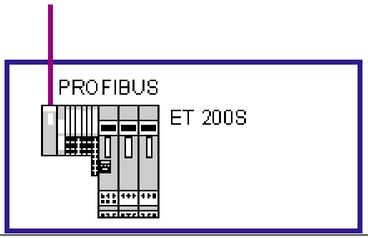
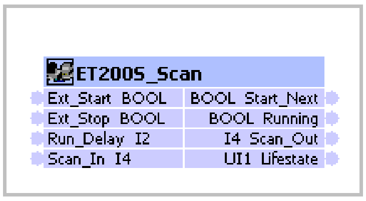
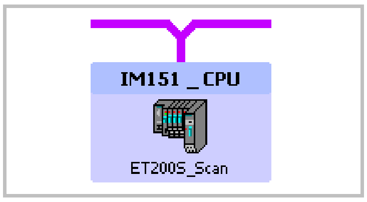
Organizational blocks required

The S7 program of an intelligent DP slave must contain the following organizational blocks in order to be able to respond to events:

- OB 82: Diagnostic interrupt, e.g. when the DP master is switched to STOP or RUN.
- OB 86: Rack failure, e.g. when the connection to the DP master is interrupted.

Example: PROFIBUS device with programmable functionality

Table 2-7 PROFInet component from an ET 200S with IM 151/CPU

Configuration	Technological function	Device
ET 200S with local PROFIBUS  	 ET200S_Scan Ext_Start BOOL BOOL Start_Next Ext_Stop BOOL BOOL Running Run_Delay I2 I4 Scan_Out Scan_In I4 UI1 Lifestate	
ET 200S  	 ET200S_Scan Ext_Start BOOL BOOL Start_Next Ext_Stop BOOL BOOL Running Run_Delay I2 I4 Scan_Out Scan_In I4 UI1 Lifestate	

2.5.3 DP slaves with fixed functionality as PROFInet components

A DP slave with fixed functionality, such as a compact (ET 200B) or modular (ET 200M) distributed peripheral device, must be configured in the station of the STEP 7 basic project on any DP master in order to create a PROFInet component from it. In this case, the DP master is not part of the PROFInet component.

Devices supported

You can create PROFInet components with fixed functionality from the following devices:

- SIMATIC distributed peripherals (standard slaves and S7 slaves)
- DP/ASI link with DP/ASI modules
- standard slaves that are described by a GSD file

See "Configurations for PROFInet components".

Special features of DP slaves with fixed functionality

The PROFInet component of a compact or modular DP slave has fixed functionality, i.e. there is no separate S7 program running on the device. The interface for the technological function is defined and set on the device. The names of the inputs and outputs of this interface are defined in the interface DB located in the station's block folder. The DP master accesses the distributed inputs/outputs of the DP slave directly, so the interface DB is an image of the process inputs and outputs (see " PROFInet DB for DP slaves with fixed functionality“).

The hardware configurations of standard and S7 slaves with fixed functionality that contain programmable modules such as FMs and CPs cannot be used to create PROFInet components.

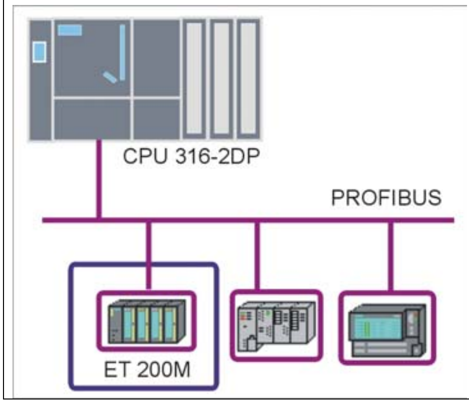
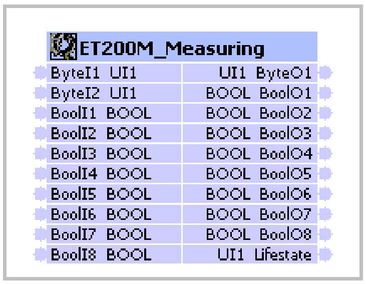
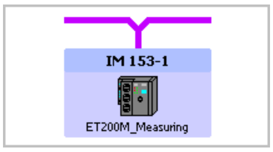
The ET 200L-SC must be configured using GSD files before a PROFInet component can be created from it.

Note

Direct exchange of data (cross-traffic) between PROFIBUS devices is not supported.

