SIEMENS

SIMATIC

Configuring Hardware and Communication Connections STEP 7

Manual

This manual is part of the documentation package with the order number:
6ES7810-4CA10-8BW0

05/2010
A5E02789741-01

Legal information

Warning notice system

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

⚠DANGER
indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠WARNING
indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠CAUTION
with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.

CAUTION
without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.

NOTICE
indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

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

Qualified Personnel

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

Proper use of Siemens products

Note the following:

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

Trademarks

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

Disclaimer of Liability

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

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E02789741-01
Ⓟ 02/2010

Preface

Purpose

This manual provides a complete overview of programming with **STEP 7**. It is designed to support you when installing and commissioning the software. It explains how to proceed when creating programs and describes the components of user programs.

The manual is intended for people who are involved in carrying out control tasks using STEP 7 and SIMATIC S7 automation systems.

We recommend that you familiarize yourself with the examples in the manual "Working with STEP 7 V5.5, Getting Started." These examples provide an easy introduction to the topic "Programming with STEP 7".

Basic Knowledge Required

In order to understand this manual, general knowledge of automation technology is required.

In addition, you must be familiar with using computers or PC-similar tools (for example, programming devices) with the MS Windows XP, MS Windows Server 2003 or MS Windows 7 operating system.

Scope of the Manual

This manual is valid for release 5.5 of the STEP 7 programming software package.

You can find the latest information on the service packs:

- in the "readme.rtf" file
- in the updated STEP 7 online help.

The topic "What's new?" in the online help offers an excellent introduction and overview of the newest STEP 7 innovations.

STEP 7 Documentation Packages

This manual is part of the documentation package "STEP 7 Basic Information“.

The following table displays an overview of the STEP 7 documentation:

Documentation	Purpose	Order Number
STEP 7 Basic Information with • Working with STEP 7, Getting Started Manual • Programming with STEP 7 • Configuring Hardware and Communication Connections, STEP 7 • From S5 to S7, Converter Manual	Basic information for technical personnel describing the methods of implementing control tasks with STEP 7 and the S7-300/400 programmable controllers.	6ES7810-4CA10-8BW0
STEP 7 Reference with • Ladder Logic (LAD) / Function Block Diagram (FDB) / Statement List (STL) for S7-300/400 manuals • Standard and System Function for S7-300/400 Volume 1 and Volume 2	Provides reference information and describes the programming languages LAD, FBD and STL, and standard and system function extending the scope of the STEP 7 basic information.	6ES7810-4CA10-8BW1

Online Helps	Purpose	Order Number
Help on STEP 7	Basic information on programming and configuring hardware with STEP 7 in the form of an online help.	Part of the STEP 7 Standard software.
Reference helps on AWL/KOP/FUP Reference help on SFBs/SFCs Reference help on Organization Blocks	Context-sensitive reference information.	Part of the STEP 7 Standard software.

Online Help

The manual is complemented by an online help which is integrated in the software.

This online help is intended to provide you with detailed support when using the software.

The help system is integrated in the software via a number of interfaces:

- There are several menu commands which you can select in the **Help** menu:
  The **Contents** command opens the index for the Help on STEP 7.
- **Using Help** provides detailed instructions on using the online help.
- The context-sensitive help offers information on the current context, for example, an open dialog box or an active window. You can open the contextsensitive help by clicking the "Help" button or by pressing F1.
- The status bar offers another form of context-sensitive help. It displays a short explanation for each menu command when the mouse pointer is positioned on the menu command.
- A brief explanation is also displayed for each icon in the toolbar when the mouse pointer is positioned on the icon for a short time.

If you prefer to read the information from the online help in printed format, you can print out individual help topics, books, or the entire online help.

This manual, as well as the manuals "Programming with STEP 7", "Modifiying the System During Operation via CiR" and "Automation System S7-400H - Fault-Tolerant Systems" is an extract from the HTML-based Help on STEP 7. For detailed procedures please refer to the STEP 7 help. As the manuals and the online help share an almost identical structure, it is easy to switch between the manuals and the online help.

You can find the electronic manuals after installing STEP 7 via the Windows Start menu: **Start > SIMATIC > Documentation.**

Further Support

If you have any technical questions, please get in touch with your Siemens representative or responsible agent.

You will find your contact person at:

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

You will find a guide to the technical documentation offered for the individual SIMATIC Products and Systems at:

http://www.siemens.com/simatic-tech-doku-portal

The online catalog and order system is found under:

http://mall.automation.siemens.com/

Training Centers

Siemens offers a number of training courses to familiarize you with the SIMATIC S7 automation system. Please contact your regional training center or our central training center in D 90026 Nuremberg, Germany for details:

Internet: http://www.sitrain.com

Technical Support

You can reach the Technical Support for all Industry Automation and Drive Technology products

- Via the Web formula for the Support Request
  http://www.siemens.com/automation/support-request

Additional information about our Technical Support can be found on the Internet pages
http://www.siemens.com/automation/service

Service & Support on the Internet

In addition to our documentation, we offer our Know-how online on the internet at:

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

where you will find the following:

- The newsletter, which constantly provides you with up-to-date information on your products.
- The right documents via our Search function in Service & Support.
- A forum, where users and experts from all over the world exchange their experiences.
- Your local representative for Industry Automation and Drive Technology.
- Information on field service, repairs, spare parts and consulting.

Contents

1 Basics of Configuring Hardware with STEP 7

1.1 Introduction to Configuring Hardware

You configure the hardware in HW Config. You can open HW Config by selecting a station in the SIMATIC Manager and then double-clicking on the "**Hardware**" icon.

Configuring

The term "configuring" refers to the arranging of racks, modules, distributed I/O (DP) racks, and interface submodules in a station window. Racks are represented by a configuration table that permits a specific number of modules to be inserted, just like a real rack.

In the configuration table, STEP 7 automatically assigns an address to each module. You can change the addresses of the modules in a station if the CPU in the station can be addressed freely (meaning an address can be assigned freely to every channel of the module, independent of its slot).

You can copy your configuration as often as you like to other STEP 7 projects, modify it as necessary, and download it to one or more existing plants. When the programmable controller starts up, the CPU compares the preset configuration created in STEP 7 with the actual configuration of the plant. Any errors are therefore recognized immediately and reported.

Assigning Parameters

The term "assigning parameters" refers to the following:

- Setting properties for programmable modules in a local configuration and in a network. For example: a CPU is a programmable module. Its watchdog time is a parameter you can set.
- Setting bus parameters, master and slave parameters for a master system (PROFIBUS) or making other settings for exchanging data between components.

The parameters are downloaded to the CPU and transferred during startup by the CPU to the respective modules. Modules can easily be replaced because the parameters set with STEP 7 are automatically downloaded to the new module during startup.

When Should You "Configure Hardware"?

The properties of the S7 programmable controllers and modules are preset with default values such that in many cases you do not need to configure them.

Configuration is necessary in the following cases:

- If you want to change the default parameters of a module (for example, enable a hardware interrupt for a module)
- If you want to configure communication connections
- For stations with a distributed I/O (PROFIBUS DP or PROFINET IO)
- For S7-400 stations with a number of CPUs (multicomputing) or expansion racks
- For fault-tolerant (H) programmable control systems

1.2 Basic Procedure for Configuring Hardware

Window for Configuring

Configuring a programmable controller involves the use of two windows:

- The station window in which you place the racks for the station structure
- The "Hardware Catalog" window from which you select the required hardware components, for example, racks, modules, and interface submodules

Displaying the Hardware Catalog

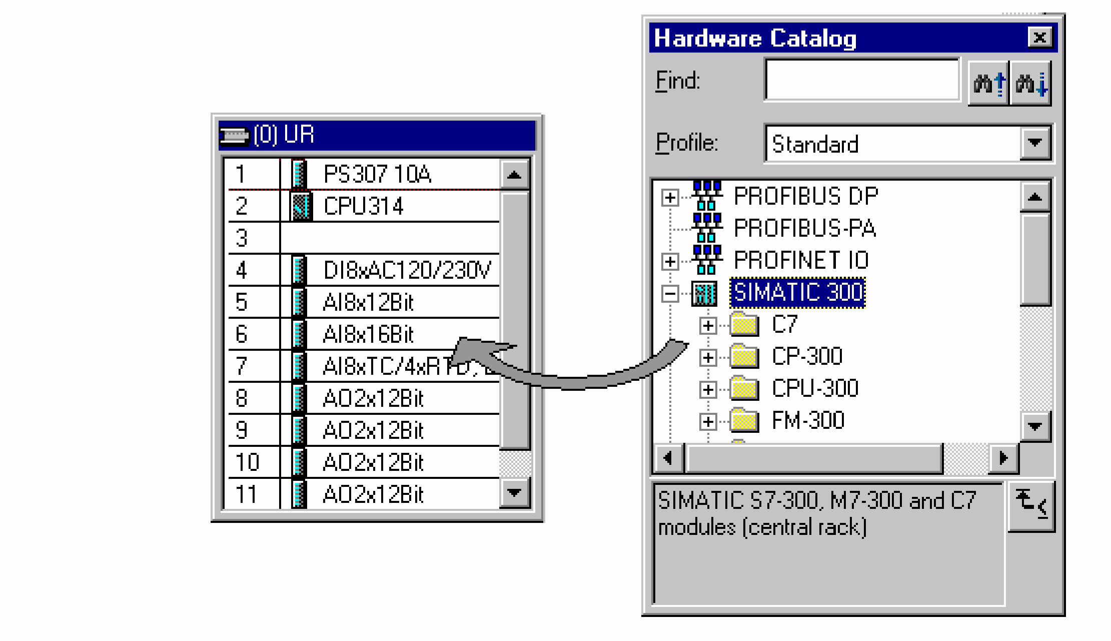
If the "Hardware Catalog" window is not displayed, select the menu command **View > Catalog**. This command toggles the display of the Hardware Catalog on and off.

1.2.1 Basic Steps for Configuring a Station

Independent of which structure a station has - you always configure using the following steps:

1. Select a hardware component in the "Hardware Catalog" window.
2. Copy the selected component to the station window using drag & drop.

The following figure shows the basic operation:



(0) UR

1	PS307 10A
2	CPU314
3	
4	DI8xAC120/230V
5	AI8x12Bit
6	AI8x16Bit
7	AI8xTC/4xRTD,
8	AO2x12Bit
9	AO2x12Bit
10	AO2x12Bit
11	AO2x12Bit

Hardware Catalog

Find:

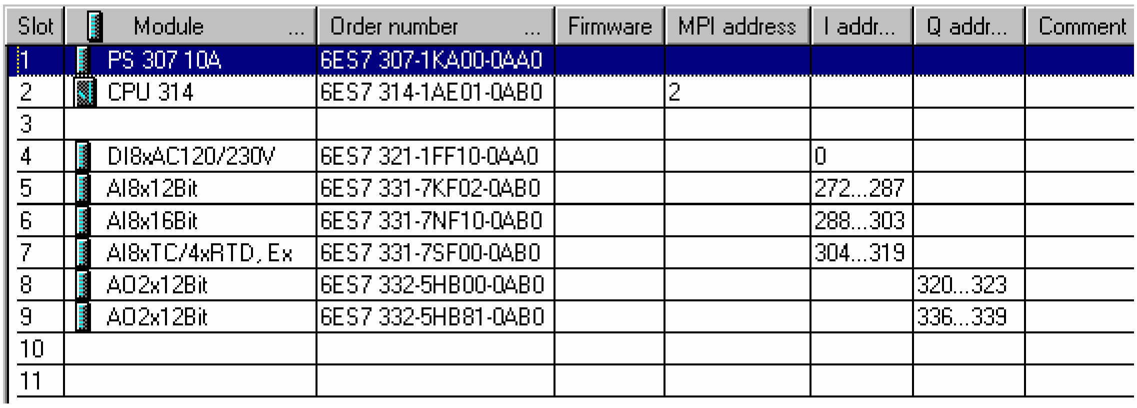
Profile: Standard

SIMATIC S7-300, M7-300 and C7 modules (central rack)

1.2.2 Layout of the Station Window

The lower part of the station window shows a detailed view of the inserted/selected rack. The order numbers and addresses of the modules are shown here in table form.

The table has the structure shown below for a central rack equipped with modules (detailed view):

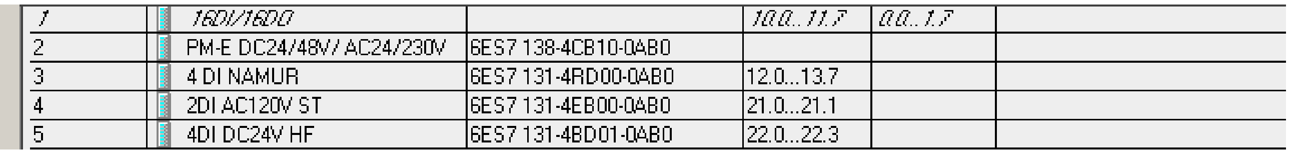


Slot	Module ...	Order number ...	Firmware	MPI address	I addr...	Q addr...	Comment
1	PS 307 10A	6ES7 307-1KA00-0AA0					
2	CPU 314	6ES7 314-1AE01-0AB0		2			
3							
4	DI8xAC120/230V	6ES7 321-1FF10-0AA0			0		
5	AI8x12Bit	6ES7 331-7KF02-0AB0			272...287		
6	AI8x16Bit	6ES7 331-7NF10-0AB0			288...303		
7	AI8xTC/4xRTD, Ex	6ES7 331-7SF00-0AB0			304...319		
8	AO2x12Bit	6ES7 332-5HB00-0AB0				320...323	
9	AO2x12Bit	6ES7 332-5HB81-0AB0				336...339	
10							
11							

Representation of Modules in the Detailed View

The following special features apply to the display of modules and submodules:

- If DP slaves were configured in "DPV0" interrupt mode, the modules of these DP slaves have a faded appearance in the detailed view. The faded appearance indicates the modules do not provide any diagnostic information.
- If the name of the module is shown in italics, this is integrated in the central module and cannot be configured individually.



1	*16DI/16DO*		*10.0...11.7*	*0.0...1.7*	
2	PM-E DC24/48V/ AC24/230V	6ES7 138-4CB10-0AB0			
3	4 DI NAMUR	6ES7 131-4RD00-0AB0	12.0...13.7		
4	2DI AC120V ST	6ES7 131-4EB00-0AB0	21.0...21.1		
5	4DI DC24V HF	6ES7 131-4BD01-0AB0	22.0...22.3		

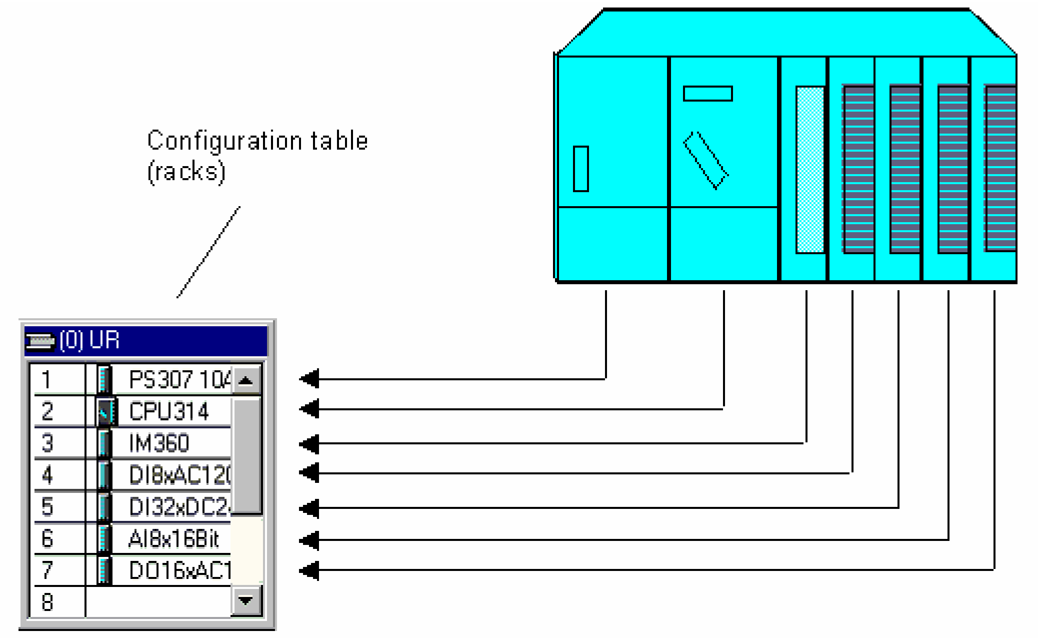
1.2.3 Configuration Table as a Representation of a Rack

For a local configuration you arrange the modules beside the CPU in a rack and continue into additional expansion racks. The number of racks which can be configured depends on the CPU you used.

Just as you do in a real plant, you arrange your modules in racks with STEP 7. The difference is that in STEP 7 racks are represented by "configuration tables" that have as many rows as the rack has slots for modules.

The following figure shows an example of how a real structure is converted into a configuration table. The configuration table corresponds to the rack used; STEP 7 automatically places the number of the rack in brackets in front of the name.

Example: (0) UR corresponds to the central rack (Universal Rack) number 0.



(0) UR

1	PS307 10(
2	CPU314
3	IM360
4	DI8xAC12(
5	DI32xDC2
6	AI8x16Bit
7	DO16xAC1
8	

1.2.4 Setting the Properties of Components

Once you have arranged your components in the station window, you always arrive in the following manner in a dialog box for changing the default properties (parameters or addresses):

- Double-click the component or select the menu command **Edit > Object Properties.**
- Right mouse button: Move the cursor on the component, press the right mouse button and select the command **Object Properties** from the pop-up menu.

Properties of CPUs

The properties of the CPUs have a special significance for the behavior of the system. In the dialog boxes for a CPU, you can set the following, for example: startup characteristics, local data areas and priorities for interrupts, memory areas, retentive behavior, clock memory, protection level, and password - to name only a few. STEP 7 "knows" what can be set and within what range limits.

In the "General" tab of the CPU or via the properties of the CPU interface you can assign parameters to the interfaces (for example, MPI or integrated PROFIBUS-DP interfaces). Via these dialog boxes you can also access the properties dialog box for the corresponding subnet to which the CPU is to be connected.

Other Ways of Assigning Parameters

For S7-300 and S7-400 programmable controllers you can set the parameters for some modules in the user program (for example, for analog modules). You need to call the system functions (SFCs) WR_PARM, WR_DPARM, and PARM_MOD in the user program to do this. These settings are lost following a warm restart.

You will find more detailed information about system functions in the *System Software for S7-300 and S7-400, System and Standard Functions Reference Manual*

For M7-300 and M7-400 programmable control systems you can set the parameters for signal modules in the C program. You need to call the M7 API function "M7StoreRecord" in the C program to do this. This function transfers the parameters to a signal module.

You will find more information about the M7 API functions in the manuals on the system software for M7-300 and M7-400.

1.2.5 Opening Objects in HW Config

In STEP 7, you can open objects in order to edit them.

In the SIMATIC Manager, for example, select the "Hardware" object and then select the menu command **Edit > Open Object** to start HW Config. You can edit the station configuration with HW Config.

Double-clicking on the object has the same effect as "Edit Objects".

Special Feature of HW Config

If want to edit an object in HW Config, the double-clicking it produces the same result as selecting the menu command **Edit > Object Properties**. In general, this is the place to assign addresses, set parameters or enter other information. If there is an application available to edit the object, then a button for starting this application is located in the property sheet for the object.

As of STEP 7 V5.4, the menu command **Edit > Open Object With** is available. You can use this menu command to open an application to edit the object as required. This menu command is only available if such an application is necessary and available for use.

1.2.6 What You Should Know About Slot Rules and Other Rules

STEP 7 offers you support with configuring a station so that a message is generally displayed immediately if, for example, a module cannot be inserted in the slot you want to insert it in.

As of STEP 7 V5.2 you can identify available module slots at a glance, provided you have your monitor display set for more than 256 colors. When you select a module in the Hardware Catalog and this module can be inserted in a configured module rack, all slots available for the insertion of this module are highlighted in color. This feature increases configuring efficiency.

Furthermore, because address ranges are checked automatically, double address assignments cannot occur.

In this connection, please pay attention to the status bar at the lower edge of the window and to any displayed messages that provide details on the causes and effects of an operation. You can also use the online help to obtain additional information for the messages.

Other additional, temporary rules (for a specific release), such as restrictions to the slots you can use owing to a functional restriction to individual modules, are not taken into account. Consequently, always consult the documentation or the current Product Information for the modules.

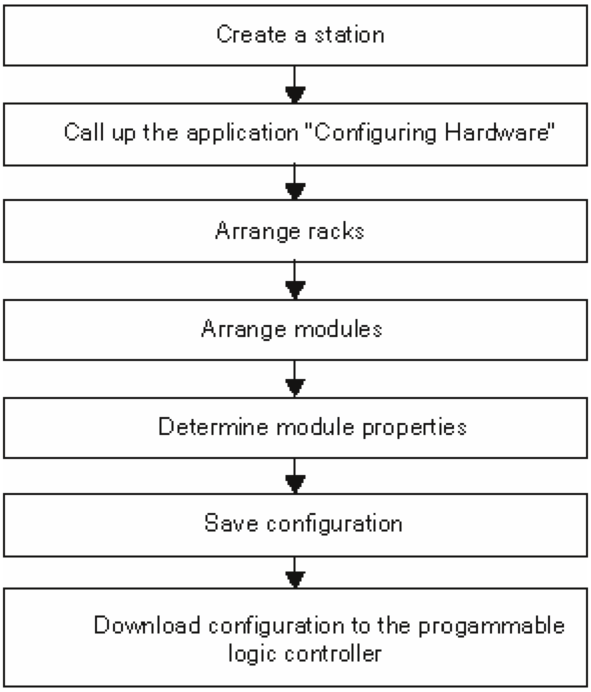
1.3 Overview: Procedure for Configuring and Assigning Parameters to a Local Configuration

Requirement

You must have opened a project or created a new project in the SIMATIC Manager.

Basic Procedure

To configure and assign parameters to a structure, proceed as shown below:


Create a station
Call up the application "Configuring Hardware"
Arrange racks
Arrange modules
Determine module properties
Save configuration
Download configuration to the progammable logic controller

Summary

As usual in Windows applications, you can put together the whole configuration in STEP 7 using drag and drop. You will find detailed information on handling and setting up your real plant configuration, for example, how you configure the connection to expansion racks or how you configure special components in the STEP 7 online help.

1.4 Working with the Hardware Catalog

1.4.1 Customizing the Hardware Catalog

In addition to the standard Hardware Catalog supplied with STEP 7, you can tailor your "own" catalog. This can result in a number of different catalog profiles. The basis for each new catalog profile is the standard Hardware Catalog with all modules/components - the profile for this catalog is called "Standard"; you can assign your own names for the catalog profiles you create yourself.

Example: You can create a catalog profile that contains only the modules you are currently using.

From STEP 7 V5.0, Service Pack 3 onwards, the following catalog profiles are available in addition to the "Standard" profile:

- Current modules, (for example, November 1999 or July 2000 version)
  In this profile, no "old" modules are available (only current order numbers)
- SIMATIC Outdoor (for example, November 1999 or July 2000 version)
  This profile offers only current modules which can be used in extended environmental conditions (higher mechanical stress and extended climatic conditions are permitted)

Note

The additional profiles available for selection are for modules that are available in the hardware catalog of the STEP 7 standard package. Modules that are installed by means of optional packages, GSD files or hardware updates are only available in the 'Standard' catalog profile!

Procedure

1. Select the menu command **Options > Edit Catalog Profiles**.
   In the application that opens, two catalog profiles are displayed: the "Standard" profile and an "empty" profile that does not contain any components as yet.
2. Drag the required folders and modules from the standard profile window and drop them in the empty profile window. You can also adapt the structure to your own requirements using the menu command **Insert > Folder**.
3. Save the new catalog profile using the menu command **File > Save As**; choose a suitable name for the new profile.
   The name of the new profile then appears in the "Profile" list box in the Hardware Catalog and can be selected.

GSD Files only update "Standard" Profile

DP slaves that have been installed later via GSD files are only added to the "Standard" profile's Other Field Devices folder and are not included in the catalog profiles you created yourself.

1.4.2 Searching in the Hardware Catalog

If you know the order number for a component or the name of a function that is displayed in the info text in the Hardware Catalog, then you can search for this component.

The upper section of the Hardware Catalog contains a "Search" box.

Procedure

1. Select and entry in the Hardware Catalog
2. Enter a search term, such as the first part of the order number **6ES7 331**.
3. To start searching, click on "Search down" or "Search up", as appropriate.

↓	Search down
↑	Search up

STEP 7 locates the first occurrence of the search term. If the component you are looking for is in another folder that is not open or outside of the visible area, then the folder is automatically opened and moved in to the visible area.

STEP 7 saves the search term that you entered. You can then easily select it again from the drop-down list in the search tool.

What Can Be Entered in the Search Box

You can enter any test, infotext, catalog entry or order number in the search box.

The search is not case sensitive.

Wildcards (*, ?,...) are supported.

You can also enter parts of a larger/complete search term (e.g. **331**).

Spaces, tabs or hyphens must be entered as they are shown in the Hardware Catalog; otherwise, the item being searched for will not be found (i.e. be sure to note the difference between the spaces and the hyphens in an order number!).

1.5 Tips for Editing Station Configurations

Docking the "Hardware Catalog" Window on the Side of the Station Window

To prevent the "Hardware Catalog" window obscuring the contents of the station window, you can "dock" it on one side edge of the station window; meaning it is then positioned and fixed there. Simply double-click the area above the "Profile" list box in the Hardware Catalog. To release the "docked" window, double-click in this area once again.

You can change the size (height and width) of the "Hardware Catalog" window when the window is undocked. When the window is docked, the width of the "Hardware Catalog" window can be changed.

Recognizing Available Slots

As of STEP 7 V5.2 you can easy recognize available slots for modules at first glance, provided your monitor display is set to more than 256 colors. If you select a module in the hardware catalog and the module can be inserted in a configured rack, then the possible slots for this module are highlighted in color. This feature makes configuration faster and easier for you.

If you click a module in the "Hardware Catalog" window, and no slot is selected, the module will then be placed in the first available slot.

Selecting a Number of Rows in the Configuration Table

If you want to select a number of rows in the configuration table to copy (for example, using the pop-up menu command Copy) or delete (for example, using the pop-up menu command Delete) a number of modules or insert a number of modules of the same type, proceed as follows:

To select all rows:	Select the menu command **Edit > Select > All**.
To select a group of consecutive rows:	Click on the first row of the group you want to select. Keep the SHIFT key pressed and click on the last row of the group you want to select.
To select a number of rows:	Press CTRL, keep it pressed, and click on each row you want to select.

You can also insert copied modules beyond the limitations of stations if you do not violate any slot rules when inserting.

Handling Complex Stations

If you have a complex station structure with a lot of racks, for example, you can set the configuration tables to their minimum size.

1. Select the configuration table.
2. Press the right mouse button and select the menu command **Minimize** in the popup menu.

You can also set this overview using the menu command **Options > Customize**.

Arranging the Racks in the View Automatically

Using the menu command **View > Arrange Automatically** you can let STEP 7 arrange your current view automatically.

Displaying Insertable ("Pluggable") Components in the Hardware Catalog

If you have selected a slot in a rack, then you can use the menu commands **Insert > Insert Object** or **Insert > Change Object** to display a selection of modules than can be inserted. This method saves you from having to search through the hardware catalog. You can select from all modules that are available in the current catalog profile.

1.5.1 Exchanging and Moving Modules

You can adapt an existing configuration not only by copying or deleting components, but also by moving and exchanging.

Moving Modules

You can move modules or other components in a simple manner by dragging and dropping them into another valid slot within the station.

Exchanging Modules

If you have already created a configuration and wish to exchange a module for which parameters have been specified previously (for example, CPU or analog module) with another module without "losing" the assigned parameters or connection configuration, then proceed as follows:

1. Drag the new module (for example, CPU) to the slot containing the module you want to replace.
2. Confirm you want to replace the module in the dialog box that appears.

If the message "The slot is already occupied" appears, you must activate the function first using the menu command **Options > Customize** and selecting the option "Enable module exchange."

Alternative:
After you have selected a slot of a module rack, you can open the context-sensitive menu (right-click) **Insert Object** or **Replace Object** to view a list of modules you can insert. This feature saves you the search through the Hardware Catalog. You select from all the modules that are listed in the currently available catalog profile.

You can only exchange "compatible" modules. If the modules are not compatible, you have to delete the old module, insert the new module, and assign parameters to it again. STEP 7 uses an appropriate message to acknowledge an attempt to exchange modules that are not compatible.

Example: You cannot exchange a CPU with parameters assigned for a CPU with a new order number - the entire parameter assignment (for example, the MPI address) will be adopted by the new module.

1.5.2 Exchanging Racks, C7 Devices, and DP Slaves

As of STEP 7 V5.1, Service Pack 1, if you have configured a station and assigned addresses and parameters to the modules in a SIMATIC 400 rack, a C7 control system, or a DP slave, you can exchange the rack if you maintain the module arrangement and settings. In this case, a C7 control system and a DP slave are considered racks.

Exchanging racks is not relevant for the SIMATIC 300 systems with only one rack type (standard rail).

When Does It Make Sense to Exchange A Rack?

It is always necessary to exchange the rack of a SIMATIC 400 station if functionality can only be expanded with the help of a different rack.

- Replacement of a power supply that does not have redundant capability with a power supply that does have redundant capability, but is not supported by the old rack.
- Replacement of a "short" rack (9 slots) with a "long" rack (18 slots) in order to insert additional modules.
  For racks that are configured as expansion racks (UR or ER with Receive-IM), the Receive-IM is automatically placed in the last slot.
- A station that was configured originally with a long rack must be configured on a short rack for reasons of space.
  However, this exchange will not be permitted for long racks that are configured as expansion racks (UR or ER with Receive-IM).

Exchange of a C7 control system is always necessary in the following cases:

- The old C7 control system is to be replaced by a new one (order number with higher function status, newer firmware version)
- The functionality of the old C7 control system is to be expanded (new C7 control system with more integrated modules).

Procedure

If you want to exchange a rack that has been fitted with modules, a C7 control system, or a DP slave in a station configuration, proceed as follows:

1. In the station configuration, select the components (for example, a rack) that you want to exchange.
2. In the Hardware Catalog window, select a component that is compatible with the selected component (see below), but that has a different order number. You cannot exchange identical components.
   For DP slaves, the slave interface module (for example, the IM 153-2) is selected from the Hardware Catalog window for the exchange.
3. Double-click on the component you want in the Hardware Catalog.
   If the component is compatible, the exchange is made and the modules of the originally configured component are adopted with their addresses and parameters (to the extent possible).

An exchange is also possible if you use a drag-and-drop operation to move the component from the Hardware Catalog window to the component to be exchanged.

When Is a Rack "Compatible"?

A rack of a SIMATIC 400 station can be exchanged for another rack only if the basic rules listed below are adhered to. If one or more of the basic rules are not adhered to, STEP 7 does not permit an exchange and terminates the procedure with an error message that provides information on the cause of the termination.

- A segmented rack (CR2) cannot be exchanged for a non-segmented rack (for example, UR1) and vice versa.
- You must be able to insert all modules of the rack to be exchanged into the new rack; that is, no slot rules must be violated when the modules are inserted into the new rack.

Example 1: An attempt to replace a UR1 rack that has a CPU with an ER1 expansion rack is not permitted. Inserting a CPU in the ER1 violates a slot rule - no CPUs can be inserted in an ER1.

Example 2: An attempt to replace a segmented central rack in an S7-400 station (CR2) with a Universal rack (for example, UR1) is not permitted. The slots of the modules from two segments cannot be assigned uniquely to the slots in a different, non-segmented rack, and vice versa. A CR2 rack therefore can be replaced only by a CR2 rack with a different order number, for example, to make it possible to insert redundant power supply modules without having to enter the rest of the configuration again.

When Is a C7 control System "Compatible"?

A C7 control system contains integrated modules (CPU and I/O) and can be exchanged for another one if the following rules are adhered to:

- For each slot (row in the configuration table) of the new C7 control system, the corresponding slot in the old C7 control system meets the following conditions:
  - Has no modules
  - Has an identical or exchangeable module (also applies to the CPU)
- The new C7 control system has just as many or more integrated modules as the old C7 control system.

When Is a DP Slave "Compatible"?

If DP slaves can be exchanged, then they can only be exchanged within the same family, for example, within the ET 200M family. DP slaves can also be displayed in NetPro; however, they can only be exchanged in the Configuring Hardware application.

The same basic rules for exchange apply to modular DP slaves as apply to racks: the number of inserted modules must have a place in the new DP slave and the functionality of the old DP slave must also be provided by the new DP slave.

If functions that are used by more than one station were configured, such as direct data exchange between DP slaves, you must provide this configuration to the new DP slave.

The following table provides information on each individual slave family:

DP Slave Type	Exchangeable	Conditions for Exchange
DP slaves whose properties are specified by GSD files (additional information...)	No	-
DP-As-i Link	No	-
DP/PA Link	Yes	Maximum configuration of the DP master cannot be exceeded (I/O address space, number of slots) A redundantly operated DP/PA link can only be exchanged for a DP/PA link that can be operated redundantly.
ET 200L SC and IM SC	Yes	You must be able to put all inserted modules into the new DP slave.
ET 200M (IM 153-x)	Yes	Cannot be exchanged if: • For the old ET 200M, the function "Change Modules during Operation" was activated and the new ET 200M does not support the function • CPs/FMs with their own MPI address (for example, FM 353) • Modules are inserted in the old ET 200M which cannot be inserted in the new ET 200M • The old ET 200M was operated redundantly and the new ET 200M cannot be operated redundantly
ET 200S (IM 151-1)	Yes	Cannot be exchanged with IM 151/CPU
ET 200S (IM 151/CPU)	Yes	Cannot be exchanged with IM 151-1
ET 200X (X-BM 141.., X-BM 142..)	Yes	You must be able to put all inserted modules into the new DP slave. An X-BM 143/DESINA cannot be exchanged for other ET 200X basic modules and vice versa Cannot be exchanged with X-BM 147/CPU
ET 200X BM 147/CPU	Yes	Cannot be exchanged with X-BM 141..., X-BM 142... and X-BM 143...

Which DP Slaves Are Specified by GSD files?

DP slaves whose properties are specified by GSD files (device database files) are located in the Hardware Catalog in the "Other Field Devices" folder or also in the "CP 342-5 as DP-Master" folder, for example. Those DP Slaves (also called "Standard Slaves") can be identified by specifying the GSD file or type file in the "Order no." row or in the "DP Slave properties" dialog, "General" tab.

1.5.3 Displaying Information on Components in the Hardware Catalog

In HW Config, you can access information on modules and components on the Internet.

Requirements

You programming device (PG) has an Internet connection and a browser installed to view Internet pages.

There is information on a module available on the Internet. (The exact extent of the information available will depend on the module and when you are searching for information.).

You have enabled the function in HW Config (menu command **Options > Customize)** and have selected and found a valid Internet page.

Procedure

1. In the hardware catalog or in the module rack, select the component for which you need information.
2. Right-click to display a shortcut menu with the following options:
   Product Support Information
   FAQs or
   Manuals

If an Internet page is available, it will contain links to FAQs or product information that have to do with the component selected. If you select "Manuals", a search is carried out for manuals or operating instructions that contain a description of the selected component.

Other Sources of Information

The Help page for a component property sheet or the module-related Help on a message also provides links to "More Information and FAQs". STEP 7 automatically searches for appropriate information on the Internet. The results of this search are displayed in the browser.

1.5.4 Installing Hardware Updates

As of STEP 7 V5.2., you can subsequently install components for the Hardware Catalog.

In this way you can use individual components such as new CPUs or new I/O devices in the current version of STEP 7 without having to install a new service pack.

Basic Information on Hardware-Updates

Install a HW update in the following two steps:

- Download the update from the Internet or copy it from the CD to the appropriate folder.
- Install the HW update that you downloaded or copied.

Once this is completed, these subsequently installed components are available in the Hardware Catalog **in the "Standard" catalog profile**. If necessary, you can follow this procedure so that you can also edit stations that were created with a later version of STEP 7.

Requirements

You must have an Internet address and a path set to the "HW Updates" storage folder. To make these settings, click the "Customize" button to open the dialog for configuring the HW update procedure.

You must have an Internet connection available to download updates.

If the Internet connection is established over a proxy server requiring authentication, you will need to know the user name and the password.

Procedure

1. Go to your STEP 7 program folder in the Start menu and select the entry **Install Hardware Updates** or select the menu command **Options > Install HW Updates** in HW Config**.**
2. In the dialog that appears, specify whether the HW update should be downloaded from the Internet or copied from a CD or
   whether you want to install updates that have already been download (selection available)
3. Select the component to be installed and then click the "Install" button.

2 Configuring Central Racks

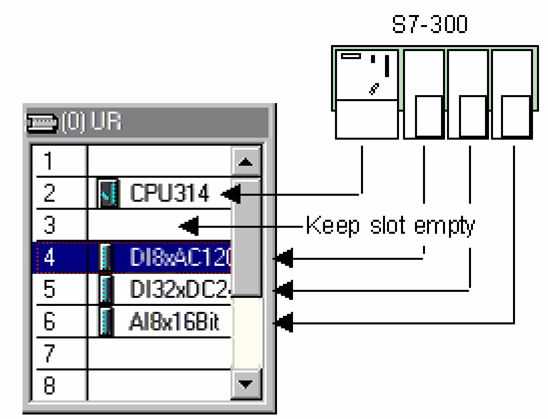
2.1 Rules for Arranging Modules (SIMATIC 300)

2.1.1 Rules for Arranging Modules (SIMATIC 300)

Basic Rule

Modules must be inserted in the rack without gaps.

Exception: For installations with one rack, one slot in the configuration table must remain free (reserved for the interface module). With the S7-300, this is slot 3 and with M7-300, the slot after the module group (slot 3, 4, 5, or 6). In the actual configuration there is no gap because the backplane bus would be interrupted.



Slot Rules (S7-300)

Rack 0:

- Slot 1: Power supply only (for example, 6ES7 307-...) or empty
- Slot 2: CPU only (for example, 6ES7 314-...)
- Slot 3: Interface module (for example, 6ES7 360-.../361-...) or empty
- Slots 4 through 11: Signal or function modules, communications processors, or free.

Racks 1 to 3:

- Slot 1: Power supply module only (for example, 6ES7 307-...) or empty
- Slot 2: Free
- Slot 3: Interface module
- Slots 4 through 11: Signal or function modules, communications processors (dependent on the inserted interface module), or free.

2.1.2 Special Rules for the Dummy Module (DM 370 Dummy)

A dummy module (DM 370 Dummy) is a module that you can insert instead of a module that will be used later.

Depending on the switch setting, the module may or may not reserve address space for a module. For example, address space is reserved for a digital input/output module but not for an interface module.

Switch Setting on DM 370 Dummy	Meaning	Order Number
A	Address space can be reserved. Modules in a modular DP slave ET 200M: reserve 0 bytes of address space.	6ES7 370-0AA00-0AA0
NA	No address space reserved.	None (Module is "not visible"; it is not configured)

2.1.3 Special Rules for the Digital Simulation Module (SIM 374 IN/OUT 16)

The SIM 374 IN/OUT 16 digital simulation module can be used to simulate digital inputs and outputs.

You **cannot** find this module in the "Hardware Catalog" window. You must place the module you want to simulate in the configuration table instead of the SIM 374.

Switch Setting on SIM 374 IN/OUT 16	Module to Place
16xOutput	6ES7322-1BH00-0AA0
8xOutput 8xInput	6ES7323-1BH00-0AA0
16xInput	6ES7321-1BH00-0AA0

2.1.4 Special Rules for M7-300

When you configure an M7-300 module group, each module in the group occupies a slot.

If the first module of the module group is an M7-300 CPU, then the next slot after the module group can be occupied only by either an interface module or it remains empty.

Module Group (M7-300)

An M7-300 module group is formed when expansion modules (EXM) or a mass-storage module (MSM) are used to extend an M7 CPU or an M7 FM (function module). All modules in the module group are connected with each other via their AT-ISA bus and form, in effect, the automation computer.

First arrange the basic module (M7 CPU or M7 FM) and then the expansion modules in the configuration table. Otherwise you will not be able to insert the expansion modules.

Arranging Modules in a Module Group (M7-300)

A module group results in new slot rules:

- The M7 CPU or the expandable M7 FM is always the **first** module in the module group.
- The mass-storage module (only one can be inserted) is always the **last** module within the module group.
- The M7 CPU or M7 FM may not be expanded by more than three modules (MSM or EXM). The permitted number of expansion modules is documented for the appropriate M7 FMs.

2.2 Rules for Arranging Modules (SIMATIC-400)

The rules for the arrangement of modules on a rack of the S7-400 depend on the type of the installed rack.

Central Rack

The following rules apply:

- Insert power supply modules only in slot 1 (exception: power supply modules with redundant capability)
- Insert a maximum of six interface modules (send IMs), not more than two with power transmission
- Connect a maximum of 21 expansion racks using interface modules to the central rack
- Connect not more than one expansion rack **with power transmission** to an interface of the send IM (IM 460-1 coupled with IM 461-1);
- Connect a maximum of four expansion racks **without power transmission** (IM 460-0 with IM 461-0 or IM 460-3 with IM 461-3).

Expansion Rack

The following rules apply:

- Insert power supply modules only in slot 1
- Insert the interface module (receive IM) only in the extreme right slot (slot 9 or slot 18).
- Communication bus modules should only be inserted in expansion racks with a number not greater than 6 (otherwise, they cannot be addressed).

2.2.1 Special Rules for Power Supply Modules with Redundant Capability (S7-400)

Power supply modules with redundant (standby) capability can be inserted in a rack twice. These modules can be recognized by their info text in the "Hardware Catalog" window.

You should observe the following rules:

- It is only possible to insert power supply modules with redundant capability in the racks intended for this purpose (these can be recognized by the higher order number and info text in the "Hardware Catalog" window).
- Power supply modules with redundant capability can be operated only with the CPUs that are designed for this purpose; CPUs that are not suited for this purpose (for example, older models) are rejected during configuration.
- The power supply modules with redundant capability must be inserted in slot 1 and the slot immediately next to it (no gaps allowed).
- Power supply modules with and without redundant capability cannot be inserted in the same rack (no "mixed" configurations possible).

2.2.2 Special Rules for M7-400

An M7-400 module group is formed when expansion modules (EXM, ATM) or a mass-storage module (MSM) are used to extend an M7 CPU or an M7 FM.