Example

Table 2-8 PROFInet components from an ET 200 M

Hardware configuration	Technological function	Device
	ET200M_Measuring ByteI1 UI1 UI1 ByteO1 ByteI2 UI1 BOOL BoolO1 BoolI1 BOOL BOOL BoolO2 BoolI2 BOOL BOOL BoolO3 BoolI3 BOOL BOOL BoolO4 BoolI4 BOOL BOOL BoolO5 BoolI5 BOOL BOOL BoolO6 BoolI6 BOOL BOOL BoolO7 BoolI7 BOOL BOOL BoolO8 BoolI8 BOOL UI1 Lifestate	

2.5.4 Special case: DP/DP coupler

Using the DP/DP coupler in SIMATIC iMap

The DP/DP coupler is used to exchange data between two PROFIBUS-DP masters, one of which may be a non PROFInet-compliant device, e.g. an IM308-C or a CPU 4xx.

It is thus possible to integrate PROFIBUS networks into SIMATIC iMap projects, even without PROFInet-compliant devices. Such networks cannot be shown in SIMATIC iMap, however.

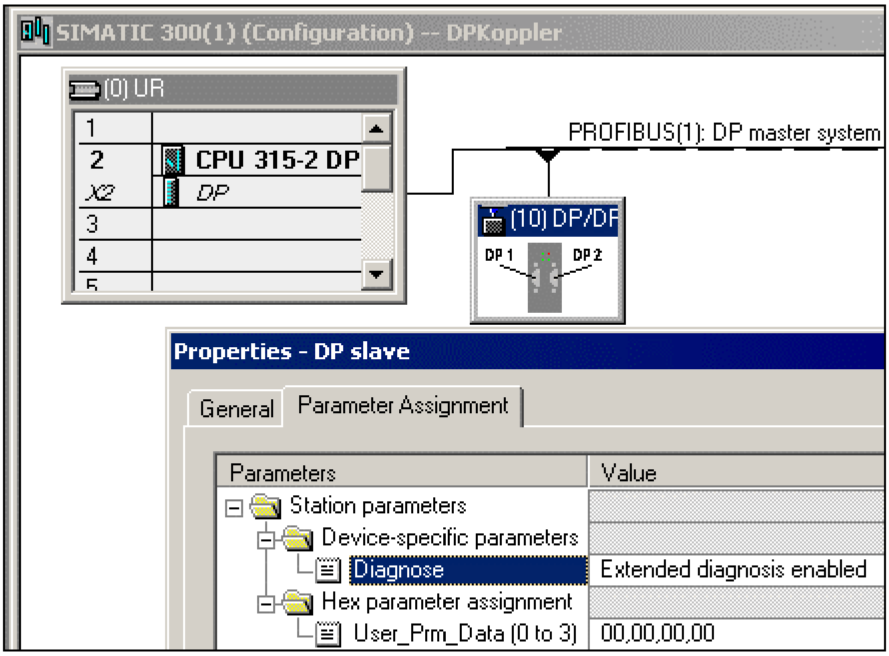
Configuration

In STEP 7, the DP/DP coupler is configured as a modular slave in both PROFIBUS networks using a GSD file. The desired length of the input and output data is set during configuration. The output data from one of the slaves is used as the input data for the other slave, and vice versa,

Example:
PROFIBUS network with a PROFInet-compliant master: 2-byte inputs, 4-byte outputs
PROFIBUS network without a PROFInet-compliant master: 4-byte inputs, 2-byte outputs

Note

When you configure the DP/DP coupler for the PROFInet-compliant master, make sure that the option "Extended diagnostics enabled" is set in the Properties dialog (see diagram). This means that the diagnostics in SIMATIC iMap can identify whether the master on the other network is available.


SIMATIC 300(1) (Configuration) -- DPKoppler
(0) UR
PROFIBUS(1): DP master system
(10) DP/DP
DP 1
DP 2
Properties - DP slave
General
Parameter Assignment

1	
2	CPU 315-2 DP
X2	DP
3	
4	
5	

Parameters	Value
Station parameters	
Device-specific parameters	
Diagnose	Extended diagnosis enabled
Hex parameter assignment	
User_Prm_Data (0 to 3)	00,00,00,00

Figure 2-7 Configuration of the diagnostics for the DP/DP coupler

Addressing

The DP/DP coupler may have different addresses in the two PROFIBUS networks.