First arrange the basic module (M7 CPU or M7 FM) and then the expansion modules in the configuration table. Otherwise you will not be able to insert the expansion modules.

Arranging Modules in a Module Group (M7-400)

The following rules apply:

- Not more than one mass-storage module (MSM) can be inserted.
- The M7 CPU may not be expanded by more than three modules (EXM, ATM, or MSM).
- The modules of the module group can be arranged only in the following order to the right of the M7 CPU:
  - EXM module(s)
  - MSM module
  - ATM module(s).

2.2.3 Special Rules for PROFIBUS-DP Interface Submodules (M7-400)

If you wish to use an interface submodule for PROFIBUS-DP in an M7-400 programmable controller (for example, IF 964-DP as DP master), then you should note the following:

- No other interface submodule can be inserted in the CPU, FM, or EXM under this interface submodule.
- An interface of the CPU, FM, or EXM that is below this interface submodule cannot be used.

Reason: The PROFIBUS bus connector covers the module slot or interface below.

Recommendation: Insert an interface submodule for PROFIBUS-DP only in the lowest or in the lower left module slot of a CPU, FM, or EXM.

2.3 How to Configure Central Racks

2.3.1 Creating a Station

Requirement

You must have opened the SIMATIC Manager and opened a project or created a new project.

Procedure

A station can only be created directly beneath a project.

1. Select the project in the left half of the project window.
2. Select the menu command **Insert > Station > SIMATIC 300-Station** or **... > SIMATIC 400-Station**.
   The station is created with a preset name. You can replace the name of the station with a more relevant name of your own.

2.3.2 Starting the Application to Configure the Hardware

Requirement

You must have created a station (SIMATIC 300, SIMATIC 400).

Procedure

1. Select the "Station" object in the project window so that the "Hardware" object becomes visible in the right half of the station window.

	"Station" object

2. Double-click on the "Hardware" object.

	"Hardware" object

Alternatively you can select the "Station" object and then the menu command **Edit > Open Object**.

Result: A station window and the module catalog (if it was still open when you last exited the application) appear on the screen. You can position racks and other components in accordance with the station structure; select the components required for the station structure from the module catalog ("Hardware Catalog" window).

Opening More Stations

Using the menu command **Station > New** you can configure another station in the same project; using **Station > Open** you can open an existing (offline) station configuration for editing.

2.3.3 Arranging the Central Rack

Requirement

The station window must be open and you must have a plan of the hardware configuration of the station.

Procedure

1. Select a suitable central rack ("Rack") for your configuration from the "Hardware Catalog" window; in SIMATIC 300 the Rail, in SIMATIC 400 the Universal rack (UR1), for example.
2. Drag the rack to the station window.
   The rack appears in the form of a small configuration table in the upper part of the station window. In the lower part of the window, the detailed view of the rack appears with additional information such as the order number, MPI address, and I/O addresses.
   Alternatively to steps 1 and 2, you can also double-click the rack in the "Hardware Catalog" window.

2.3.4 Arranging Modules in a Rack

Requirements

You have arranged a rack in a station window and the rack is not shown minimized (the rack slots are visible).

If you want to have the visual advantage of having the available slots of a selected module highlighted in color in STEP 7, your monitor display must set for more than 256 colors.

Procedure

1. Select a module (for example, a CPU) from the "Hardware Catalog" window.
   The slots available for this module are highlighted in color.
2. Drag & drop the module into the appropriate row of the rack (configuration table). STEP 7 checks whether any slot rules are violated (an S7-300 CPU must not, for example, be inserted in slot 2).

🛇	Symbol that appears when the slot rules are violated.

3. Repeat steps 1 and 2 until the rack has been fully equipped with the required modules.

Alternatively, you can also select the appropriate row in the configuration table and double-click on the required module in the "Hardware Catalog" window.

If no row in the rack is selected and you double-click a module in the "Hardware Catalog" window, the module will then be placed in the first available slot.

Tip

After you have selected a slot of a module rack, you can open the context-sensitive menu (right-click) **Insert Object** or **Replace Object** to view a list of modules you can insert. This feature saves you the search through the Hardware Catalog. You select from all the modules that are listed in the currently available catalog profile.

Display of Interfaces and Interface Submodules

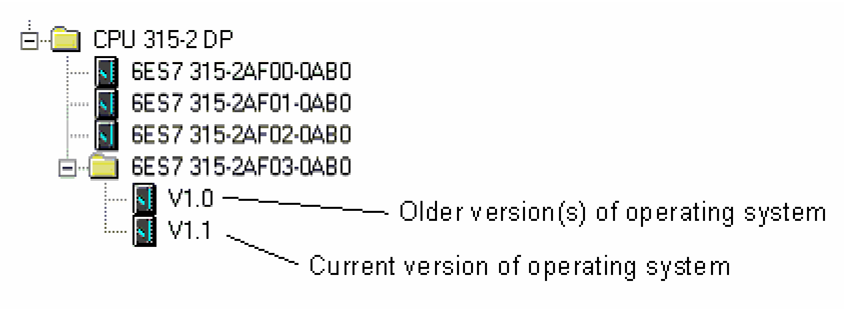
The interfaces and interface submodules are shown in the configuration table in a separate row. The row has the same name as the interface (for example, X1) or - if the module has slots for interface submodules - bears the prefix "IF" (for example, IF1).

With **integrated interfaces** the name of the interface appears in the "Module" column; with modules with slots for **interface submodules** you can copy a suitable interface submodule (IF) from the "Hardware Catalog" window to the appropriate row using drag & drop.

2.3.5 Displaying the Version of the CPU Operating System in the 'Hardware Catalog' window

When a CPU has more than one operating system this CPU is shown as a folder with different order number icons in the "Hardware Catalog" window. The different versions of the CPU operating system are arranged under the order number icons.

Please check the version of your CPU's operating system and select the correct version in the "Hardware Catalog" window.



Older version(s) of operating system

Current version of operating system

2.3.6 Arranging C7 Control Systems (Special Features)

In a C7 control system (C7-620), the following components are integrated in one housing:

- SIMATIC 300 CPU
- Inputs and outputs (digital and analog)
- Interface module IM 360 for connecting further SIMATIC 300 modules
- Line-oriented operator panel with a printer port

Simplified Procedure

The C7 control system is not mounted on a rail - this means you do not have to arrange a rack.

Requirement

The station window and "Hardware Catalog" window must be visible.

Procedure

1. Select a C7 control system from the "Hardware Catalog" window. These systems can be found under SIMATIC 300.
2. Drag the C7 control system to the station window.
3. If you want to expand the C7 control system:
   - Select rails as racks from the "Hardware Catalog" window.
   - Drag the racks to the station window one by one.
   - Assign modules to the racks. Important: The interface modules (IM) must be inserted in all racks so that connecting up is possible.

2.3.7 Assigning Properties to Modules/Interfaces

Introduction

Properties of components such as modules or interfaces are addresses and parameters. Only if you want to change the default values do you need to read the following sections.

Requirement

You have arranged in the configuration table the component for which you want to modify the properties.

Procedure

Every component (module, interface, or interface submodule) has default properties; for example, default measurement types and measuring ranges for analog input modules.

If you want to change these settings, proceed as follows:

1. Double-click in the configuration table on the component (for example, module or interface submodule) that is to have parameters assigned or select the row and select the **Edit > Object Properties** menu command.
   Using the right-hand mouse button: Move the mouse pointer to the component, press the right-hand mouse button, and select the **Object Properties** command from the pop-up menu
2. Use the displayed tabbed dialog boxes to assign the component properties.

2.3.8 Assigning Addresses

There is a difference between assigning addresses to nodes and assigning input/output addresses (I/O addresses).

Node addresses are addresses of programmable modules (MPI, PROFIBUS, Industrial Ethernet addresses); they are required in order to be able to address the various nodes in a subnet, for example, in order to download a user program to a CPU. You will find information on assigning node addresses in the subnet in the chapter on networking stations.

Input/output (I/O) addresses are required in order to read inputs and set outputs in the user program.

Special Feature: MPI Addresses of FMs and CPs (S7-300)

Communications processors (CPs) and function modules (FMs) with their own MPI address have a special feature: their MPI address is determined automatically by the CPU and assigned according to the following pattern:

- First CP/FM after the CPU: MPI address of the CPU + 1
- Second CP/FM after the CPU: MPI address of the CPU + 2

The newer S7-300 CPUs (see CPU manual or Product Information) permit free MPI address allocation for such CPs and FMs (via the "General" tab for the module).

2.3.9 Assigning I/O Addresses

STEP 7 assigns input and output addresses when modules are placed in the configuration table. This means every module has a start address (address of the first channel); the addresses for the remaining channels are based on this start address.

Requirements

- The module is inserted in a central rack or expansion rack and the CPU must permit free address assignment.
- The module is inserted in a DP slave or the module is a DP slave (compact DP slave).

Special feature: I addresses with asterisk (*)

In the detailed view of HW Config, certain addresses from the input area (I addresses) are marked with an asterisk (*). These are diagnostic addresses, for example of interface modules, that do not address user data but are required for system diagnostics.

Procedure

1. Double-click on the row of the rack containing the module whose start address you wish to set, or select the module and select the **Edit > Object Properties** menu command.
2. Select the "Addresses" tab.
3. Change the default start address.

Note

For modules within a local bus segment, formed by a function module (S7-300) or for special function modules (S7-400), you have to assign a further start address. In addition to the start address for the CPU, the module then has a start address for the FM. In the overall view of the configuration table, the start address from the point of view of the FM is always displayed in this case.

Displaying the Address Overview

You can display the input and output addresses already used and any address gaps as follows:

1. Open the station whose addresses you want to display.
2. Select the menu command **View > Address Overview**.
3. In the "Address Overview" dialog box, select the module whose assigned inputs and outputs you want to display (for example, CPU).
4. If required, you can filter the display by address type (for example, input addresses only).

The address areas "Inputs" and "Outputs" are displayed with locations for the modules (e.g. master system no., PROFIBUS address for PROFIBUS DP, rack, slot, interface submodule slot). Input addresses with the length 0 (for example, addresses of interface modules) are marked with an asterisk (*).

2.3.10 Assigning Symbols to I/O Addresses

Introduction

You can assign symbols to input/output addresses when you configure modules without having to go via the symbol table.

You can only assign symbols to the inputs and outputs of digital or analog modules when configuring the hardware. With integrated inputs/outputs (for example, CPU 312 IFM), in CPs, FMs, and S5 modules (for example, configured via the adapter module) you must assign the symbols in the symbol table.

The assigned symbols are not downloaded to the station (menu command: **PLC > Download**). The consequence of this is: when you upload a station configuration back into the programming device (menu command: **PLC > Upload**), no symbols are displayed.

Procedure

1. Select the digital/analog module to whose addresses you want to assign symbols.
2. Select the menu command **Edit > Symbols** or press the right mouse button and select the **Symbols** command in the pop-up menu.
   You can assign symbols in the dialog box that appears.
   If you click the "Add Symbol" button in the dialog box, the name of the address is entered as the symbol.

2.3.11 Monitoring Inputs and Modifying Outputs When Configuring Hardware

Introduction

If the CPU can be reached online and you have loaded the hardware configuration onto the CPU, you can address the inputs and outputs of the configured input and output modules directly, that is, without having to change the application.

Proceed as follows:

1. Highlight the module you want to monitor or control.
2. Select the menu command **PLC > Monitor/Modify.**
   The menu command can only be activated if the module can be monitored or modified (modules of the type DI, DO, AI, or AO, for example).
   A dialog box will open displaying the inputs and outputs of the module in a table (Address column).
   The inputs and outputs will be displayed in binary format for digital modules and in word format for analog modules.

Monitor

1. Activate the Monitor check box.
2. Click on the Trigger button to modify or change the trigger point and trigger frequencies. You will find information about the settings in the Help of the Trigger dialog box that appears when you click on the button.
3. If you want to monitor peripheral inputs directly, activate the I/O Display check box; if the check box is not activated, the process image of the inputs is monitored.
4. If you choose the trigger frequency once, you must update the display in the Status Value column by means of the Status Values button. The value remains "frozen" until you click again on the Status Value button.

Modify

With a defined trigger:

1. Enter the modify values in the table.
2. Click on the "Trigger" button to check or change the trigger point or trigger condition.
   You will find information about the settings in the Help of the "Trigger" dialog box that is displayed when you click the button.
3. Select the "Modify" check box. All visible addresses with a modify value will be modified.
4. Modifying variables once:
   Regardless of the trigger point or trigger condition set, you can also assign values to variables once in a separate operation. When the function is activated, an action like "Trigger immediately" is carried out, without reference to a specific point in the user program.
5. Enter the modify values in the table.
6. Click the "Modify values" button.
7. If you want to modify peripheral outputs directly, activate the I/O Display check box; if the check box is not activated, the process image of the outputs is controlled.
8. If you also want to modify outputs directly in STOP mode of the CPU, you must activate the Enable PQ check box. Otherwise, the outputs remain set in STOP mode, and they have a configured substitute value.

Monitoring Modified Addresses

Note that the display in the "Status Value" column depends on the trigger points set and on the processes in the CPU (for example, the update times of the process images).

You should set the trigger point for monitoring to "Cycle Start" and the trigger point for modifying to "Cycle End", so that the modified value can be displayed in the "Status Value" column!

Monitoring and Modifying I/O Modules

When monitoring an I/O module (for example, 8DI/8DO), note that the trigger applies to the entire module.

If, for example, you set the trigger point for monitoring to "Cycle Start", the inputs as well as the outputs are modified at that time. In this case, the control values for the inputs are activated, because they overwrite the process image of the inputs **after** the process image of the inputs has been updated; this means, directly before the cyclic program processing of the input process images was started. In this case, the values for the outputs are overwritten by the user program.

Monitoring and Modifying Distributed I/Os

Distributed I/Os can be read consistently with the SFC 14 "DPRD_DAT" and can be written consistently with the SFC 15 "DPWR_DAT". In order to access the actual status or values when monitoring, you must note the following:

- The "RECORD" input parameter of the SFCs has to be provided with "I" (input) or "O" (output). This must be exactly the configured address area that is displayed in the configuration table of the DP slave in the "I address" or "O address" column.
- If different areas are selected for saving the consistent data, non-relevant areas of the process image are displayed in the table for monitoring.

2.3.12 Configuring Point-to-Point CPs

Introduction to Configuring Point-to-Point CPs (PtP CPs)

As with other modules, you drag communication processors from the Hardware Catalog window and arrange them and assign parameters to them in the configuration table (General Settings, Addresses and Basic Parameters).

You can start the optional software for setting the procedure parameters by means of the Parameters interface in the Basic Parameters tab.

For PtP CPs in a SIMATIC 400 station, you have to configure PtP connections. The following section describes the procedure briefly. You can find a more detailed description in the manuals for the PtP CPs.

Configuring Connections for S7-400 PtP CPs

For the connection between an S7 CPU and a partner that is connected by means of a point-to-point interface, the PtP CP represents the link.

You must configure connections between the S7-400 CPU and the PtP CP.

Proceed as follows:

1. In HW Config, configure the PtP CP and, if the connection partner is also a PtP CP in an S7-400 station, configure the partner station also.
2. Start NetPro (from HW Config: menu command **Options > Configure Network**).
3. If you have already created a PtP subnet while you were configuring the PtP CP and you have networked the CP, continue with step 4;
otherwise, insert a PtP subnet and use it to network the CP.
4. Select the connection partner:
   - The connection partner is a CP 34x (PtP CP in an S7-300 station), an S5 CP PtP, a printer, or a device from another manufacturer:
     Configure an "Other Station" as connection partner. For the "Other Station," configure a PtP interface and network this interface with the PtP subnet that you created.
   - The connection partner is a PtP CP for a SIMATIC 400 station:
     You have already configured the connection partner in step 1 and so you can continue with step 5.
5. Configuring PtP Connections
   - The connection partner is a CP 34x (PtP CP in an S7-300 station), an S5 CP PtP, a printer, or a device from another manufacturer:
     Configure the PtP connection for the local CP (with "Other Station" connection partner).
   - The connection partner is a PtP CP for a SIMATIC 400 station:
     Configure the PtP connection for the local CP and for the partner station
6. Load the configuration data and the connection data into the appropriate stations.

2.3.13 Configuring S5 Modules

You can insert S5 modules in a SIMATIC 400 station. These modules are connected via:

- An S5 adapter module (IM 470) or
- An IM 463-2 for connecting S5 expansion devices with IM 314

You can find these modules in the "Hardware Catalog" window under "IM 400."

Note
You must configure the input or output address areas of the S5 modules for each connection. (Double-click the adapter module or IM 463-2 and then select the "Input Addresses" or "Output Addresses" tab.)

If the address areas are not configured, the modules mentioned above will not be saved in the system data blocks (SDB). **Result:** The configuration which was downloaded to the CPU will not contain information about these modules. If this configuration is uploaded to the programming device, these modules will not appear in the configuration table.

2.4 Expanding the Central Rack with Expansion Racks

Configuring Expansion Racks in SIMATIC 300

For SIMATIC 300 stations, only "Rails" are available as central racks and as expansion racks; this means you can position as many rails (max. 4) as there are in the actual configuration.

Expansion racks are linked in STEP 7 by inserting the appropriate interface module in slot 3 of every rack.

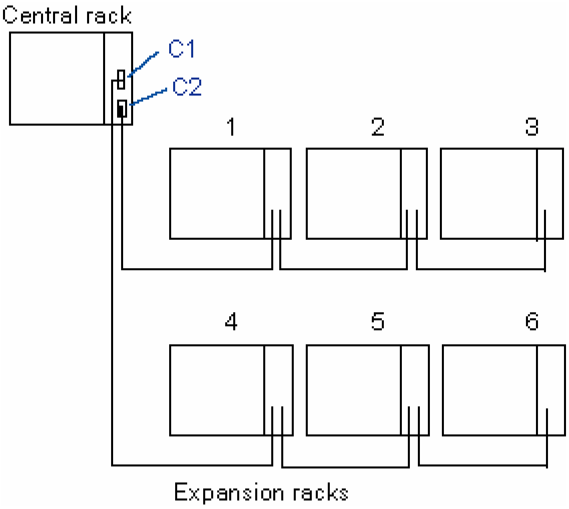
- To expand the configuration by one rack:
  racks 0 and 1: IM 365
- To expand the configuration by up to three racks:
  rack 0: IM 360; racks 1 to 3: IM 361

Configuring Expansion Racks in SIMATIC 400

In SIMATIC 400, the possibilities for expansion are more complex owing to the different racks and interface modules.

All expansion racks that are connected to an interface of the send IM on the central rack form a **cascade**.

In the following figure, three expansion racks are connected to each interface of the send IM.



2.4.1 Rules for Connecting Expansion Racks (SIMATIC 400)

If you connect expansion racks (SIMATIC 400) to an interface module (send IM) of the central rack, then the following characteristics for the send IM and the receive IM must match:

- Power transfer (with/without)
- Type of link (centralized/distributed)
- Communication bus transmission (with/without interrupt transmission).

2.4.2 Arranging the Expansion Rack (SIMATIC 400)

Procedure

1. Select the appropriate (expansion) racks from the "Hardware Catalog" window.
2. Drag the racks to the station window one by one.
3. If you wish to change the number of the rack:
   Double-click the second row of the rack in the upper half of the station window. You can change the number in the "General" tab for the rack.
4. Assign modules to the racks.
   Important: To permit the racks to be connected to each other, the interface modules must be inserted in all racks.
5. **For S7-400 only**: make the connections between the interface modules in the racks:
   - Double-click on the send IM.
   - Select the "Connect" tab
     This tab shows all racks that are not connected (racks with inserted receive IMs).
   - Select the individual racks and use the "Connect" button to connect them to the required interface of the send IM (C1 or C2).
     Connection lines then show how the racks are connected together.

2.4.3 Special Case: When the Central Rack Has More Than One CPU

If you wish to extend the configuration which consists of the segmented rack CR2 (S7-400) or a multicomputing configuration with racks you must adhere to the following order:

1. Configure the central rack (for example, CR2) with the send IM.
2. **Only** insert Receive IMs in the expansion racks.
3. Remove the connections between the interface modules (IMs) as described above.

Only then can you insert modules in the expansion racks. Reason: Because the address space exists more than once for multiple CPUs, the expansion rack must first be assigned to an address space (= a CPU).

3 Configuring the Distributed I/O (DP)

3.1 Configuring the Distributed I/O (DP)

Introduction

The distributed I/O refers to master systems, comprising DP (distributed I/O) master and DP slaves that are connected via a bus cable and communicate with each other via the PROFIBUS-DP protocol.

As DP masters and DP slaves can be different devices, this section only explains the basic procedures involved in configuring. You will find details on functionality, access procedures etc. in the manuals for the specific devices and in the online help for the special FCs (for example, DP-SEND and DP-RECEIVE for CP 342-5).

3.2 Basic Procedure for Configuring a DP Master System

If you know the principles of how to configure a central structure, you also know how to configure the distributed I/O - the procedure is the same to a large extent.

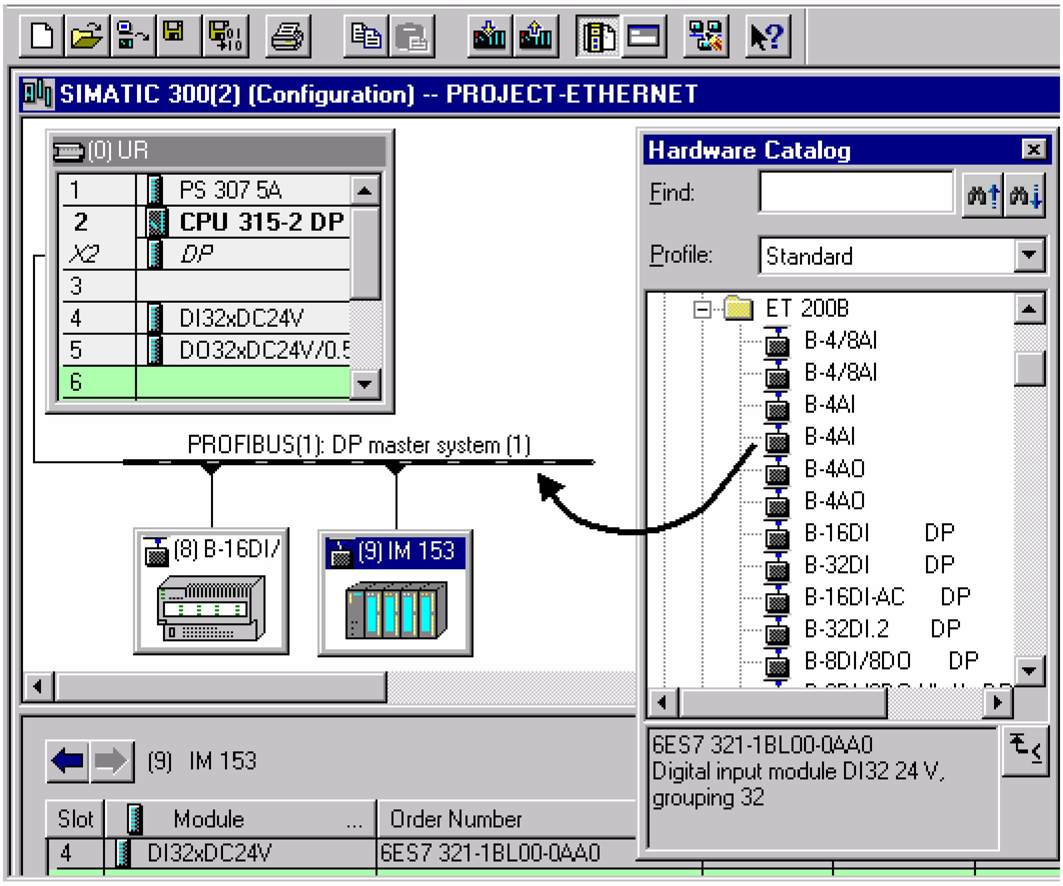
Station Window as an Image of the Real DP Master System

When you place a DP master (for example, a CPU 315-2DP), STEP 7 automatically draws a line that represents the master system. At the end of the line, place the DP slaves that are assigned to this DP master using drag & drop from the "Hardware Catalog" window under "PROFIBUS-DP."

As a DP master system is always connected to a PROFIBUS subnet, STEP 7 automatically displays dialog boxes for setting the subnet properties (for example, transmission rate) and the PROFIBUS address when you place the DP components.

DP Slave Does Not Appear in the "Hardware Catalog" Window

If a DP slave does not appear in the "Hardware Catalog" window, you must install the corresponding *.GSD file after the start of STEP 7 using the menu command **Options > Install GSD File**. The dialog boxes then guide you through the process of installing the *.GSD file. The installed DP slave then appears in the "Hardware Catalog" window under "PROFIBUS-DP - Other Field Devices."



SIMATIC 300(2) (Configuration) -- PROJECT-ETHERNET

(0) UR

1		PS 307 5A
2		CPU 315-2 DP
X2		DP
3		
4		DI32xDC24V
5		DO32xDC24V/0.5
6		

PROFIBUS(1): DP master system (1)

(9) IM 153

Slot		Module	...	Order Number
4		DI32xDC24V		6ES7 321-1BL00-0AA0

Profile: Standard

6ES7 321-1BL00-0AA0
Digital input module DI32 24 V, grouping 32

Slave Configuration in the Detailed View

If you select the DP slave, the slave structure (DP identifiers and modules/submodules) and I/O addresses are displayed in the detailed view of the station window.

Switching Between the DP Master System and the DP Slave in the Detailed View of the Station Window

If you select the symbol for the DP master system (────▭▭▭▭), all DP slaves belonging to the DP master system will be displayed in the lower half of the station window. If you select a DP slave symbol, the configuration of the DP slave is displayed in the lower half of the station window. You can easily change between displays by using the following buttons:

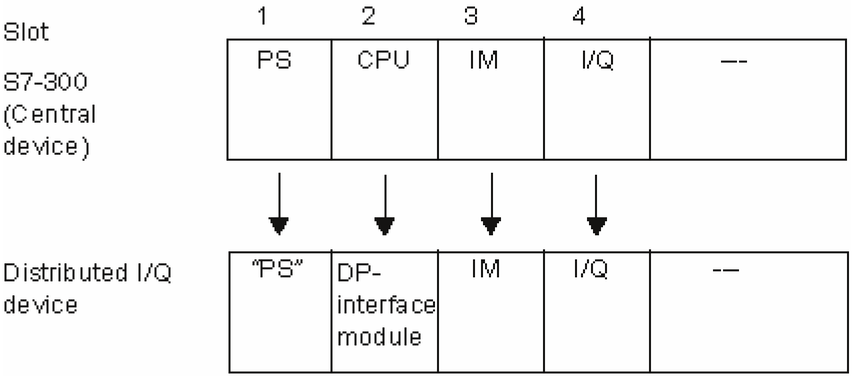
⬅ and ➡

Slot Numbering in Distributed I/O Devices

Depending on which DP slave type you are configuring, the slots in the detailed view of the DP slave begin either with "0" or "4."

With DP slaves that are configured by *.GSD files, the *.GSD file specifies at which slot the I/O addresses start; the slots before that are "empty."

Slot numbering of DP slaves such as ET 200M that are completely integrated in STEP 7 is derived from the structure of an S7-300 station according to the following pattern:



Remarks on the slots of a DP slave, which is structured like the S7-300 station:

- The "actual" I/O (inputs/outputs) always starts with slot 4.
- Independent of whether a power supply module (PS) is inserted in the real configuration or not: slot 1 is always reserved for a PS.
- Slot 2 is always reserved for the DP interface module.
- Slot 3 is always reserved for an expansion interface module (IM), independent of whether a "real" I/O device can be expanded or not.

This pattern is used for all DP slave types: modular and compact. The slot arrangement is important for evaluating diagnostic messages (the slot that triggers diagnostics).

3.3 Where Are the DP Slaves in the Hardware Catalog Window?

All DP slaves can be found in the "Hardware Catalog" window under the "PROFIBUS-DP" folder.

Here, the following points should be noted. In these paragraphs, CP 5611 and CP 5613 are used to mean groups of CPs:

- "CP 5611" stands for all CPs of types CP 5611, CP 5611 A2, CP 5511, CP 5512
- "CP 5613" stands for all CPs of types CP 5613, CP 5614, CP 5613 FO, CP 5614 FO, CP 5613 A2, CP 5614 A2

The DP master is...

- ...a SIMATIC 300 or SIMATIC 400 CPU with integrated PROFIBUS-DP interface, a PROFIBUS CP (without CP 342-5DA00) in a SIMATIC 300/400 or a SIMATIC PC station with a PROFIBUS CP (without CP 5611/CP 5613)

  The DP slaves appear under their "family name" (e.g. PROFIBUS-DP\ET 200B).
- ...a CP 342-5DA00 with PROFIBUS-DP interface or an SIMATIC PC station with CP 5611/CP 5613:
  The DP slaves appear either in the "DP V0 Slaves" folder and then under their family name (e.g. PROFIBUS DP\DP V0 Slaves\ET 200B).

The "DP V0 Slaves" contains the DP slaves represented by their GSD file or type file (referred to as "Standard Slaves").

Folders designated with the family name and located directly under PROFIBUS DP (e.g. PROFIBUS DP\ET 200B) usually contain the DP slaves whose properties are represented by internal STEP 7 information (referred to as "S7 Slaves").

The DP slave was purchased separately (with a new *.GSD file)

The DP slave will appear under "Other Field Devices" after you have installed the *.GSD file.

The DP slave is an intelligent DP slave

Examples: You can configure stations as a DP slave with:

- CP 342-5 DP
- CPU 315-2 DP, CPU 316-2 DP, CPU 318-2 DP
- Basic submodule ET 200X (BM 147/CPU)
- IM 151/CPU (ET 200S)

The DP slave appears under the "Configured Stations" folder after configuring the station. The procedure (how does a station appear in the "Configured Stations" folder?) is described in detail in the section about configuring the intelligent DP slaves.

3.4 Integrating Tools for Configuring Distributed Devices (Tool Calling Interface)

As of STEP 7 V5.4 SP2, external tools can be seamlessly integrated in STEP 7 for configuration and diagnostics of distributed devices ("Device Tools").

These tools provide features that go beyond the capabilities of GSD configuration, for example, they offer options for advanced graphical input.

The following are considered distributed devices:

- PROFIBUS DP slaves
- Modules within a DP slave
- PROFINET IO devices

Modules within an IO device

Requirement

- The TCI specification is sufficient for the calling interface of the tool.

  HW Config parameters and commands can be sent to the distributed device via this call interface (Tool Calling Interface).
- The tool and STEP 7 must be installed on the same computer.

Procedure

1. Install the tool on the computer where STEP 7 is installed.
2. Install the GSD file of the distributed device.
3. Configure the distributed device.

   The network setting is especially important, for the tool to later use the online access capability of STEP 7.
4. Select the device you want to configure with the tool in HW Config.
5. Start the tool from HW Config with the menu command **Edit > Start Device Tool**.

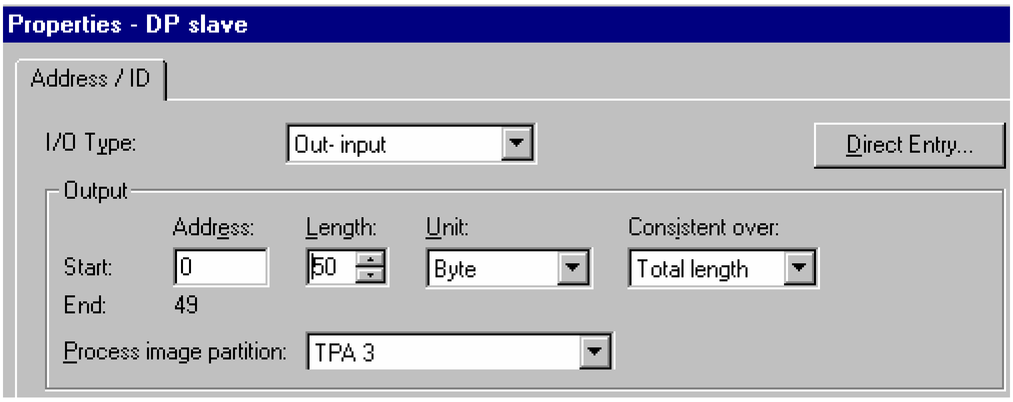
3.5 Distributed Reading and Writing of Consistent Data (> 4 Bytes)

Previously, you were able to access consistent data (> 4 Bytes) on a DP slave by means of SFC 14 and SFC 15.

As of Version 3.0 of CPU 318-2 and CPUs 41x, you can also access a consistent data area by accessing the process image (such as L IW).

Configuring Consistent Data Areas > 4 Bytes

1. In the hardware configuration, select the "Addresses" tab for the DP slave.
   Depending on the type of DP slave, the consistency area either has standard default settings and cannot be changed (such as specified by a GSD file), or you can define the consistency area in the fields "Length", "Unit" and "Consistency via".
2. If necessary, define the length of the consistent area and enter this area into a process image. To do this, proceed as follows: In the "Process image partition" field, select OB1-PI or, in the case of S7-400, select yet another process-image partition (PIP) such as PIP 3. If you do not put the data in a process image partition, then you must use SFC 14 or SFC 15 for data exchanges.



Properties - DP slave

Address / ID

I/O Type: Out- input

Output

Address: Start: 0, End: 49

Length: 50

Unit: Byte

Consistent over: Total length

Process image partition: TPA 3

Upon update of the process image, the operating system transfers these data consistently, and you can use download and transfer commands to access the data in the process image. This provides you with an easy and efficient (low run-time load) way to access consistent data.

3.6 Configurations for PROFIBUS-DP

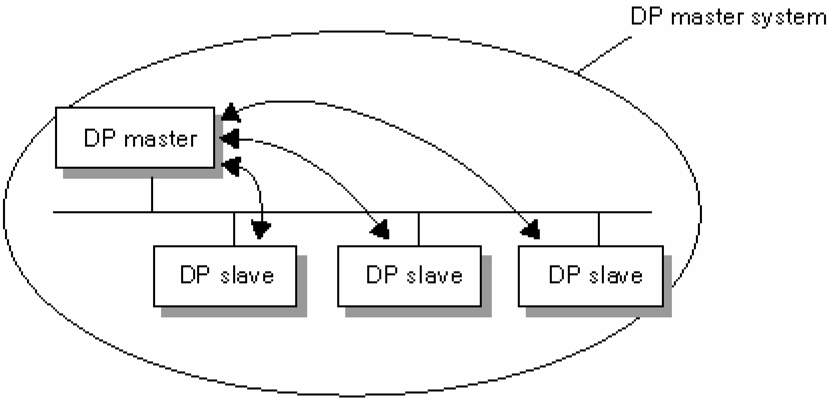
The following topics provide examples of configurations for PROFIBUS-DP that you can create with STEP 7.

- Configuration with "Simple" (Modular or Compact) DP Slaves (Data Exchange: Slave <> Master)
- Configuration with Intelligent DP Slaves (Data Exchange: Slave <> Master)
- Configuration with Intelligent DP Slaves (Direct Data Exchange: Slave > I Slave)
- Configuration with Two DP Master Systems (Direct Data Exchange: Slave > Master)
- Configuration with Two DP Master Systems (Direct Data Exchange: Slave > I Slave)
- Example of Configuring Direct Data Exchange (Lateral Communication)

3.6.1 Configuration with 'Simple' (Modular or Compact) DP Slaves (Data Exchange: Slave <> Master)

In this configuration, data exchange takes place between the DP master and simple DP slaves, that is, I/O modules, by means of the DP master. The DP master polls each configured DP slave, one after the other, in the master's polling list within the DP master system and transmits the output data or receives the input data from the slaves. The I/O addresses are assigned automatically by the configuration system.

This configuration is also known as a mono-master system because a single master with its associated DP slaves are connected to one physical PROFIBUS DP subnet.

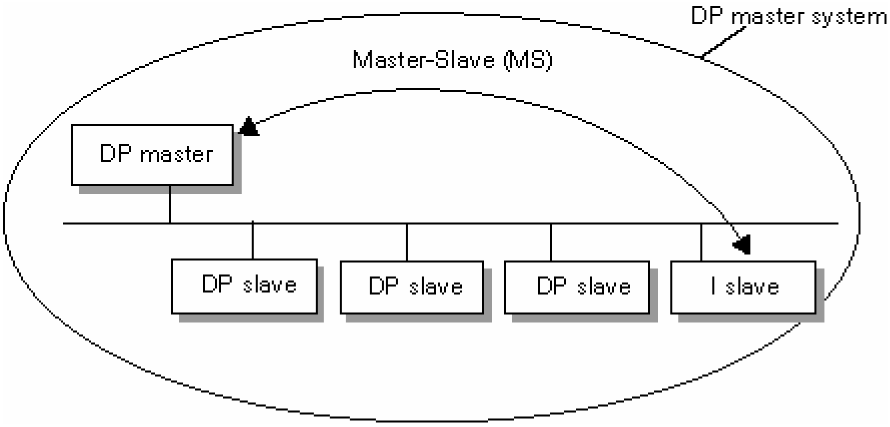


3.6.2 Configuration with Intelligent DP Slaves (Data Exchange: I Slave <> Master)

You can divide automation tasks into subtasks that can be controlled by a higher-level programmable logic controller. These control tasks, which can be accomplished independently and efficiently, run as preprocessing on a CPU. This CPU can be implemented in the form of an intelligent DP slave.

For configurations with intelligent DP slaves (I slaves), such as a CPU 315-2DP, the DP master does not access the I/O modules of the intelligent DP slave; instead, the master accesses only the address areas of the CPU of the I slave. This means that this address area cannot be assigned for real I/O modules in the I slave. This assignment must be made when the I slave is configured.

Examples of intelligent DP slaves (= DP slaves with preprocessing): station with CPU 315-2DP, CPU 316-2DP, CPU 318-2DP

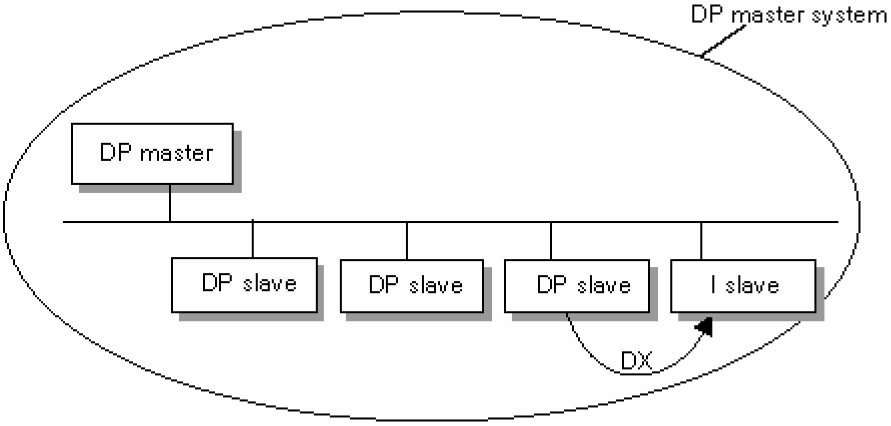


3.6.3 Configuration with Intelligent DP Slaves (Direct Data Exchange: Slave > I Slave)

With this configuration, input data from DP slaves can be transmitted very rapidly to intelligent DP slaves on the PROFIBUS-DP subnet.

In this manner, all simple DP slaves (as of a certain release) or other intelligent DP slaves can, in principle, provide selected input data for direct data exchange (DX) between DP slaves. Only intelligent DP slaves such as CPU 315-2DP can be used to receive these data.

Example of stations that can be configured as intelligent DP slaves: CPU 315-2DP, CPU 316-2DP, CPU 318-2DP

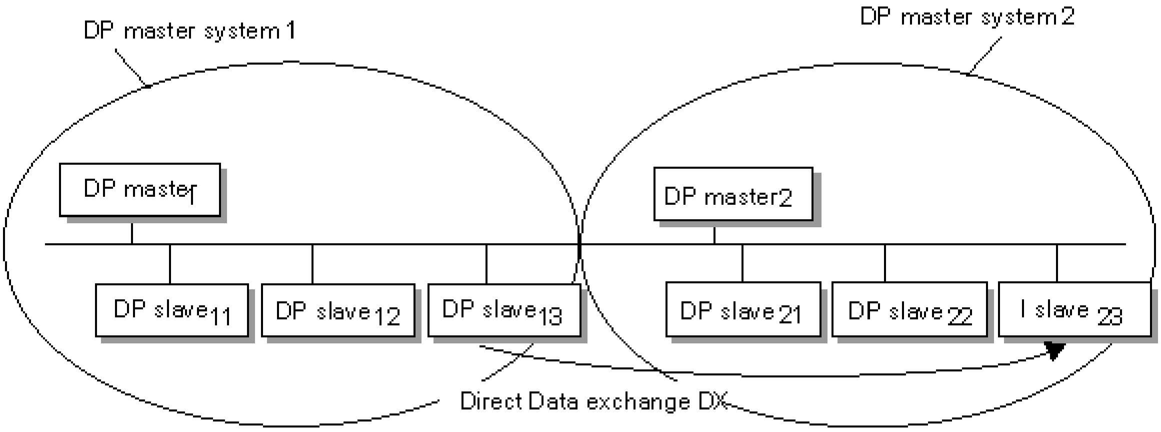


3.6.4 Configuration with Two DP Master Systems (Direct Data Exchange: Slave > I Slave)

Several DP master systems on one physical PROFIBUS-DP subnet are also called a multiple master system. In this configuration, input data from DP slaves can be read very rapidly by intelligent DP slaves on the same physical PROFIBUS-DP subnet. The intelligent DP slaves can be located on the same DP master system or on another system.

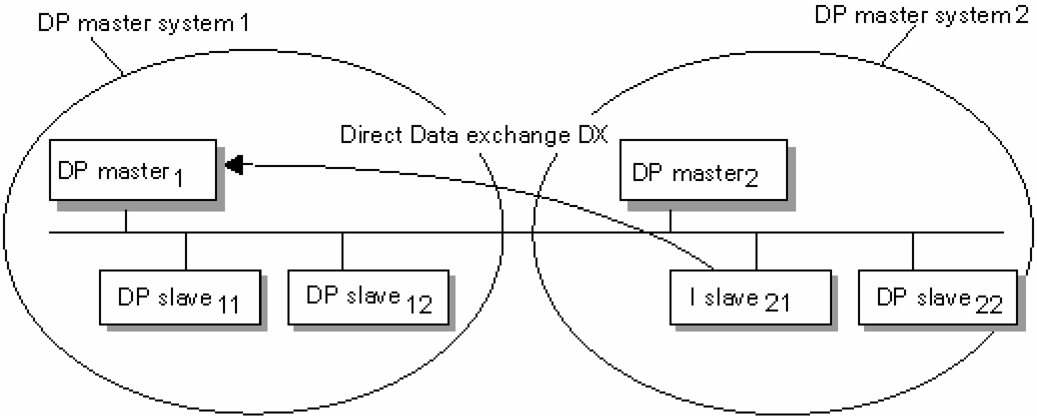
In this manner, an intelligent DP slave such as a CPU 315-2DP can allow input data from DP slaves to be transmitted directly to its input data area, even from different DP master systems (that is, multiple master systems).