The addressing must be carried out both on the device and in SIMATIC iMap.

Note

Only use addresses 1 to 124. No other addresses are permitted on PROFIBUS-DP. Changed addresses do not take effect until you switch the power off and on again.

Creating the PROFInet component with STEP 7

The user creates the PROFInet component for the DP/DP coupler in STEP 7 and stores it in the file system in the same way as for a DP slave with fixed functionality.
The procedure is described in the section entitled "Creating PROFInet components with STEP 7".

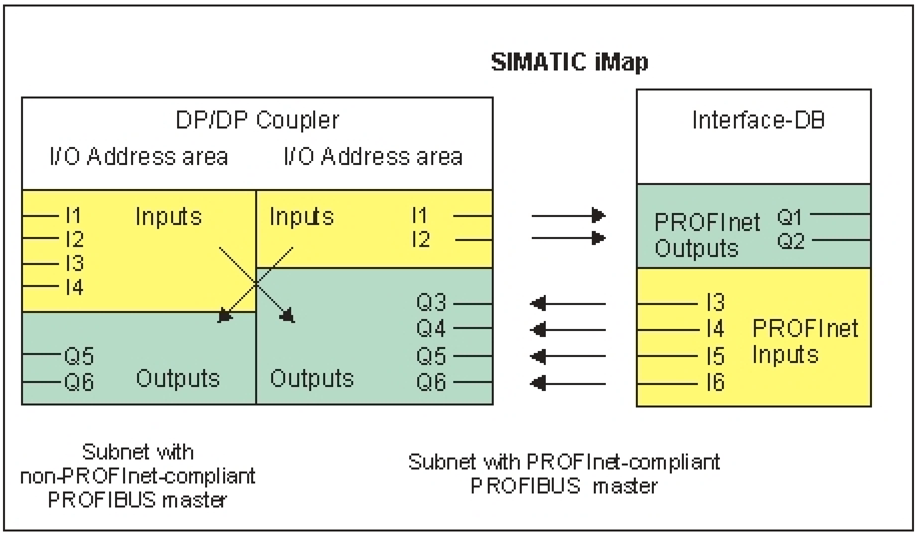


Figure 2-8 Example: Interface DB for a DP/DP coupler

Working with the PROFInet component in SIMATIC iMap

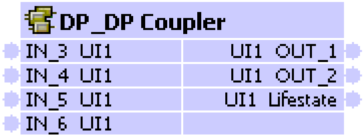
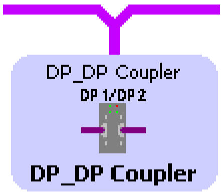
- Import the PROFInet component for the DP/DP coupler from the file system to a SIMATIC iMap library.
- In the net view, link the PROFInet component for the DP/DP coupler to the desired PROFIBUS.
- In the Properties dialog box for the DP/DP coupler in the net view, enter the PROFIBUS address set using DIL switches on the device.
  Note down as a comment which PROFIBUS networks are linked by the DP/DP coupler.
- Interconnect the inputs and outputs in the plant view.
- Generate the project, then download the programs to the associated DP master with proxy functionality.
  Download the interconnections to the DP/DP coupler.

Representation in SIMATIC iMap

The DP/DP coupler is a PROFIBUS device and is represented accordingly in the SIMATIC iMap plant and net views.

No connection can be seen between the two PROFIBUS networks linked in this way.

Table 2-9 PROFInet component of the DP/DP coupler

Technological function	PROFIBUS device
 DP_DP Coupler IN_3 UI1 IN_4 UI1 IN_5 UI1 IN_6 UI1 UI1 OUT_1 UI1 OUT_2 UI1 Lifestate	

Function in the PROFInet environment

The DP/DP coupler acts as a DP slave at the PROFIBUS interface.

PROFIBUS device: The PROFInet component of the DP/DP coupler has a PROFIBUS connector.

Technological function: The input and output signals of the DP/DP coupler may be represented as a technological function via the interface DB.

Diagnostics

The DP/DP coupler has two diagnostics options:

- The "Diagnostic" DIL switch on the top of the DP/DP coupler can be used to show the validity of data within the I/O data by setting the DIL switch to "1".

  The least significant bit of the first configured input byte (LSB) indicates whether data is being exchanged between the two PROFIBUS masters without faults.
  If the value of the bit it "1", all the DP/DP coupler data is valid.
  If the bit is set to "0", all the data is invalid.

- HW Config may be used to activate "extended diagnostics". To do this set the value to "Extended diagnostics enabled" during configuration in STEP 7.

  If the DP/DP coupler signals a diagnostic event, OB82 is called in the master and an entry is made in the diagnostic buffer.

  In SIMATIC iMap, diagnostics can be carried out via online monitoring or via the module state.

Further information

The structure, configuration, commissioning and diagnosis of the DP/DP coupler are described in detail in the DP/DP coupler manual.

The manual and further information can be obtained from:

http://www.ad.siemens.de/support

2.6 HMI devices in PROFInet components

When you create a PROFInet component, you can set up an HMI device (e.g. operator panel or multi-panel) as part of the PROFInet component.

This chapter describes the points to note when integrating HMI devices into a PROFInet component.

It only describes integrated HMI devices that were configured, generated and downloaded with ProTool/Pro CS.

Network configurations with integrated HMI devices

You can integrate an HMI device into a PROFInet component by incorporating it into the internal networking of the station. You have the following possibilities:

- either on the local PROFIBUS
- or on the local MPI bus
- or within a PC station

The local bus systems of the PROFInet component, and the devices that they network, cannot be seen in SIMATIC iMap. Consequently, an integrated HMI device is not displayed as such in SIMATIC iMap.

Rules for integrating HMI devices

- A PROFInet component may have one and only one HMI device that was configured with ProTool. Several HMI devices in a PROFInet component are not supported.
- The station of a PROFInet component may not contain two local bus systems (PROFIBUS and MPI). If a local PROFIBUS is configured, an HMI device may not be configured via MPI and vice versa.
- For a complete C7 6xy device, the HMI device may only be integrated via MPI. It cannot be configured on the PROFIBUS.

Example: Integrated HMI device on the local PROFIBUS of a PROFInet device

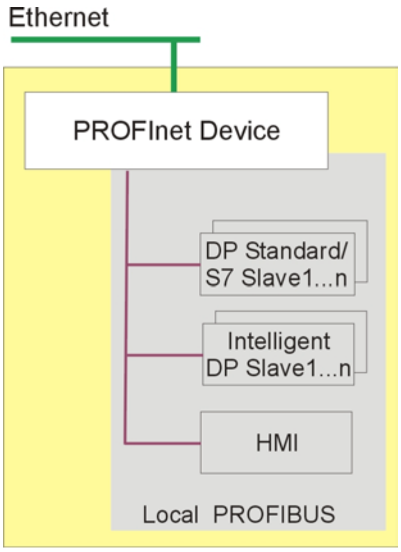


Figure 2-9 HMI device on the local PROFIBUS of a PROFInet device

Example: Integrated HMI device on the local MPI bus of a PROFInet device

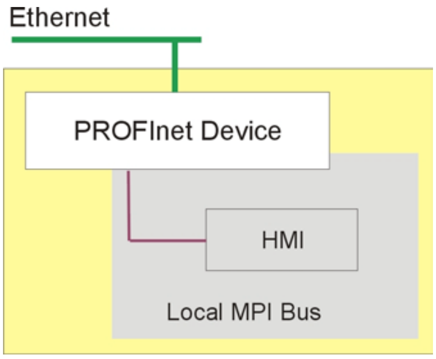


Figure 2-10 HMI device on the local MPI bus of a PROFInet device

Example: Integrated HMI device on the local PROFIBUS of a PROFIBUS device

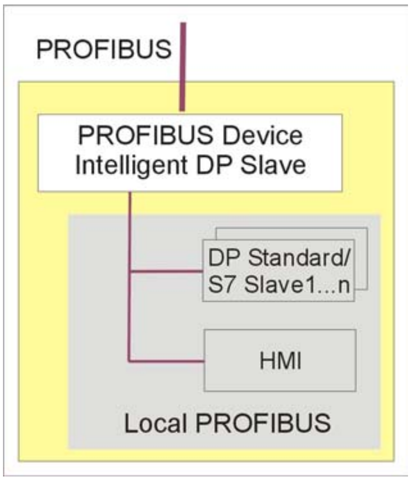

PROFIBUS
PROFIBUS Device Intelligent DP Slave
DP Standard/ S7 Slave1...n
HMI
Local PROFIBUS