In principle, all DP slaves (as of a certain release) can provide selected input data for direct data exchange (DX) between DP slaves. These input data can in turn be used further only by intelligent DP slaves such as CPU 315-2DP.



3.6.5 Configuration with Two DP Master Systems (Direct Data Exchange: Slave > Master)

Several DP master systems on one physical PROFIBUS-DP subnet are also called a multiple master system. In this configuration, input data from intelligent DP slaves or simple DP slaves can be read directly by the master of a different DP master system on the same physical PROFIBUS-DP subnet.



3.7 How to Configure Distributed I/O Systems

3.7.1 Creating a DP Master System

Requirement

You have arranged a rack in a station window and the rack is shown as being open (the rack slots are visible).

DP Master

You can use the following items as DP master:

- A CPU with a fixed integrated or installable DP master interface (fixed integrated, for example, CPU 315-2 DP).
- An interface submodule that is assigned to a CPU/FM
  (for example, IF 964-DP in CPU 488-4).
- A CP in conjunction with a CPU (for example, CP 3425, CP 4435).
- An interface module with DP master interface (for example, IM 467)

Procedure

1. Select a DP master from the "Hardware Catalog" window (for example, CPU 315-2 DP).
2. Drag the module to a suitable row in the rack. The "Properties - PROFIBUS Node" dialog box opens.
   Here you can do the following:
   - Create a new PROFIBUS subnet or select an existing subnet
   - Set properties for the PROFIBUS subnet (transmission rate etc.)
   - Set the PROFIBUS address of the DP master
3. Confirm the settings with "OK."
   The following symbol appears: ——▭▭▭▭▭
   This symbol is the "anchor" for the DP slave of the master system.

Tip: If you cannot find the symbol immediately, it may be hidden by the configuration table. Make the width of the configuration table containing the DP master smaller. If the symbol for the DP master system is still not visible, select the menu command **Insert > Master System**.

3.7.2 Handling DP Master Systems and DP Interfaces

Separating DP Master Systems

If you insert a CPU with an integrated PROFIBUS-DP interface or a PROFIBUS CP in a station (configurable as an intelligent DP slave) and you have configured the CPU or CP as a DP master with master system, you can separate the master system from the DP master:

1. Select the DP master interface
2. Select the menu command **Edit > Master System > Separate**.
   Alternative: Use the right mouse button to get a pop-up menu and select the command **Master System > Separate**.

The master system is maintained as an "orphaned" master system and is visible in the station. The configured direct data exchange (lateral communication) is maintained.

If no slaves are connected to the master system, the master system is deleted.

Inserting a DP Master System

If you configured one or more DP master systems and separated them from the DP master interface, you can insert one of the orphaned DP master systems again at the selected DP master interface by means of the menu command **Edit > Master System > Insert**.

Giving the PROFIBUS-DP Interface Online Capability

If you want the integrated PROFIBUS-DP interface of a CPU to behave like an active node on the PROFIBUS-DP (thus enabling programming device functions by means of this interface), make sure that the following conditions are met:

1. The PROFIBUS-DP interface must be configured as "networked," that is, when the PROFIBUS-DP interface is selected, a PROFIBUS subnet must be selected or newly created by means of the menu command **Edit > Object Properties**.
2. This configuration must be loaded to the CPU.
   Afterwards, such functions as monitoring with the programming device are possible by means of the PROFIBUS-DP interface.

Changing the Properties of a Master System

From STEP 7 V5.0, Service Pack 3 onwards you can change the properties of a master system (name and number):

1. Double-click the line representing the master system in an open station configuration.
2. Select the "General" tab and adapt the name and master system number to your requirements.
   You can also edit the corresponding subnet using the "Properties" button.

3.7.3 Selecting and Arranging DP Slaves

Types of DP Slaves

When configuring DP slaves, we differentiate between:

- Compact DP slaves
  (modules with integrated digital/analog inputs and outputs, for example, ET 200B)
- Modular DP slaves
  (interface modules with S5 or S7 modules assigned, for example, ET 200M
- Intelligent slaves (I slaves)
  (S7-300 stations with, for example, CP 342-5, CPU 315-2DP, or ET 200X with BM 147/CPU)

Note

Note the technical specifications (max. number of nodes, max. number of slots, max. number of user data) of the DP master when configuring the master system. It is possible that you may not be able to configure the maximum number of nodes owing to the limits for the number of slots or user data.

Requirement

A DP master system must be present and visible in the station window.

Symbol for the DP master system:

If the symbol is not visible (for example, it was deleted), you can create it by selecting the row for the DP interface of the DP master and selecting the menu command **Insert > Master System**.

3.7.4 Copying Multiple DP Slaves

1. Hold CTRL pressed and click on the DP slaves you want to copy one by one.
   Result: The DP slaves are selected.
2. Select the menu command **Edit > Copy**.
3. Select the DP master system to which the copied DP slaves are to be added.
4. Select the menu command
   Edit > Paste ("normal" copying) or
   Edit > Redundant Paste (when copying for software redundancy).

3.7.5 Configuring Compact DP Slaves

Procedure

1. Select a compact DP slave (for example, ET 200B) from the "Hardware Catalog" window.
2. Drag the DP slave to the following symbol for a DP master system: ─▭▭▭▭▭
   The "Properties - PROFIBUS Node" dialog box opens. Here you can set the following:
   - The properties for the PROFIBUS subnet (transmission rate etc.)
   - The PROFIBUS address of the DP slave.
3. Confirm the settings with "OK."
   Result: A symbol is attached to the DP master system to represent the compact DP slave. The I/O configuration of the compact DP slave is displayed in the lower half of the station window (detailed view).

3.7.6 Configuring Modular DP Slaves

Procedure

1. Select an interface module for a modular DP slave (for example, IM 153 for ET 200M) from the "Hardware Catalog" window.
2. Drag the interface module to the following symbol for the DP master system: ─▭▭▭▭▭
   Result: The "Properties - PROFIBUS Node" dialog box opens. Here you can set the following:
   - The properties of the PROFIBUS subnet (transmission rate, etc.).
   - The PROFIBUS address of the DP slave.
3. Confirm your settings with "OK."
   A symbol for the DP slave is attached to the DP master system. The detailed view of the DP slave appears in the lower half of the station window showing all the possible slots or DP identifiers.
4. Assign the modules for the modular DP slave in the lower half of the station window.
   For modular DP slaves, the possible modules are arranged in the "Hardware Catalog" window below the associated DP slave "family."
   These are e.g.:
   - Terminal blocks (TB...SC) for Smart Connect (family ET 200L SC)
   - SC submodules (family ET 200L SC)
   - AS-i slaves (family DP/AS-i Link)
   - S7-300 modules (family ET 200M)

3.7.7 Assigning DP Slaves to SYNC or FREEZE Groups

A DP master with the appropriate function can send the SYNC and/or FREEZE control commands simultaneously to a group of slaves to synchronize the DP slaves. To do this, you must assign the SYNC and FREEZE groups for the DP slaves.

Requirement

You must have created a DP master system.

Procedure

1. Select the DP master system symbol containing the DP slave that you wish to assign to a group.
2. Select the menu command **Edit > Object Properties**.
   Result: The "Group Assignment" tab appears with a table in which you can assign the SYNC/FREEZE groups for the DP slave.

Note

You can assign a maximum of one SYNC and one FREEZE group to each DP slave.
Exception: If a CP 3425 is used as DP master, a maximum of eight groups (SYNC and/or FREEZE groups) can be assigned to each DP slave of this master system (see CP 342-5 documentation).

What You Should Know About the SYNC and FREEZE Control Commands

The SYNC and FREEZE commands can be used to perform event-driven synchronization of the DP slaves. The DP master sends the control commands simultaneously to a group of DP slaves of its master system. Those slaves that have failed or are currently reporting diagnostics are ignored.

The requirement for synchronization using control commands is that you have assigned the DP slaves to SYNC and/or FREEZE groups.

For an S7 CPU, use SFC11 (DPSYC_FR) to synchronize the DP slaves.

SYNC Control Command

The DP master uses the SYNC control command for a group of DP slaves to freeze the states of their outputs at their current values.

In the following frames, the DP slaves store the output data of the DP master, but the states of the DP slaves' outputs remain unchanged.

After every new SYNC control command, the DP slave sets its outputs to the values that it stored as output data of the DP master.

The outputs are only then cyclically updated again when the DP master sends the UNSYNC control command.

FREEZE Control Command

After receiving the FREEZE control command from the DP master, the DP slaves of a group freeze the current state of their inputs and transfer these cyclically to the DP master.

After every new FREEZE control command, the DP slave freezes the states of their inputs again.

The input data are only then cyclically transferred from the DP slave to the DP master when the DP master sends the UNFREEZE control command.

3.8 Other DP Slave Configurations

3.8.1 ET 200L and DP/AS-i Link

When configuring the DP slaves ET 200L and DP/AS-i Link (distributed I/O/actuator-sensor interface), the following applies:

- ET 200L can be expanded using Smart Connect (SC) a channel at a time
- DP/AS-i Link is configured with actuator-sensor interface slaves; see below.

When placing a DP/AS-i Link, a configuration table is displayed automatically in which you can place the actuator-sensor interface slaves from the "Hardware Catalog" window.

3.8.2 ET 200S

Introduction

DP slaves and IO devices of the ET 200S family are configured like other modular DP slaves and IO devices.

Special feature: Digital electronic modules with an address space of 2 or 4 bits initially occupy 1 byte when they are inserted in the configuration table (Details view). However, the occupied address space can be compressed after configuration by means of the Pack Addresses button.

Example:

	Before Packing Addresses	**After Packing Addresses**
Module	**I Address**	**I Address**
DI_1_Module	10.0 to 10.1	10.0 to 10.1
DI_2_Module	11.0 to 11.1	10.2 to 10.3

Note

Addresses cannot be packed with the GSD variant of the IM 151-3AA00.

General Procedure: "Packing" Address Spaces Together

1. Select **one** contiguous range of modules whose addresses are to be packed.
2. In the Details view of the configuration table, click on the Pack Addresses button.

   The address areas for inputs, outputs, and motor starters are packed separately.

   - The beginning of the address area is defined by the address of each module that is selected first and has the form X.0.
   - If the bit address is not 0, the next (available) byte address from which the selected area can be inserted is used automatically, for example, (X+1).0.
   - If there is no other contiguous range, packing takes place automatically into available address spaces.

Procedure for GSD Variants: "Packing" Address Spaces

If you are using GSD variants, the "Pack Addresses" button does not exist. It is nevertheless possible to pack addresses by following the steps below:

1. Configure a "normal" module.
2. Configure up to three "packed" modules under the "normal" module.
   You will recognize the packed modules by the asterisk (*) after the short name of the module in the hardware catalog. All modules (for which packing is possible) exist twice in the hardware catalog.

Repeat steps 1 and 2 as often as necessary.

Note

There is no consistency check. Incorrect configurations are not detected by STEP 7!

The addresses of the "packed" modules are not displayed but are calculated. You take the byte address (X) from the "normal" module above the "packed" module, the bit addresses of the module (X.Y) can be obtained by counting off the inputs or outputs starting with the normal module.

Example: You have configured four modules (2DI), the first (normal) module has addresses 3.0 and 3.1 (displayed "3") the second (packed) module then has addresses 3.2 and 3.3, the third (packed) module 3.4 and 3.5, the fourth (packed) module 3.6 and 3.7.

Undoing Address Packing

If you want to set the start address of a module with a "packed" address again as a byte address, proceed as follows:

1. Select the module with the packed address (e.g. 100.2).
2. Click the "Pack Addresses" button.

   The start address of the module will be set to the next free byte address (e.g. 101.0, if this address is still free).

Special considerations for modules with packed addresses

When a module has a packed address, it can no longer be assigned a slot from the point of view of the CPU.

If you use SFC5 or SFC70 on a module with packed addresses and have assigned a diagnostic address, the diagnostic address is returned. If you have not assigned a diagnostic address, RETVAL W#16#8099 (slot not configured) is returned.

SFC49 (LGC_GADR) or SFC71 (LOG_GEO) and the SZL-ID W#16#xy91 (module status information) can be used for a module with packed addresses. The slot information is, however invalid because a slot assignment is no longer possible.

From the point of view of the CPU, interrupts can also not be assigned for a module with packed addresses. Therefore, in DPV1 mode an additional diagnostic address is automatically assigned for the module.

With older devices (for example IM 151-1BA00, IM 151-1BA01), the features "Pack addresses" and "Plug/remove interrupt" are mutually exclusive.

Rules for Configuring an ET 200S

- Slot 1: Only power module (PM-E or PM-D)
- To the left of an electronic module (EM): only an EM or a power module (PM-E or PM-D)
- To the left of a motor starter (MS): only an MS or PM-D or a power module (PM-D Fx (1..x..4) or PM-X)
- To the left of a PM-X: only a motor starter or a PM-D
- You must pay attention to the assignment of the PM-E voltage range and the EM voltage range.
- A maximum of 63 modules and one interface module are permitted.

Special Features of Assigning Parameters to Reference Junctions

Note the following sequence:

1. In the configuration table (Details view) of the ET 200S: place an analog electronic module and set a channel for the reference junction function to the measuring range "RTD-4L Pt 100 Cl."
2. Double-click the ET 200S (properties of the DP slave interface connection):
   specify reference junction(s), slot and channel of the RTD module
3. Place an analog electronic module for measuring temperature by means of thermocouple (TC module) and assign the reference junction number (of the RTD module) as parameter.

3.8.3 ET 200S with Option Handling

Requirements for Handling Options

To be able to handle options, you require power modules and interface modules that support option handling, for example:

- Power module:
  - PM-E DC 24V HF with diagnostics (as of 6ES7138-4CA60-0AB0)
  - PM-E DC 24V-48V with diagnostics (as of 6ES7138-4CA50-0AB0)
  - PM-E DC 24-48V, AC 24-230V, with diagnostics and fuse (as of 6ES7138-4CB11-0AB0)
- Interface module:
  - IM 151-1 STANDARD (as of 6ES7 151-1AA03-0AB0)
  - IM 151-1 FO STANDARD (as of 6ES7 151-1AB02-0AB0)
  - IM 151-1 HF (as of 6ES7 151-1BA02-0AB0)

Overview of the Procedure

Option handling allows you to prepare the ET 200S for future expansions (options).

The following section provides overview of the function and procedure (for a detailed description, see the *ET 200S Distributed I/O* manual):

1. Install, wire, configure and program the **planned** maximum configuration for ET 200S.
2. Instead of using the electronic submodules that will be required later, use economical RESERVE submodules (138-4AA00 or 138-4AA10) for the first installation. The ET 200S can be completely prewired ("master wiring") because a RESERVE submodule is not connected to the terminals of the terminal submodule and thus is not involved in the process.
3. For those slots that will first be fitted with RESERVE submodules instead of electronic submodules, enable option handling (in the property sheet for the IM 151-1 STANDARD, go to the "Option Handling" tab).
4. In the property sheet for the power submodule, reserve the required address space in the process image output table (PIQ) and the process image input table (PII) for the control and feedback interface by enabling option handling.
5. Later on, you can replace the RESERVE submodules with the configured submodules without having to repeat the configuration procedure.

Rules

Option handling may be enabled for exactly **one** PM E-DC24..48V or PM EDC24..48V/AC24.. power submodule.

Functionality: Option Handling During Start-up

If "Startup when defined and actual configuration differ" is disabled, then the ET 200S will also start up if a RESERVE module is inserted instead of the configured electronic submodule and option handling is enabled for the slot.

Functionality: Option Handling During Operation

- Option handling is enabled for a slot:
  The slot may have the RESERVE submodule (option) or the configured electronic submodule inserted in it. If another submodule is detected in this slot, a diagnostic message is triggered (no submodule, incorrect submodule).
- Option handling is disabled for a slot:
  The slot may only have the configured electronic submodule inserted in it. If another submodule is detected in this slot, a diagnostic message is triggered (no submodule, incorrect submodule).

Substitute Values for the RESERVE submodule

Substitute value for digital inputs: 0

Substitute value for analog inputs: 0x7FFF

Modifying and Evaluating in the User Program

The ET 200S has a control and feedback interface for the "option handling" function.

The modifying interface is in the process image output table (PIQ). Each bit in this address area modifies one of the slots 2 to 63:

- Bit value = 0: The option handling parameters apply. RESERVE submodules are permitted.
- Bit value = 1: The option handling parameters do not apply. RESERVE submodules are not accepted in this slot:

The feedback interface is in the process image input table (PII). Each bit in this address area provides information about what submodule is actually inserted in slots 2 to 63:

- Bit value = 0: The slot contains the RESERVE submodule, an incorrect submodule or no submodule (submodule removed).
- Bit value = 1: The slot contains the configured submodule.

The addresses of these interfaces are reserved as soon as option handling is enabled in the power submodule ("Addresses" tab).

Please note that the option handling function must also be enabled in the DP slave (interface module IM 151-1 STANDARD). If it is not enabled, the address reserved for the control and feedback interface will be released again! Repeated enabling and disabling of option handling may change the address of the control and feedback interface.

For more information on the assignment and meaning the bytes in the process image, see the *ET 200S Distributed I/O* manual.

3.8.4 ET 200S in DPV1 Mode

DPV1 functions provide you with extended functions such as interrupts. You can assign parameters to these functions with the respective DP interface module IM 151.

To be able to set the DPV1 mode, the DP master interface must also be set to the DPV1 mode.

Procedure

1. Configure a station with a DP master that supports DPV1 (such as a CPU S7-41x DP with firmware version 3.0) and the respective ET 200S (IM 151).
   The DP interface for the DP master is set to DPV1 mode.
2. Double-click the DP slave icon (IM 151).
3. Click the "Operating Parameters" tab.
   This register contains additional parameters such as DP Interrupt Mode and DPV1 Interrupts.
4. Set the parameters.

Special Features:

There are certain dependencies among the parameters, which are shown in the following table:

Parameter	DPV0 Mode	DPV1 Mode
Operation with expected does not match actual configuration	No operational restrictions	No operational restrictions
Diagnostic interrupt	Not operational, not set	No operational restrictions
Hardware interrupt	Not operational, not set	No operational restrictions
Insert/remove interrupt	Not operational, not set	If insert/remove interrupt is activated, 'Start-up with expected does not match actual configuration' is activated automatically.

Interrupts for Modules with Packed Addresses

If the submodule can trigger interrupts and its addresses are packed (that is, it has a bit address not equal to 0), then you must use the ET 200S address dialog to assign a dialog address.

The diagnostic address is essential to assigning a DPV1 interrupt to the submodule as the interrupt trigger. Only when a submodule has this "unpacked address" can the CPU assign an interrupt and store information about the interrupt in the start information for the interrupt OB or in the diagnostic buffer. The CPU cannot use a "packed" address to do this.

This diagnostic address assigned to the submodule is only valid for interrupt processing (interrupt OB). For processing input and output data in the user program, the submodule still has the packed addresses.

Note

With older devices (for example the IM 151-1BA00, IM 151-1BA01), the features "pack addresses" and "remove/insert interrupt" are mutually exclusive.

3.8.5 ET 200iS

You can either configure ET 200iS and its electronic modules in the HW Config or conveniently do so with the SIMATIC PDM option package. The respective system requirements and procedure are described below:

Configuration

System Requirement

STEP 7 as of version 5.1, SP 2, Hotfix 1 or PCS7 as of version 5.2.

In this case, ET 200iS is in the STEP 7 hardware catalog. Diagnostic interrupts, process interrupts, inserting and removing interrupts and time stamps are supported.

Configuring the ET 200iS

1. Start the SIMATIC Manager.
2. Configure ET 200iS with HW Config.
   - Create new project.
   - Drag and drop the module from the hardware catalog to the configuration table.
3. Configure the time stamping (optional).
4. Save or download the configuration to the DP master.

Assigning Parameters

System Requirements

STEP 7 as of version 5.1, SP 2, Hotfix 1 and SIMATIC PDM optional package as of version 5.1, SP 2 or PCS7 as of version 5.2.

To be able to work with PDM online, you require a PROFIBUS DP interface e.g. CP5611 (6GK1 561-1AA00). The CP has to be set at the PROFIBUS DP interface (in the SIMATIC Manager via: **Extras > Setting the PG/PC Interface**).

Assigning Parameters to the Electronic Module

1. In HW Config, double-click the first electronic module in the configuration table.
2. SIMATIC PDM is started with the current parameters and identification data of the module.
3. Set the parameters of the electronic module using the SIMATIC PDM, save the parameters (via **File > Save** menu command) and load them via the **Device > Load Into Device** menu command into the electronic module; exit SIMATIC PDM.
4. Double-click the next electronic module in the configuration table and repeat steps 2 and 3 until you have configured all electronic modules.