Figure 2-11 HMI device on the local PROFIBUS of a PROFIBUS device

Example: Integrated HMI device on the local MPI bus of a PROFIBUS device

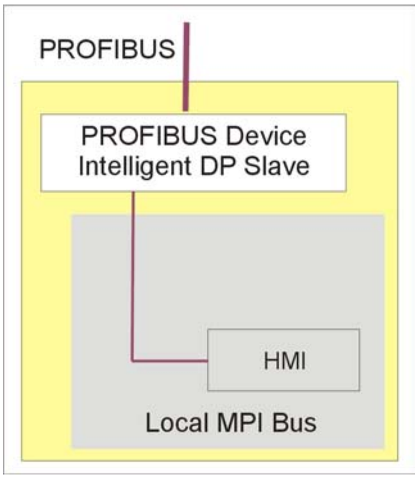


Figure 2-12 HMI device on the local MPI bus of a PROFIBUS device

Processing in SIMATIC iMap

An integrated HMI device as part of a PROFInet component cannot be seen in SIMATIC iMap. The device must be configured and the program downloaded to the device using ProTool/Pro CS.

Requirement

A direct connection between the PC and the HMI device is required in order to download to the HMI device. Set the access point of the S7ONLINE (STEP7) application to one of the following interface parameter settings in the PG/PC interface:

- MPI
- PROFIBUS
- Point-to-point connection.

Configuration and downloading

The following steps are required:

1. Generate the project in SIMATIC iMap using the **Generate >**menu command so that the generation status of the device is "Generated".
2. Select the device from the net view and select **Special > HMI** from the context menu. This calls the associated project in ProTool/Pro CS for the HMI device.
3. In ProTool/Pro CS: Change the configuration, if necessary, and download the data to the HMI device.

Caution

When the SIMATIC iMap project is regenerated, any changes to the HMI device configuration will be lost.

When you change the configuration, we therefore recommend that you recreate the entire PROFInet component and replace it in the SIMATIC iMap project.

Singleton components 3

3.1 Properties of singleton components

What are singleton components?

Singleton components are PROFInet components for which the device-specific data is stored and handled separately. The device-specific configuration information and program data is located in the STEP 7 basic project for the PROFInet component, rather than in the common shadow project.

Advantages in use

The use of singleton components has the following advantages:

- Special hardware configurations with SIMATIC device that were not supported in PROFInet components in the past can now be incorporated into PROFInet communication as singleton components.
- The device-specific configuration and program data is stored separately, so singleton components in SIMATIC iMap are excluded from certain time-consuming functions such as generation or program downloads, and thus do not affect the processing time of the overall project.
- Singleton components may have default names and addresses. Thus, these properties do not have to be configured for instances of singleton components in SIMATIC iMap.

Rule

Singleton components can only be created for PROFInet devices without proxy functionality. Thus, singleton components have only one bus connector on the Industrial Ethernet.

Special hardware configurations

The singleton concept allows previously unsupported hardware configurations with SIMATIC devices to be incorporated into PROFInet communication, such as:

- S7-400 automation systems
- Fail-safe systems – Distributed Safety – S7-300F
- Micromaster-type drives
- Integration of process diagnostics
- Functional modules (FM)
- Integration of PROFInet I/O

3.2 Creating singleton components

Requirements

The requirements for standard PROFInet components must be fulfilled, i.e. the STEP 7 project with hardware configuration, the PROFInet interface DB and the S7 program must have been created.

The Ethernet interface must also be configured with the correct IP address and subnet mask.

Note

For singleton components that can only have one instance in the SIMATIC iMap project it is not possible to change the names and address(es) of the instance.

To create singleton components:

1. Open the project in SIMATIC Manager, if you have not already done so.
2. Highlight the station and select the **Create PROFInet component** menu command.
3. On the "General" tab:
   - Select "Create component from the station".
   - Enter the name, the version number (optional) and a comment, if necessary.
4. On the "Component type" tab, check the "Singleton" check box.
5. On the "Storage area" tab, define the storage area of the component: target library and/or file system.
6. On the "Additional Properties" tab:
   - Define the paths for the icon files.
   - Define the path or address of the documentation link (optional).
   - In the "Identification" box, decide whether an existing identification should be retained or whether a new identification should be assigned.
7. Click on OK to confirm your input.

Result: The singleton component is created in the specified storage area.

Notice

Singleton components have no STEP 7 component project.