Assigning Parameters to the Interface Module

1. In HW Config, double-click on DP slave "IM 151-2" (in the upper part of the station window). SIMATIC PDM is started.
2. Set the parameters of the IM 151-2 interface module, save the parameters (via **File > Save** menu command) and download the parameters into the interface module via the **Device > Load Into Device** menu command; exit SIMATIC PDM.

Assigning Parameters to all ET 200iS modules

1. In HW Config, double-click on DP slave "IM 151-2" (in the upper part of the station window). SIMATIC PDM is started and all ET 200iS modules are loaded.
2. Download all module parameters (via the **File > Load into Programming Device/PC** menu command).
3. Assign parameters to all necessary modules. You can navigate to all ET 200iS modules from the left panel of SIMATIC PDM.
4. Save the changes (via **File > Save** menu command) for the file to be updated.
5. Load all parameters into the modules (via **Device > Load into Device** menu command); exit SIMATIC PDM.

You can also parameterize ET 200iS only with SIMATIC PDM (For more information, see the SIMATIC PDM online Help.)

3.8.6 PROFIBUS PA

In order to configure field devices for the PROFIBUS PA (PROFIBUS for Process Automation), you should note the following:

DP/PA Coupler

You **cannot** configure the DP/PA coupler in Hardware Configuration, because it is "invisible" in the station configuration. You only have to set the transmission rate for the PROFIBUS subnet to 45.45 Kbps in the properties dialog box of the PROFIBUS interface of the DP master or DP slave. The coupler reduces the transmission rate to 31.25 Kbps for the PA field devices.

DP/PA Link

The DP/PA link is a gateway between PROFIBUS DP and PROFIBUS PA. The DP/PA link is a DP slave, which for its part (acting as a kind of "master") "opens" a PROFIBUS PA for connecting PROFIBUS PA devices.

The device can be assigned to a DP master system as a DP slave from the "Hardware Catalog" window.

The representation of the DP/PA link also incorporates a symbol for the PA master system as well as the device symbol itself - this is similar to the DP master system. You can assign PA field devices (PA slaves) to this symbol.

The PROFIBUS PA must be running at a transmission rate of 45.45 Kbps when you connect PA devices.

Procedure for Configuring the DP/PA Link

1. Install the optional software SIMATIC PDM (PDM=Process Device Manager); this will enable you to configure the PA slaves from the "Hardware Catalog" window at a later stage.
2. Configure a DP master system.
3. Drag the DP/PA link (IM 157) from the "Hardware Catalog" window to the DP master system.
4. Select the DP/PA link; the DP slave structure will be displayed in the lower half of the station window.
5. Slot 2 represents the master for the PA devices (PA master); double-click slot 2 to configure the PROFIBUS PA.
6. Click the "Properties" button (under "Interface") and then select the subnet with the transmission rate of 45.45 Kbps.
7. Then configure the PA devices.
   You will find the PA devices in the "Hardware Catalog" window under "PROFIBUS-PA". This entry can only be seen if the SIMATIC PDM optional software package is installed.

3.8.7 HART Modules

HART modules are analog modules to which HART transducers can be connected (HART=Highway Addressable Remote Transducer).

HART modules are required for distributed use with the IM 153-2 (ET 200M).

To **assign parameters to the HART transducers**, start the parameter assignment tool SIMATIC PDM

Requirement:

SIMATIC PDM is installed on the programming device/PC.

Representing HART Measuring Transducers

The transducers for HART modules are represented as interface submodules in the configuration table.
Example: The module is located in slot 4. The transducer for the first channel is then represented as slot 4.1.

To start SIMATIC PDM:

- Double-click one of the "slots" for HART transducers.

As you can also use the SIMATIC PDM tool for assigning parameters to PROFIBUS-PA field devices, you can start it as follows:

- Assign a PA field device to a PA master system by dragging & dropping it from the "Hardware Catalog" window, and then double-click this PA field device.

3.8.8 Configuring Software Redundancy

The configuration of a "warm standby" system consists of:

- Two S7 stations with one PROFIBUS-DP master interface each
  (each of these interfaces forms its **own subnet**)
- One or more slaves with the property "can be used redundantly with active backplane bus" (e.g. ET 200Ms with IM 153-3) which are connected to **both** subnets.

This configuration guarantees that the standby station is able to take over the processing of the user program if one station (this is, one of the two DP masters) fails. The slaves that are connected to both stations are then controlled by the reserve station.

Concept Behind Configuring Software Redundancy

With SW redundancy, each of individual stations does not provide any indication that it is being operated together with another station. The coordination of the redundant stations is up to the user and is not, in contrast to H stations, not supported by the system

The ET 200M, which physically is only present once, is configured as two identical S7 stations (same modules, identical addresses, identical settings). To perform this configuration, go to HW Config and select the **Edit > Insert Redundant** menu command.

Procedure

1. Configure the first station completely with all ET 200Ms (IM 153-3).
2. Configure the second station without the ET 200Ms.
3. Select the slaves in the first station, select the master system in the second station, and insert these slaves in the DP master system of the second station (menu command: **Edit > Insert Redundant**).

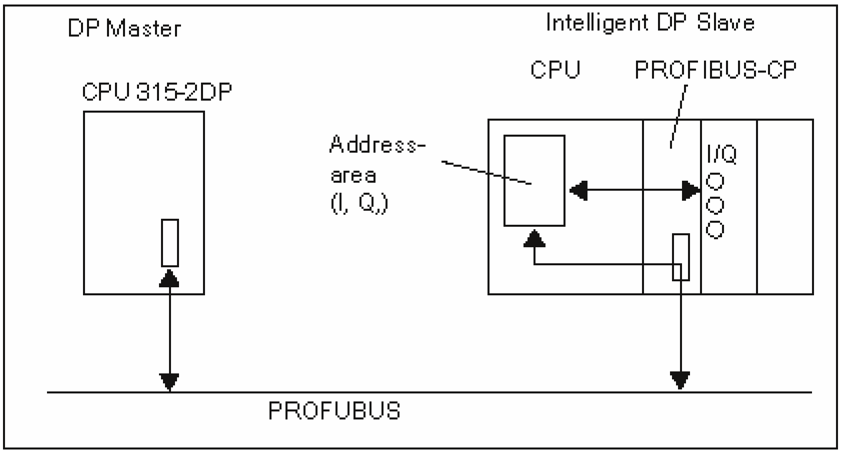
Note

The DP slaves must be configured in each of the two stations, which means they appear as two separate objects - although they are physically one and the same DP slave. If you change the settings for one of the ET 200M DP slaves, you must also **copy** this modified DP slave to the other station **again** to ensure consistency.

3.9 Intelligent DP Slaves

3.9.1 Configuring Intelligent DP Slaves

An intelligent DP slave does not directly provide the DP master with input and output data from a real input or output, but rather from the CPU which, together with the CP, makes up the DP slave.



Difference: "Normal" DP Slave - Intelligent DP Slave

In a "normal" DP slave such as a compact (ET 200B) or modular (ET 200M) DP slave, the DP master accesses the distributed inputs/outputs.

In an intelligent DP slave, the DP master does not access the connected inputs/outputs of the intelligent DP slave, but accesses a transitional area in the input/output address space of the "preprocessing CPU." The user program for the preprocessing CPU must take care of data exchange between the address area and the inputs/outputs.

Note

The configured input/output areas for data exchange between master and slaves must not be "occupied" by I/O modules.

Applications

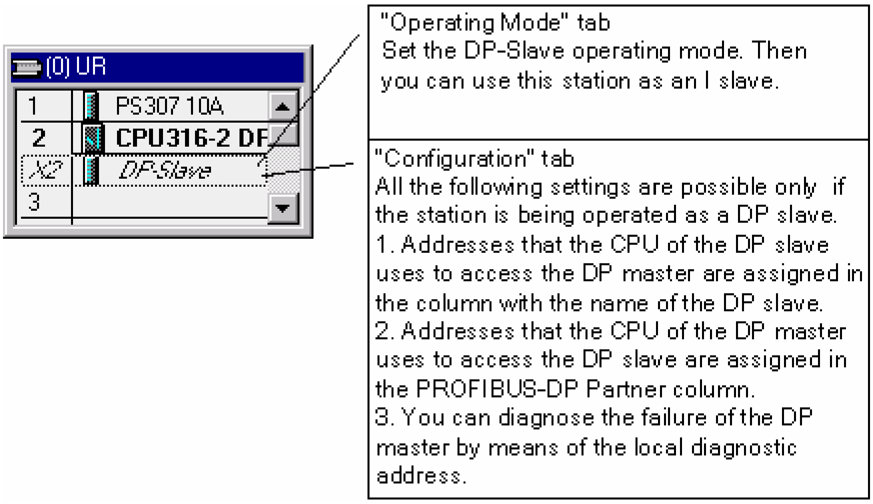
Configurations with intelligent DP slaves:

- Data Exchange Intelligent Slave <> Master
- Direct Data Exchange Slave > Intelligent Slave

Basic Procedure

Two steps are required to insert an intelligent DP slave into a DP master system:

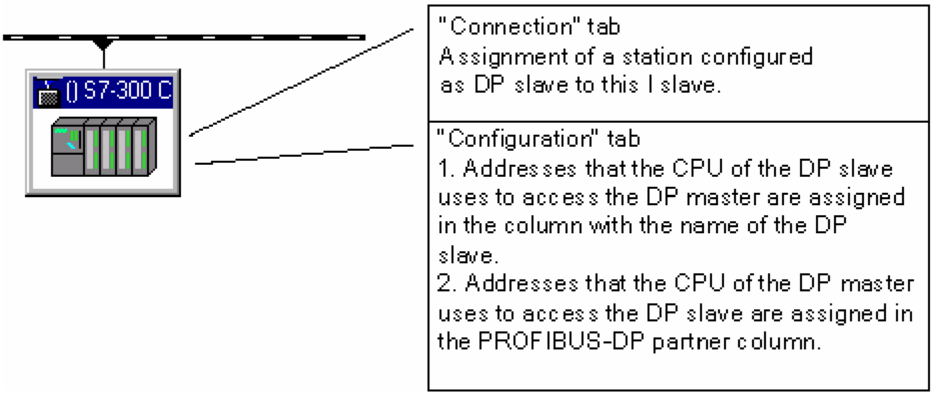
1. Configure a station in which a module with a PROFIBUS-DP interface is being operated as a DP slave (such as a CPU 316-2 DP).



"Operating Mode" tab
Set the DP-Slave operating mode. Then you can use this station as an I slave.

"Configuration" tab
All the following settings are possible only if the station is being operated as a DP slave.
1. Addresses that the CPU of the DP slave uses to access the DP master are assigned in the column with the name of the DP slave.
2. Addresses that the CPU of the DP master uses to access the DP slave are assigned in the PROFIBUS-DP Partner column.
3. You can diagnose the failure of the DP master by means of the local diagnostic address.

2. Configure an additional station in which this DP slave (I slave) is assigned (that is, is connected) to a DP master.



"Connection" tab
Assignment of a station configured as DP slave to this I slave.

"Configuration" tab
1. Addresses that the CPU of the DP slave uses to access the DP master are assigned in the column with the name of the DP slave.
2. Addresses that the CPU of the DP master uses to access the DP slave are assigned in the PROFIBUS-DP partner column.

Configuring the CP 342-5 as a DP Slave

The CP 342-5 can be configured for operation as a DP slave. The station in which the CP was configured is then an "intelligent slave."

Procedure

1. Configure a station with the CP 342-5 DP as DP slave.
   (Select the option "DP slave" in the "Operating Mode" tab of the CP.)
2. Configure a DP master (CPU with integrated PROFIBUS-DP interface or CP with PROFIBUS-DP interface) in another station.
3. Drag the CP 342-5 from the "Hardware Catalog" window (**Configured Stations** folder) and drop it onto the symbol for the DP master system (——▭▭▭▭).
   A dialog box appears in which you can select configured intelligent DP slaves.
4. Confirm your selection with "OK."
5. Configure the DP IDs and addresses for the input and output areas in the displayed configuration table for the DP slave. To do this, drag and drop the "Universal Module" from the "Hardware Catalog" window (**Configured Stations** folder) into the configuration table (lower half of the station window) and then double-click the corresponding row.

Note

Data exchange between a "preprocessing CPU" and a CP 342-5 DP within the DP slave is described in the SIMATIC NCM documentation.

Configuring the CPU 31X-2 DP or CPU 41X-..DP as a DP Slave

CPUs with integrated DP interface (such as CPU 315-2 DP) can be configured for operation as a DP slave. The station is which the CPU was configured is then the "intelligent slave."

The procedure for a CPU 315-2 DP is described below. The procedure is always the same, save for the selection of the CPU type (also refer to Example for the Configuration of an S7-400 as I-Slave).

Procedure

1. Configure a station, for example, with the CPU 315-2 DP as DP slave.
   (Double-click the row 2.1 (interface) in the configuration table and select the DP Slave option in the "Operating Mode" tab.)

   You can set the local I/O addresses and the diagnostic address in the "Configuration" tab.
2. Configure a DP master (CPU with integrated PROFIBUS-DP interface or CP with PROFIBUS-DP interface) in another station.
3. Drag the CPU, for example the 315-2 DP, from the "Hardware Catalog" window (**Configured Stations** folder) and drop it onto the symbol for the DP master system (——▭▭▭▭).
4. Double-click the symbol for the intelligent DP slave and select the "Connection" tab. In this tab you determine which station should represent the intelligent DP slave here.
5. Select the intelligent DP slave and click the "Connect" button.

6. Select the "Configuration" tab and assign the addresses to each other:
   - For data exchange with the DP master by means of the I/O areas, select the "MS" (master-slave) mode.
   - For direct data exchange with a DP slave or DP master, select the "DX" (direct data exchange) mode.
7. Confirm your entries with "OK."

Configuring the ET 200X (BM 147/CPU) as a DP Slave

The BM 147/CPU basic submodule is configured like an intelligent DP slave. In contrast to other intelligent DP slaves, the basic submodule can be found in the "Hardware Catalog" window under PROFIBUS-DP/ET 200X/BM147/CPU.

Procedure

1. Configure the DP slave ET 200X (with BM 147/CPU) as an S7-300 station.
   - Create a new station of the type **S7-300** (menu command **Station > New**).
   - Select the directory PROFIBUS-DP/ET 200X/BM147/CPU in the "Hardware Catalog" window.
   - Drag & drop the object "BM 147/CPU" to the empty station window.
   - Configure the DP slave with the required I/O expansion submodules.
   - Save the station (the intelligent DP slave).
2. Configure a DP master (CPU with integrated PROFIBUS-DP interface or CP with PROFIBUS-DP interface) in another station.
3. Drag the DP slave ET 200X (with BM 147/CPU) from the "Hardware Catalog" window (**Configured Stations** folder) and drop it onto the DP master system symbol (━━▭▭▭▭▭).
4. Double-click the symbol for the intelligent DP slave and select the "Connection" tab. In this tab, you can assign the station that is to represent the intelligent DP slave.
5. Select the intelligent DP slave and click the "Connect" button.
6. Select the "Configuration" tab and assign the addresses to each other.
7. Confirm your entries with "OK."

Configuring the ET 200S (IM 151/CPU) as a DP Slave

The IM 1151/CPU is configured like an intelligent DP slave. In contrast to other intelligent DP slaves, this DP slave can be found in the Hardware Catalog window under
PROFIBUS DP/ET 200S/IM 151/CPU.

Procedure

1. Configure the DP slave ET 200S (with IM 151/CPU) as an S7-300 station.
   - Create a new station of the type **S7-300** (menu command **Station > New**).
   - Select the directory PROFIBUS-DP/ET 200S/IM151/CPU in the Hardware Catalog window.
   - Drag & drop the object IM 151/CPU to the empty station window.
   - Configure the DP slave with the required I/O electronic modules.
   - Save the station (the intelligent DP slave).Configure a DP master (CPU with integrated PROFIBUS DP interface or CP with PROFIBUS DP interface) in another station.
2. Drag the DP slave ET 200S (with IM 151/CPU) from the "Hardware Catalog" window (**Configured Stations** folder) and drop it onto the DP master system symbol (——▭▭▭▭▭).
3. Double-click the symbol for the intelligent DP slave and select the Connection tab. In this tab, you can assign the station that is to represent the intelligent DP slave.
4. Select the intelligent DP slave and click the Connect button.
5. Select the Configuration tab and assign the addresses to each other.
6. Confirm your entries with OK.

3.9.2 Example of Configuring an S7-400 as I-Slave

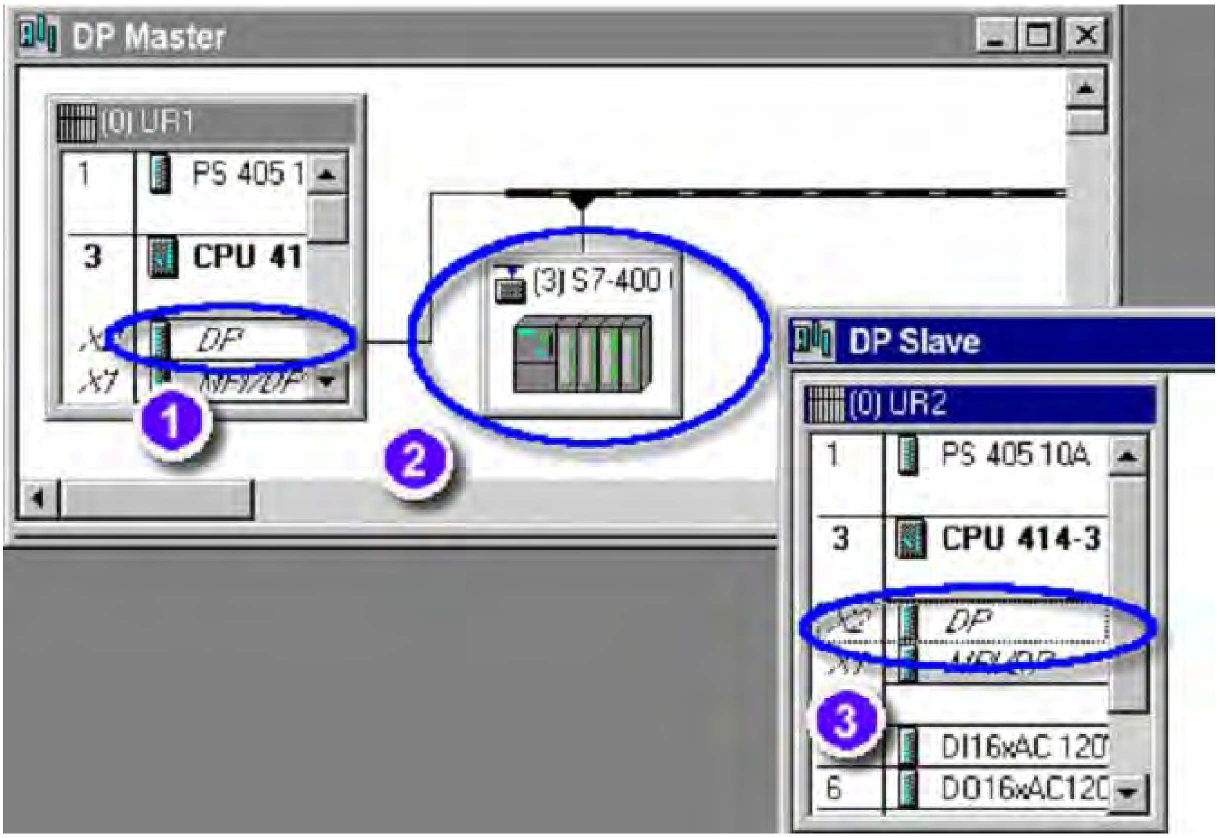
Below we shall create the following configuration:

- Master Station (Name "DP Master") with CPU 417-4 as DP Master
- (DPV1).
  S7-400-CPUs with integrated DP interface can only be customized as DPV1 Master as of Firmware Version 3.0.
- Slave Station (Name "DP Slave") with CPU 414-3 DP as DP I-Slave

As usual for the configuration of intelligent DP Slaves, proceed with the following steps:

Step	Procedure	Explanation
1	Creating the Master Station with the CPU 417-4. Name: DP Master	The PROFIBUS address should be 2.
2	Creating the Slave Station with the CPU 414-3 DP. Name: DP Slave.	The operating mode of the DP interface of the CPU 414-3 DP must be set to "DP Slave". The PROFIBUS address should be 3.
3	Filling out the first row in the "Configuration" tab of the slave station. To create new line, click on the "New" button	If this line is not filled out, the data will be inconsistent. In the properties dialog, select the "MS" (master-slave) mode and apply the default settings.
4	Inserting the DP I-Slaves into the Master Station from the folder "Configured Stations".	From the hardware catalog under PROFIBUS DP select a CPU 41x from the folder "Configured Stations" and drag-and-drop this "Dummy" symbol for the DP Slave station created to the DP Master System.
5	Coupling the inserted DP I-Slaves to the configured Slave Station.	Coupling is prerequisite for assigning the "Dummy" symbol to the configured station, or for assigning the DP Slave inputs to the DP-Master outputs and vice versa.