The STEP 7 basic project of a singleton component is only needed for the following actions:

- Downloading programs to the target system
- Re-editing singleton components
- Looking up device-specific diagnostics

We recommend that you store the STEP 7 basic project in the STEP 7 folder provided (step7\s7proj) so that it can be quickly traced in SIMATIC Manager.

Further procedure

- If you have specified a target library as the storage area, you can insert the singleton component immediately into a SIMATIC iMap project.
- If you have specified a path in the file system as the storage area, you will first have to import the singleton component into a library in SIMATIC iMap.

Appendix 4

4.1 Naming conventions for PROFInet components

The following rules must be followed when you assign names:

File names and paths

- A file name must not contain any of the following characters:
  \ / :„?“ * < > |
- A path name must not contain any of the following characters:
  / :„?“ * < > |

Names of PROFInet components

- The name may not be left blank.
- Illegal characters

  The name must not contain characters with the following ISO 8859-1 codes: 0 to 31, 33, 34, 38, 42 to 44, 46, 47, 58 to 63, 91 to 94, 96, 124, 127 to 159, 180.

  The characters that are not permitted are thus:

  - all non-printing characters (with ISO 8859-1 codes 0 to 31, 127 to 159)
  - AND !„ “ & * + ‘ . , / : < = > ? [\ ] ^ ` | ´
- Spaces are not permitted at the start and end of the name.
- The first character must not be an underscore.
- The name of a PROFInet component created in STEP 7 must not be more than 24 characters long.

Names of connectors

- The names of connectors may only contain the following characters (ISO 10646-1):

  Letters A to Z and a to z, digits 0 to 9, and the underscore character "_".
- The first characters of a connector name must be a letter (not underscore or a number).
- The name of a connector must not exceed 24 characters in length.

4.2 Data Types of Connectors

Data types, data lengths and ranges of values for connectors

A PROFInet data type conforming to Microsoft OLE 2.0 is assigned to every S7 data type . The following table shows the assignment, data lengths and ranges of values.

Table 4-1 Data types, data lengths and ranges of values for connectors

PROFInet data type	S7 data type	Data length in bytes	Range of values
BOOL	BOOL	2	TRUE / FALSE
I1	CHAR	1	-128 to +127
UI1	BYTE	1	0 to 255
R4	REAL	4	3.4E +/- 38
I2	INT	2	-32,768 to +32,767
UI2	WORD	2	0 to 65,535
I4	DINT	4	- 2.147.483.648 to +2.147.483.647
UI4	DWORD	4	0 to 4.294.967.295
DATE	DT (DATE_AND_TIME)	8	01.01.1900 00:00:00 to 31.12.9999 23:59:59*)
BSTR	STRING	4 + 2 * number of characters	Character string
ARRAY	ARRAY	Number * data type length	One-dimensional, from simple data types**)
STRUCT	STRUCT	Sum of the data type lengths	Made up from simple data types**)

*) There may be restrictions in the range of values, depending on the type of device. For example, for SIMATIC devices this would be: 01.01.1990 00:00:00 to 31.12.2089 23:59:59.

**) Simple data types are all the listed data types, apart from ARRAY and STRUCT. The maximum length depends on the amount of memory in the device.

Data types supported in PROFInet runtime version V2.0.0.0 or later

The following data types are supported by PROFInet devices from runtime version V2.0.0.0 onwards:

- Data type STRUCT with all simple data types, including STRING
- Data type ARRAY with the data types BOOL, REAL, DATE_AND_TIME, STRING

Earlier runtime versions do not support these data types.

4.3 Attributes for PROFInet interfaces

The attributes for PROFInet interfaces are set automatically in the PROFInet interface editor. In SIMATIC Manager, the attributes can be viewed in the object properties, "Attributes" tab. They start with **CBA**.

Caution

Attributes of PROFInet interfaces may not be modified, added or deleted manually in the object properties. This could lead to errors when creating the PROFInet component and to run-time errors when the component is used in the plant.

Always use the PROFInet interface editor to create or edit PROFInet interfaces.

Note

Remember that the attributes are case-sensitive.