Arrange the Master Station and Slave Station configuration to view them side-by-side. The result is shown in the screenshot below:



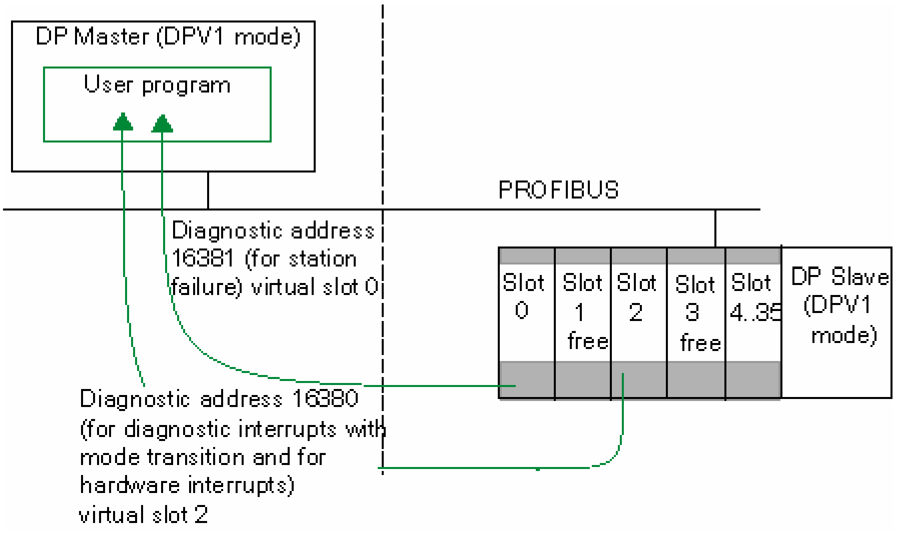
How to Assign Addresses

Step	Procedure	Explanation
1	A double-click on the DP interface of the DP Master in the Master Station opens a dialog and a number of tabs	**"General" tab**: Name of the interface and PROFIBUS address with network assignment. The PROFIBUS address should be set to 2. **"Address" tab**: Logic address of the DP interface (system messages identify the interface via this address). **"Operating mode" tab**: The option "DP Master" must be selected. Also, DP mode "DPV1" must be set. The other tab contents are not relevant for this example.

Step	Procedure	Explanation
2	A double-click on the DP Slave symbol in the Master Station opens a dialog with several tabs	**"General" tab**: Editable designation of the DP Slave symbols and diagnostic address(es). Here, two diagnostic addresses are assigned to a DPV1 Master: The first diagnostic address is assigned to the virtual slot 0 of the DP Slave. Via this address the DP Master receives diagnostic data and/or interrupts that are not assigned to a DP Slave slot. In our example, this is the address 16381. The second diagnostic address is assigned to the virtual slot 2 of the DP Slave. Via this address the DP Master receives diagnostics or interrupts that are assigned to the DP interface. In our example, this is the address 16380. (Also refer to the section "DP Slave from the view of the DP Master"). **"Coupling" tab**: Assigning the DP Slave symbol in the Master station to the configured Slave station. Once the assignment is made, the designation of the configured Slave station appears in the lower section of the tab under "Active Coupling". **"Configuration" tab**: Assignment of I/O areas for the data exchange between the Master station and the Slave station. Double-click the first line, and in the dialog that is displayed, fill out the fields completely.
3	A double-click on the DP Slave interface in the Slave station opens a dialog that contains a number of tabs	**"General" tab**: Name of the interface and PROFIBUS address with network assignment. The PROFIBUS address should be set to 3. **"Address" tab**: Logical address of the DP interface (the system messages reference the interface via this address). **"Operating mode" tab**: The option "DP-Slave" must be selected. Actuate the checkbox "Programming and Status/Control..." if a PG/PC is connected to the same PROFIBUS and if, for example, you want to download programs using this path. A **Diagnostic address** is visible. This address can be used by the intelligent DP slave to diagnose conditions such as a failure of the DP-Master. An ***"Address for the virtual slot 2"*** is only visible if the DP Master supports the "DPV1" mode and if this mode is set. This address can be used in the DP Slave user program to generate a hardware interrupt in the DP Master (also refer to the section "DP Master from the view of the DP Slave "). DP mode: Here, DPV1 should also be selected if "DPV1" mode is set for the DP Master. Otherwise, the ***"Address for the virtual slot 2"*** cannot be selected. **"Configuration" tab**: For the meaning of this tab refer to 2 Difference: The "Local: ..." column can be edited, even if the intelligent DP Slave has not yet been assigned to a DP Master (refer to the "Coupling" tab). For our example all other tab contents are not relevant.

DP Slave from the View of the DP Master

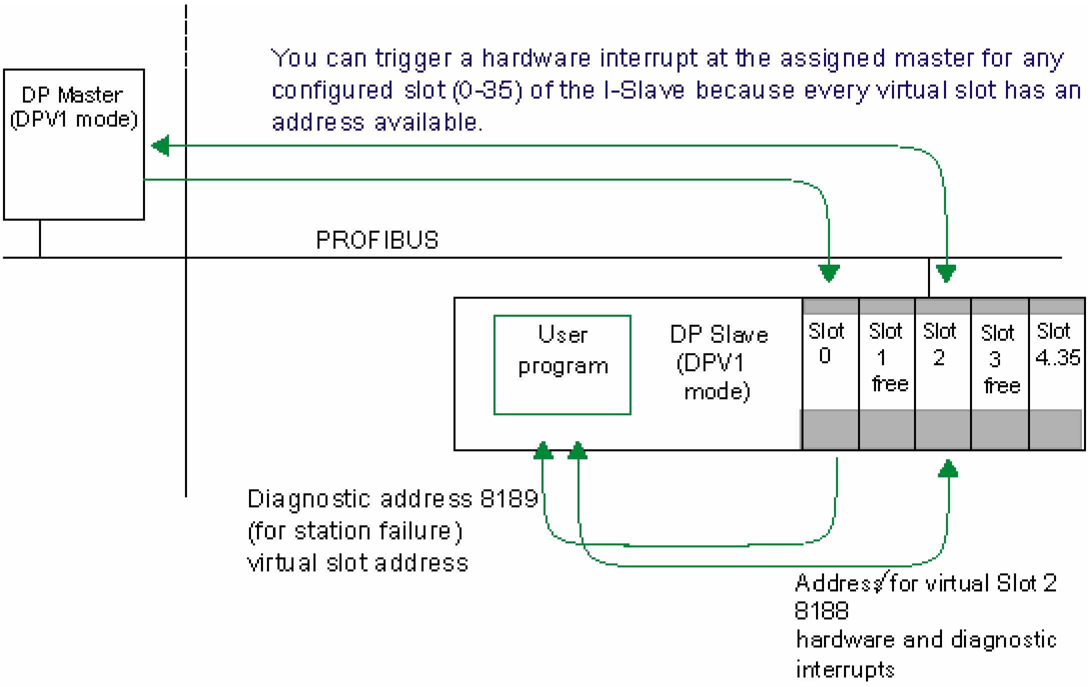
You can use the DP Master CPU user program to obtain DP Slave status information via the configured diagnostic address 16381 that is assigned to virtual slot 0 (Slot 0) of the DP Slave. Slot 0 is, subordinate to DPV1, representative for the entire Slave station. SFB 54 "RALRM", for example, uses the output parameter ID as a pointer to this address if an interrupt is triggered by this Slave and if this interrupt is not assigned to a slot. If the virtual slot 2 of this DP Slave triggers the interrupt (that is, the CPU section) it can be identified via diagnostic address 16380.


DP Master (DPV1 mode)
User program
PROFIBUS
Diagnostic address 16381 (for station failure) virtual slot 0
Slot 0
Slot 1 free
Slot 2
Slot 3 free
Slot 4..35
DP Slave (DPV1 mode)
Diagnostic address 16380 (for diagnostic interrupts with mode transition and for hardware interrupts) virtual slot 2

DP Master from the View of the DP-Slave

You can use the DP-Slave CPU user program, for example, to read DP Master failure information via the configured diagnostic address 8189.

If you have set the DP Slave to DPV1 mode you can, for example, trigger a Hardware Interrupt. Here, in the DP Slave user program you must utilize SFC 7 "DP_PRAL" to trigger the interrupt in the DP Master via the configurable address for the virtual slot 2 (Slot 2). This address is 8188 in our example. The sample below shows user program segments of the DP Slave that has triggered the hardware interrupt and user program segments of the DP Master that evaluate this interrupt.


You can trigger a hardware interrupt at the assigned master for any configured slot (0-35) of the I-Slave because every virtual slot has an address available.
DP Master (DPV1 mode)
PROFIBUS
User program
DP Slave (DPV1 mode)
Slot 0
Slot 1 free
Slot 2
Slot 3 free
Slot 4..35
Diagnostic address 8189 (for station failure) virtual slot address
Address for virtual Slot 2 8188 hardware and diagnostic interrupts

DP-Slave Program (in the I-Slave CPU) for Triggering a Hardware Interrupt

```
//
// ...
      L      DW#16#F0F0               //Constant for the identification of the I-Slave event that
triggered the Hardware Interrupt
      T      MD    100
//...
      CALL   "DP_PRAL"
       REQ      :=M1.0                //trigger Hardware Interrupt if REQ=1
       IOID    :=B#16#54              //Input area (configured address = E 8188)
       LADDR   :=W#16#8189            //Address (configured address = E 8188)
       AL_INFO:=MD100                 //MD 100 is an interrupt info directed towards the DP Master
                                      //(can be read from the start info of OB 40 in the Master as
                                      //OB40_POINT_ADDR)
       RET_VAL:=MW10                  //Return value (0000, if no error)
       BUSY     :=M1.1                //If BUSY =1 the Hardware Interrupt is not yet
                                      //acknowledged by the Master
// ...
```

DP Master Program (in the DP Master station CPU) for the Evaluation of a Hardware Interrupt

```
// ...
      L      #OB40_POINT_ADDR         //load interrupt info
      L      DW#16#F0F0               //compare with the constant which the I-Slave has "passed
                                      //on" when it generated the interrupt
      ==I                             //Interrupt info from I-Slave?

      SPB    m001                     //if yes, jump to respective program segment
      BEB

m001: CALL   FC    100                //Program segment with interrupt processing for I-Slave
// ...
```

3.9.3 Creating an Alarm in the I Slave with SFB75 'SALRM

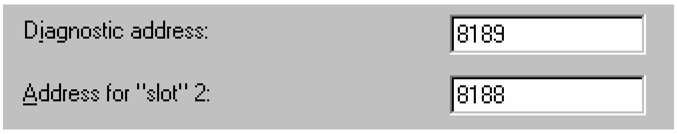
Intelligent DP slaves with SFB 75 'SALRM' can trigger interrupts in their assigned DP master. The following table shows the possible types of interrupts depending on the DP mode setting:

Interrupt Type	DP Mode: S7-compatible?	DP Mode: DPV1
Diagnostic interrupt (OB 82)	Yes	Yes
Hardware interrupt (OB 40 to 47)	Yes	Yes
Insert/remove module interrupt (OB 83)	Yes (if the I slave supports this interrupt)	Yes
Status interrupt (OB 55)	No	Yes
Update interrupt (OB 56)	No	Yes
Vendor-specific interrupt (OB 57)	No	Yes

Addresses for triggering interrupts

You can use each address configured in the "Configuration" tab in the I slave configuration dialog to trigger interrupts using SFB 75. These addresses are not assigned to actual, "real" modules but to virtual "slots".

The addresses for "Slot" 0" (diagnostic address) and "Slot" 2 (Address for "Slot" 2) cannot be used to trigger interrupts.



Diagnostic address: 8189

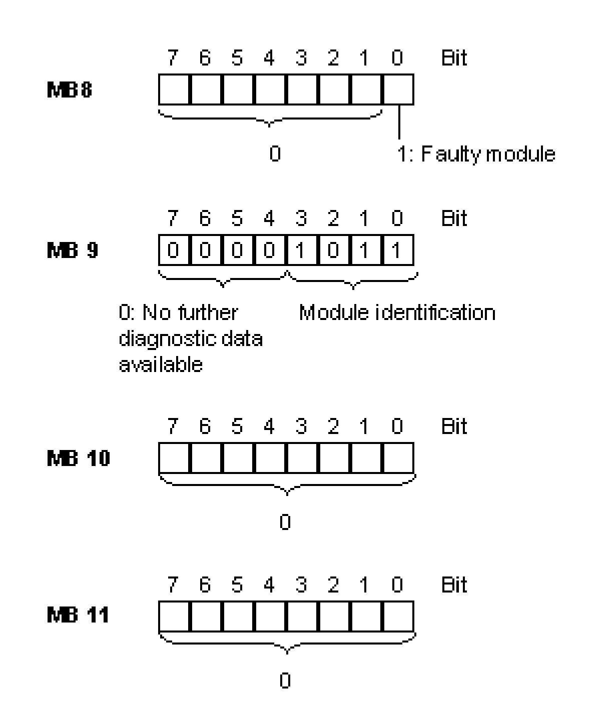
Address for "slot" 2: 8188

Basic Method of Creating Interrupts

The basic procedure for creating an interrupt will be shown as it applies to a diagnostic interrupt.

- In the I slave, the output address 0 in the "Configuration" tab is assigned to a virtual slot.
- In the example, the output address 0 is used to trigger a diagnostic interrupt (OB 82) in the DP master.

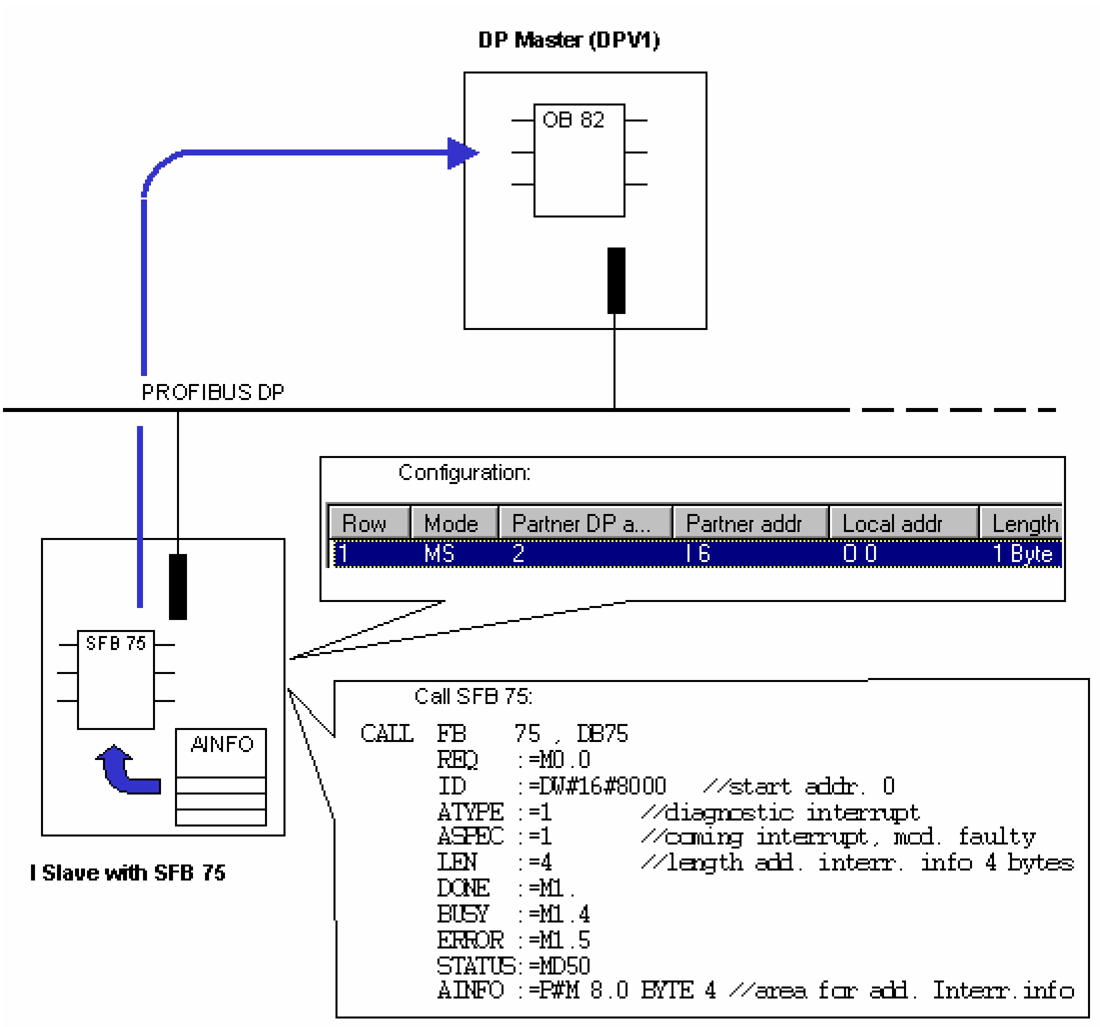
For each diagnostic interrupt, data must also be input with the user program (AINFO). These data have to correspond to the basic structure for additional interrupt information.
For example, you can use the following simplified structure (a detailed description is available in the manual "System and Standard Functions for S7-300/400" in the chapter "Diagnostic Data"):



Note

The additional interrupt information has an influence on the module status data and the "SF" error LED at the I slave. Similarly, the interrupt also influences the module status data and the error LED at the DP master. For this reason, you must be sure to observe the meaning of the diagnostic data sets (data set 0 and data set 1) when providing data for additional information about the interrupt.

The procedure for creating the interrupt is shown in the following illustration:



Configuration:

Row	Mode	Partner DP a...	Partner addr	Local addr	Length
1	MS	2	I 6	O 0	1 Byte

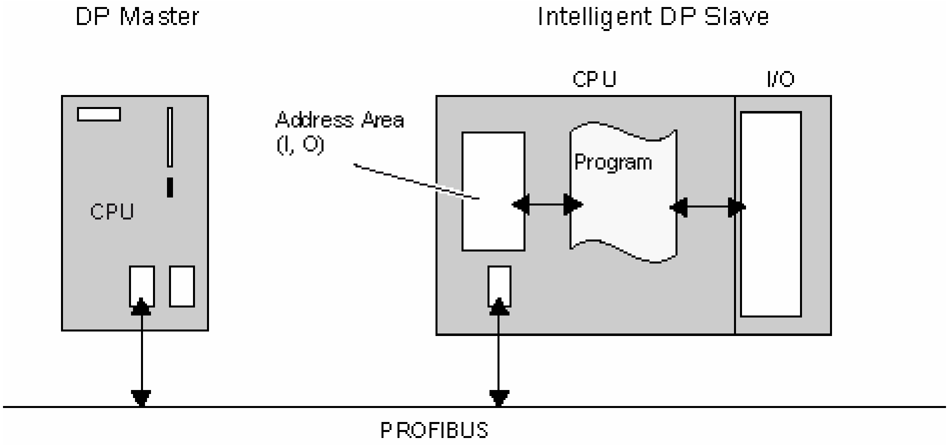
Call SFB 75:

```
CALL  FB   75 , DB75
      REQ   :=M0.0
      ID    :=DW#16#8000   //start addr. 0
      ATYPE :=1       //diagnostic interrupt
      ASPEC :=1       //coming interrupt, mod. faulty
      LEN   :=4       //length add. interr. info 4 bytes
      DONE  :=M1.
      BUSY  :=M1.4
      ERROR :=M1.5
      STATUS:=MD50
      AINFO :=P#M 8.0 BYTE 4 //area for add. Interr.info
```

3.9.4 Direct Data Access to a Module in the Lower-level I Slave

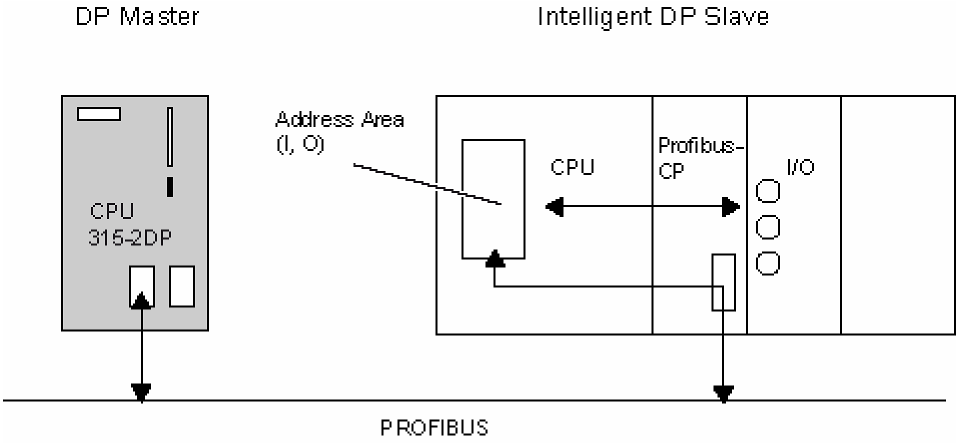
Introduction

The following normally applies to an intelligent slave (I slave, SIMOTION CPU): The addresses for the transfer area of the data and the address for the module in the I slave differ. The start address used by a module can therefore no longer be used in the transfer memory. Therefore, when the higher-level master is to access data of a module in the I slave, you need to program this data communication between module and transfer area in the user program of the I slave CPU.



As of STEP 7 V5.4, Service Pack 2, you have the option of addressing modules in the I slave almost directly from the view of the higher-level master. The start address of the module coincides with the corresponding address in the transfer memory.

Due to the simple address assignment in the transfer memory for a module in the I slave, incorrect addressing caused by due to incorrect user program is avoided for the most part.