Attributes for blocks

Table 4-2 PROFInet attributes for blocks

Attribute	Value	Meaning
CBA_db	true	active PROFInet interface DB for a PROFInet component with programmable functionality
	true_s	active PROFInet interface DB for a PROFInet component with fixed functionality (e.g. standard or S7 slave)
	FALSE	inactive PROFInet interface DB
CBA_intern_db	true	active HMI interface DB or FB
	FALSE	inactive HMI interface DB or FB
CBA_interface_version	a.b.c.d	current version number (4 numbers separated by dots)
CBA_islave_copy_fc	in	PROFInet function PN_IN (FC10), copy block for intelligent DP slaves
	out	PROFInet function PN_OUT (FC11), copy block for intelligent DP slaves
CBA_islave_copy_db	in	PROFInet data block PN_IO DB, copy block for intelligent DP slaves
CBA_iecp_copy_fb	true	PROFInet function block (FB88) for CP 3xx PN
CBA_islave_copy_db	true	PROFInet In Out data block

4.4 The "lifestate" connector in SIMATIC devices

The technological functions of the PROFInet components that were created in STEP 7 receive an additional output of the type UI1,called "Lifestate". This output allows the PROFInet components to be monitored by the user program of a PROFInet communication partner. The "Lifestate" output may be interconnected as required for diagnostic purposes.

Lifestate values

The lifestate output can return the following values which describe the current state of the device:

Value	Designation of the state	General significance
0x00	NonExistent	The device is not powered and is thus unable to communicate. This operating state cannot be transferred via the interconnection since no communication is possible. It is only listed for the sake of completeness. This value is also the default substitute value.
0x01	Initializing	The device is initializing, e.g. after a power off or reset.
0x02	Ready	The device is ready for use, but the component is not active. The device outputs are in the safe state. This state corresponds to the STOP operating state of a CPU, for example.
0x03	Operating	The device is operating normally. This state corresponds to the RUN operating state of a CPU, for example.
0x04	Defect	There is an error in the device that cannot be rectified without further intervention. Typical examples include hardware or firmware errors. The device can only be restarted by switching the power off and on again. Only limited communication is possible with the device, depending on the type of error.

The value is 1 byte long.

Abbreviations 5

Abbreviations

Abbreviation	Explanation
CbA	**C**omponent **b**ased **A**utomation
COM	**C**omponent **O**bject **M**odel
DCOM	**D**istributed **C**omponent **O**bject **M**odel
GUID	**G**lobal **U**nique **ID**entifier
HMI	**H**uman **M**achine **I**nterface
MES	**M**anufacturing **E**ngineering **S**ystem
MAC	**M**edium **A**ccess **C**ontrol
PN	**P**ROFI**N**et
PNO	**P**ROFIBUS **N**utzer**O**rganisation e.V. (PROFIBUS User Organization)
SRT	**S**oft **R**eal-**T**ime
TIA	**T**otally **I**ntegrated **A**utomation
UNC	**U**niform **N**aming **C**onvention
URL	**U**niform **R**esource **L**ocator
XML	E**X**tended **M**arkup **L**anguage

Glossary

6

Term	Description
Basic project	Project folder in the vendor-specific configuration tool from which a PROFInet component is created. Example: STEP 7 basic project
Chart	In SIMATIC iMap: Folder that contains the technological function and their interconnections. The chart is used to structure the plant. We distinguish between the -> Main chart and -> subordinate charts
Chart, subordinate	In SIMATIC iMap: A subordinate chart may be located on any lower hierarchically level in the plant view. It may contain technological functions and further subordinate charts.
Component based Automation	Concept for implementing modular, distributed automation applications based on open standards for data processing and data communications. Component based Automation is an extension of Totally Integrated Automation (TIA).
Component project	A folder containing the vendor-specific project data for the PROFInet component, e.g. in the form of a STEP 7 project.
Continuation connector	Graphical representation of the end point of an interconnection. Interconnections are represented by lines or continuation connectors in SIMATIC iMap.
Device	In Component based Automation, this is part of the PROFInet component that contains the hardware-specific data for the PROFInet component. In SIMATIC iMap, a device is the software representation of the physical device for which the PROFInet component was created. In the SIMATIC iMap net view it is represented as an object with one or more bus ports. We distinguish between -> PROFInet devices and -> PROFIBUS devices according to the communication functions to be performed.
Functionality, fixed	A PROFInet component with fixed functionality does not contain its own user program.
Functionality, programmable	A PROFInet component with programmable functionality contains its own user program.
Global library	A library that can be opened, closed archived and retrieved in SIMATIC iMap using menu commands. The global library contains PROFInet components that can be used in several projects.
Instance	The instance of a PROFInet component is one use of that component type in the SIMATIC iMap project.
Interconnection	General: A logical data link between two objects. In SIMATIC iMap: Interconnection between two technological functions. An output is always connected to an input of the same data type. Interconnections are represented by lines or continuation connectors in SIMATIC iMap.

Term	Description
Interconnections to local communication partners	Interconnections to local communication partners are: • Interconnections between two PROFIBUS devices on the same PROFIBUS • Interconnections between a PROFIBUS device and the associated PROFInet device with proxy functionality.
Interconnections to remote communication partners	Interconnections to remote communication partners are interconnections that run via Industrial Ethernet.
Interface DB	In Component based Automation, this is a data block in STEP 7 that describes the inputs and outputs of the technological function.
Library	In Component based Automation, this is a folder containing reusable PROFInet components. We distinguish between two types: > Project library and > Global library.
Main chart	In SIMATIC iMap: The main chart is the highest level of the hierarchical representation of the plant view. It contains all the technological functions and subordinate charts and their interconnections.
Net view	Representation of the devices and the networks (Ethernet, PROFIBUS) in SIMATIC iMap.
Plant view	Representation of the technological functions of the automation system and its interconnections in SIMATIC iMap. The plant view displays one chart.
PROFIBUS device	In Component based Automation, a PROFIBUS device has just one PROFIBUS connector as a slave. It does not participate directly in PROFInet communication and is integrated via a PROFInet device with proxy functionality.
PROFInet	Standard published by the Profibus User Organization (PNO) to define a cross-vendor communication and engineering model.
PROFInet device	A device on the Ethernet is a PROFInet device. A PROFInet device may also have a PROFIBUS connector as a master (PROFInet device with proxy functionality) for PROFIBUS devices.
PROFInet device with proxy functionality	The PROFInet device with proxy functionality is a proxy of the PROFIBUS devices on the Industrial Ethernet. The proxy functionality allows a PROFIBUS device to communicate with all PROFInet communication partners, as well as with its own DP master. If the device has a local (internal) PROFIBUS, then it is the DP master for the local DP slaves.
PROFInet component	Software representation of a technological module with defined functionality. An automation system is made up of several PROFInet components. A PROFInet component essentially consists of a technological function and the associated device.
Project tree	In Component based Automation: Hierarchic representation of the SIMATIC iMap project. There are two parts to this representation: The plant tree contains the technological functions. This part may be structured hierarchically using subordinate charts. The plant part always contains the main chart and may contain any number of nested subordinate charts. The net tree contains the devices. The slaves coupled to a PROFIBUS master are represented hierarchically.

Term	Description
Project library	Library which is contained in a SIMATIC iMap project and is opened, closed, archived and retrieved automatically with the project. The project library contains all the PROFInet components whose instances are contained in the project.
Project view	Representation of the assignment between PROFInet components and their instances in the SIMATIC iMap project.
Proxy functionality	see PROFInet device with proxy functionality
Proxy system	A proxy system consists of a PROFInet device with proxy functionality and all the PROFIBUS devices coupled to it.
Scanning frequency	For acyclical transfer, the scanning frequency is the frequency with which the data is polled, e.g. every 200 ms. The scanning frequency is thus the maximum time that elapses before a change in value is transferred.
Shadow project	Folder in which the vendor-specific data of the project is stored during generation in SIMATIC iMap.
SIMATIC iMap	The Siemens engineering tool for Component based Automation. It is used to configure, commission and monitor modular, distributed automation systems based on the PROFInet standard.
SIMATIC iMap - STEP 7 Add-on	Software for the SIMATIC iMap link to STEP 7.
Singleton component	In SIMATIC iMap, PROFInet components whose device-specific configuration and programming data is held and edited in the (STEP 7) basic project, rather than in the common shadow project. This component type allows you to incorporate hardware configurations with SIMATIC devices that were previously not supported into PROFInet communication.
Technological function	General: This is a task or sub-task within a technological process, e.g. measurement, control of motors, positioning. A technological function can be implemented both as hardware - e.g. a functional module - and as software – e.g. a control block. In Component based Automation, the technological function of a PROFInet component comprises the application-specific functionality of a programmable controller or field device and the component interface for communication with other PROFInet components. It is represented as a block with inputs and outputs in the SIMATIC iMap plant view.
Transfer frequency	For cyclical transfer, the transfer frequency is the time after which a data item is transferred again, e.g. every 100 ms.

Index