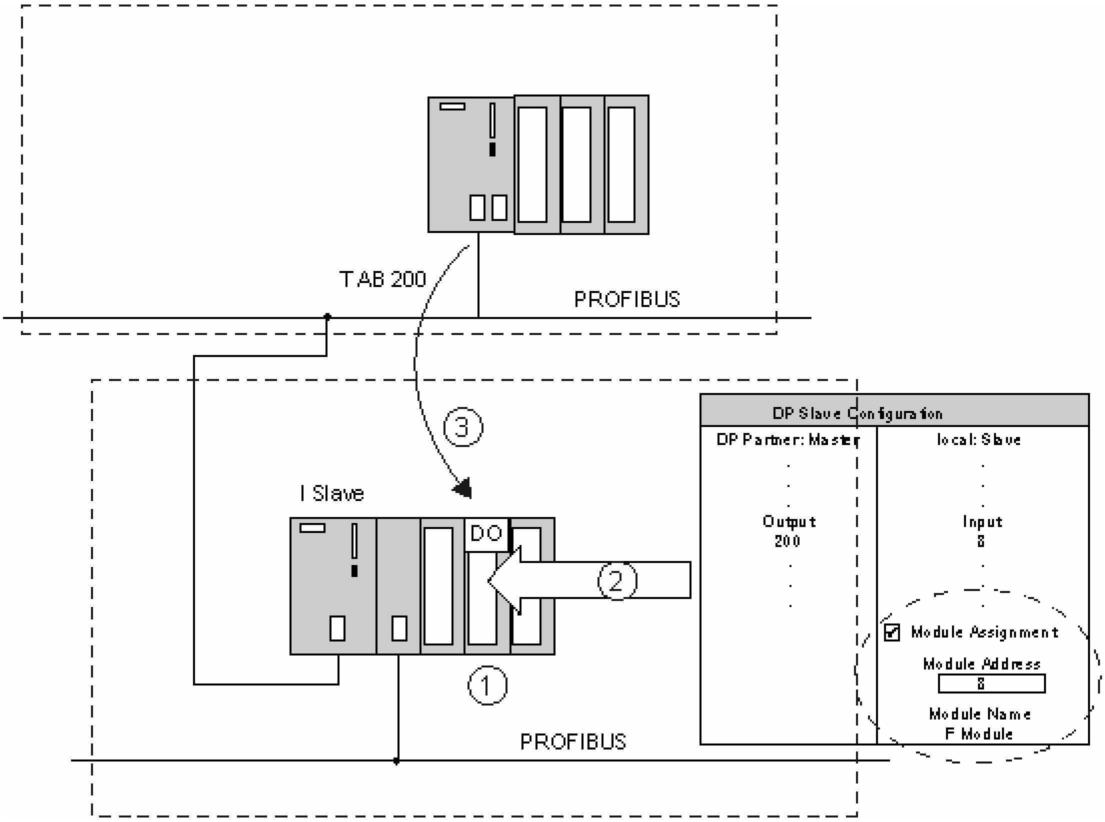


Requirement

The CPU/controller of the I slave station must support this function. There may be restrictions in regard to the modules used.

Concept

The following figure shows the basic concept. The output module has the address 8 in the example. The CPU of the higher-level master station addresses this output module with the output address 200.



1. Configured module, which can be "directly" access by the CPU in the master station.
2. "Configuration" tab of the properties dialog of the I slave station. When the "Module assignment" option is activated, the address of the configured modules must be entered. The name of the module with this address appears automatically in the box below this.
3. The user program of the CPU in the master station access the modules in the I slave almost directly.

Rules

- When you have activated the "Module assignment" check box, a module start address must be available beforehand and module must be inserted in the corresponding slot.
- If the module involved is an input module, the configured address type must be "Output" on the I slave and "Input" on the master.
- If the module involved is an output module, the configured address type must be "Input" on the I slave and "Output" on the master.

Procedure

The procedure is basically the same as that for configuring a "normal" I slave. The difference is in the settings in the "Properties - DP Slave - Configuration" dialog:

1. Create the I slave station and configure all the required modules.
2. Create the master with a DP master system.
3. Link the I slave station to the master station; in other words, drag the I slave from the "configured stations" hardware catalog to the higher-level master system.
4. Open the "Configuration" tab and assign the addresses to one another. You must create a line for each module which you wish to "directly" access from the master station:
   - Select the "MS" (Master-Slave) mode for the data communication with the DP master.
   - Select the "Module assignment" option and enter the start address of the module. If there is a module for the specified address, the name of the module appears automatically in the box below this.
   - If the master station has already been configured, you can also enter the addresses on the DP master.
5. Start a consistency check (**Station > Check Consistency**) and correct any errors that occur.

3.10 Direct Data Exchange (Lateral Communication)

3.10.1 Configuring Direct Data Exchange between PROFIBUS-DP Nodes

Introduction

In a configuration for direct data exchange (lateral communication), local input address areas of an intelligent DP slave or of a DP master are assigned to the input address areas of a PROFIBUS-DP partner.

The intelligent DP slave or the DP master uses these assigned input address areas to receive the input data that the PROFIBUS-DP partner sends to its DP master.

Number of PROFIBUS-DP Partner That Can Be Connected

The type of interface limits the total number of PROFIBUS-DP partners that are connected directly to a DP interface or are addressed over this interface by means of direct data exchange. A maximum of 32 PROFIBUS-DP partners can be addressed on the MPI/DP interface.

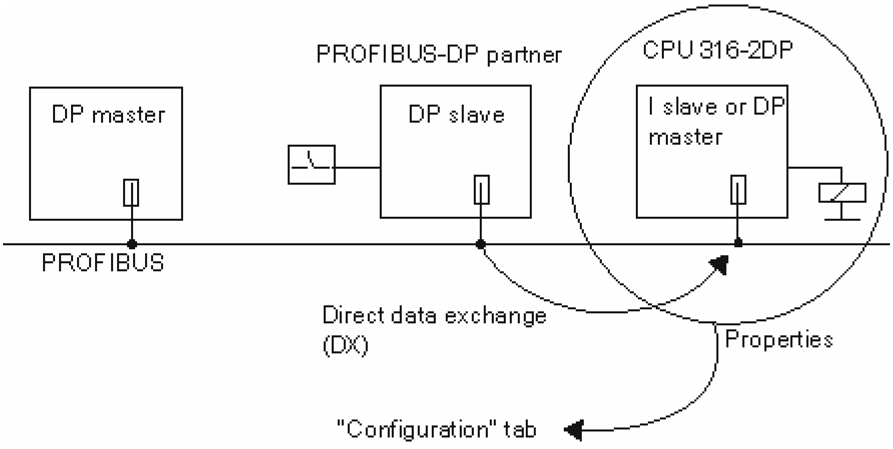
Applications

The following applications are possible for direct data exchange:

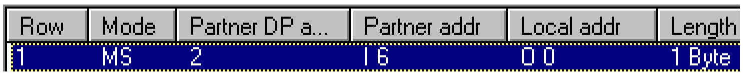
- Configuration with intelligent DP slaves (direct data exchange: slave -> intelligent slave)
- Configuration with two DP Master systems (data exchange: slave -> master)
- Configuration with two DP master systems (direct data exchange: slave -> intelligent slave)

Starting Configuration

1. Double-click the DP interface of the configured receiver (DP master or already configured I slave).
2. Select the "Configuration" tab.
3. To create new line in order to configure direct data exchange, click the "New" button.
4. In the dialog that appears, select "DX" mode and assign the input address areas to each other (you can find more information in the online help for this dialog).



Contents of the "Configuration" tab



Row	Mode	Partner DP a...	Partner addr	Local addr	Length
1	MS	2	I 6	O 0	1 Byte

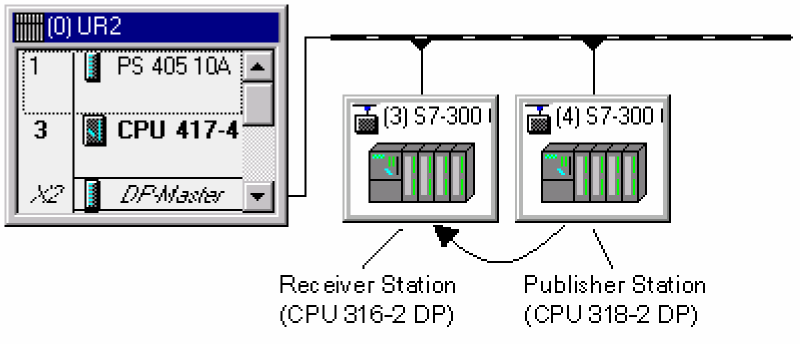
- **Mode**: "DX" for direct data exchange, ("MS" for master-slave)
- **Partner DP Addr**.: PROFIBUS address of the DP partner
- **Partner Addr**.: Address of the logically assigned address area (sender)
- **Local Addr**.: Address of the logically assigned address area (receiver)

For further information, see the online help for this tab.

3.10.2 Example of Configuring Direct Data Exchange

The configuration for the example is as follows:

- CPU 417-4 as DP master
- CPU 316-2 DP as receiver
- CPU 318-2 DP as publisher



Task

CPU 318-2 DP sends 8 words of consistent data to the DP master.

CPU 316-2 DP receives the first two bytes of these data.

Procedure

1. Configure three stations, each with the CPUs listed. Then, assign meaningful names to them such as "DP master station", "receiver station", and "publisher station".
2. Configure the publisher and receiver station as intelligent slave:
   - Double-click the DP master row.
   - Select the Operating Mode tab.
   - Select the DP Slave option.
3. In the master station:
   - Drag the icon for CPU 31x 2-DP from the "Catalog" window (PROFIBUS-DP, "Already Configured Stations" folder) to the DP master system.
     The DP Slave Properties dialog box appears with the "Connection" tab.
   - Select a slave in the Configured Slave Controllers box and click the "Connect" button. Then confirm with "OK".
   - Repeat the procedure in order to connect the second station that was configured as an intelligent slave to the master system.

4. Configure the address area of the publisher station so that the DP master reads the data from the CPU 318-2 DP by means of the address I 200:
   - Double-click the DP slave row of the CPU 318-2 DP.
   - Select the "Configuration" tab and click the "New" button. Fill in the fields in the properties dialog as follows:

Publisher Station (local)	Mode Address type Address	= = =	MS (master-slave) Output 100
PROFIBUS-DP Partner	PROFIBUS address = 2 (fixed, PROFIBUS address of the DP master) Address type Address	 = =	 Input 200
Length, Unit, Consistency	Length Unit Consistency	= = =	8 Word Entire length

5. Configure the address areas of the receiver:
   - Double-click the DP slave row of the CPU 316-2-DP.
   - Select the "Configuration" tab and click the "New" button. Fill in the fields in the properties dialog so that the CPU 316-2 uses the address I 120 to access the data of the CPU that the CPU 318-2 sends to its master:

Receiver Station (local)	Mode Address type Address	= = =	DX (direct data exchange) Input (fixed) 120
PROFIBUS- DP Partner	PROFIBUS address Address type Address	= = =	3 (offered as a selection) Input (fixed) 200 (offered as a selection)
Length, unit, consistency	Taken automatically from the settings in the publisher station. Change the length to 1 (word) because only the first two bytes are to be read.		

Special Feature:

In principle, you can set an address that is higher than I 200, such as I 202. STEP 7 adapts the length of the consistent data automatically. You can also set a length that is shorter than the one that the publisher prescribes (such as 1 byte).

Note

If the setting for consistent length at the publisher is 3 bytes or more than 4 bytes and the data are being transmitted with SFC15 (DPWR_DAT), the receiver **must** always use SFC14 (DPRD_DAT), even if only 1 byte is being read.

In this case, if you are using Load operations (L IB…), a "0" is read in (incorrect value).

Calling SFC15 in the Publisher (CPU 318-2 DP)

```
CALL   "DPWR_DAT"
       LADDR   :=W#16#64            //start address Q 100
       RECORD :=P#M 10.0 BYTE 16    //source area for user data
       RET_VAL:=MW100               //return value
```

Calling SFC14 in the Receiver (CPU 316-2 DP)

```
CALL   "DPRD_DAT"
       LADDR   :=W#16#78            //start address I 120
       RET_VAL:=MW100               //return value
       RECORD :=P#M 10.0 BYTE 2     //target area for user data
```

3.10.3 Configuring a DP-Slave (GSD Rev. 5) as a Receiver for Direct Data Exchange

As of STEP 7 Version 5.3, you can install DP slaves that can be configured as receivers for direct data exchange by means of a GSD file ("standard slaves"

Configuration of direct data exchange (lateral communication) is possible for GSD files as of Revision 5.

Key Words in the GSD File

A DP slave with the GSD entry "Subscriber_supp =1" can be configured as a receiver (subscriber). In STEP 7 this entry means that the property sheet for the DP slave contains an "Address Configuration" tab in which you can assign input and output areas.

A DP slave with the entry "Publisher_supp =1" can be used as a transmitter (publisher) for direct data exchange. The input areas for such a DP slaves can be selected ("subscribed to") in the "Address Configuration" tab. DP slaves without this entry, that is without "publishing capability", are not even offered as transmitters for direct data exchange.

Other entries in the GSD file are automatically applied by STEP 7. For example, in the consistency test the maximum number of relations for direct data exchange are considered. If this number is exceeded, a message will be displayed prompting you to reduce their number.

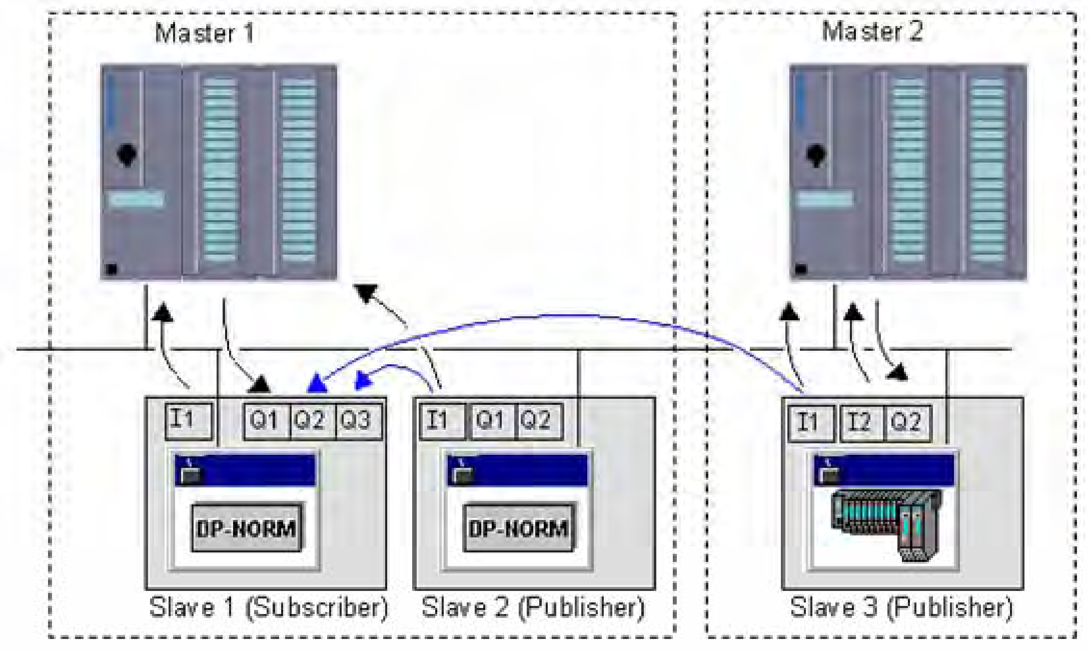
Description

As in the case of direct data exchange, such as between data-transmitting (publishing) DP slaves and data-receiving (subscribing) I slaves, data are exchanged directly between DP slaves through a DP master.

In contrast to a data-subscribing I slave, which "monitors" the input data from a publishing DP slave in its own local address area, a data-subscribing "standard slave" creates an image of the input data of the publishing DP slave directly on the outputs of the subscribing DP slave.

This means that the parameters assigned to the subscriber determine whether an output byte comes from the DP master or from a publishing DP slave.

The following illustration shows an example of this situation.



Here two master systems are shown in a configuration for direct data exchange.

Slave 1 was configured as the subscriber for direct data exchange. Data exchange is shown between masters and slaves.

Slave 1 outputs the input data from slave 2 to its output data area Q3. Both areas must have the same length (a least 1 byte).

Slave 1 also outputs the input data from slave 3 to its output data area Q2. Both areas also have the same length.

For master 1, slave 1 (subscriber) only has the output data area Q1. The output data areas Q2 and Q3 are not available to it.

Rules and Notes

The direct data exchange between subscriber and publisher is restricted to DP slaves (slave-to-slave communication).

Direct data exchange between DP slaves in different master systems is possible. In this case, both master must be connected to the same PROFIBUS subnet.

As a rule, output areas of the subscriber are assigned to the DP master and can be assigned to a publisher during configuration. It is also possible to assign an output area to neither the DP master nor to a publisher. In this case, the output areas are assigned a value of "0". The consistency test will provide warning for such areas.

Direct data exchange is no limited to standard slaves. Any slave in the Hardware Catalog ("PROFIBUS DP" folder) that is listed as a transmitter (publisher) or receiver (subscriber) for direct data exchange (see the info text in the Hardware Catalog) can be used.

Procedure

1. Import the required GSD files for slaves that are to be configured as publishers or subscribers.
2. Configure a master system with these slaves.
3. For a slave to be configured as a subscriber, carry out the following steps. Do this for each ID concerned (module).
   - Double-click the DP-ID.
   - Select the "Address Configuration" tab
   - Assign the respective address areas: either to the master (default setting), to a publisher or to no node.
     If you set an address area for the DP ID to "**DX**" **mode**, then this address area is invisible from the point of view of the DP master CPU. This means that there are no logical addresses for the DP master CPU for this address area. The user dialog shows the area of the publisher (DP partner) that modifies the outputs of the subscriber.
     In the illustration above, this is the Q2 address area from slave 1.
     From the point of view of Master 1, Q2 is invisible. During configuration, the name, PROFIBUS address and the logical address area for I1 from slave 3 (publisher) is shown (e.g. I 100).
     If you set an address area for the DP ID to "**MS**" **mode**, then this address area is visible from the point of view of the DP master CPU. This means that during configuration this address area is mirrored to logical output addresses (e.g. Q 100).
     If you set the DP ID to **Mode** "**--**" for the subscriber, then this address area is invisible from the point of view of the DP master CPU and this area is not modified by any other node. During configuration, this address area is not mirrored to logical output addresses.
     The online help on the "Address Configuration" tab contains information on selecting the address area and an example of how this is done.
   - Confirm you settings with "OK"
4. Configure the remaining slaves at the station and the master with all modules.
5. Save and compile the configuration.
6. Download the hardware configuration to the station.
   If several master systems are affected by direct data exchange, the affected stations must also be downloaded.

3.11 Working with *.GSD Files

Device Database (GSD) File for DP Slaves

All the properties of a DP slave are saved in a device database (*.GSD) file. *STEP 7* requires a *.GSD file for every DP slave in order that the DP slave can be selected in the module catalog. The manufacturer supplies a *.GSD file for non-Siemens devices that are DP slaves.

Device Database (GSD) File for IO Devices

Similar to the case for DP slaves, there are GSD (Generic Station Description) files for IO devices. The DSD file for an IO device contains all the properties of the device. These files have the file extension "*.xml".

3.11.1 Installing a *.GSD File

If a DP slave or an IO device does not appear in the "Hardware Catalog" window, you must install the corresponding *.GSD file supplied by the manufacturer. You can also use GSD files that were used in another project.

Requirements

The required GSD files must be in a folder on the hard disk or in an STEP 7 project that you have access to. GSD files are always saved with the project. In other words, all the information needed to display the device (including the symbols) are available in the project.

Procedure

1. Close all stations in HW Config.
2. Select the menu command Options > Install GSD File.
3. In the "Install *.GSD Files" dialog box that appears, select the source:

   The folder containing the *.GSD files, or

   The STEP 7 project containing the *.GSD files
4. Select one or more files from the list of *.GSD files, and then click the "Install" button.

If some of the files could not be installed or if errors occurred during the installation, STEP will log these events. To open this log file, click the "View Log" button.

Overwriting *.GSD Files

To represent DP slaves STEP 7 uses device database (*.GSD) files and symbols which are installed in STEP 7, meaning:

- They were installed automatically with STEP 7 or
- They were installed at a later date

When you install or import the files at a later date, the existing *.GSD files/symbols are not completely deleted but are stored in the following backup directory:

\\Step7\S7data\Gsd\Bkp[No.],

where [No.] is a serial number which STEP 7 assigns automatically.

Restoring Overwritten *.GSD Files

To restore *.GSD files/symbols you overwrote accidentally, proceed as follows:

1. Select the menu command **Options > Install GSD File.**
2. In the following dialog box, navigate to the directory **\\Step7\S7data\Gsd\Bkp**[**No**]**.** Make sure that you select the required backup directory (use the Explorer to find the directory with the correct date/time).
3. Click the "Open" button.

3.11.2 What You Should Know About GSD Revisions

The properties of DP slaves are provided to configuration tools by means of GSD files.

Any extended functions for distributed I/O devices will have an effect on the GSD specification, such as resulting in the definition of new keywords.

This results in there being different versions of the specification. The version of the specification that a GSD file interacts with is called "GSD-Revision".

The GSD revision is contained as an obligatory keyword "GSD_Revision" in GSD files as of GSD Revision 1. GSD files lacking this keyword will be thus be interpreted by configuration tools as GSD Revision "0".

STEP 7 V5.1, Service Pack 3

This version of STEP 7 can interpret GSD files up to GSD Revision 4. This means that you can also use the following new functions for DP slaves that are installed by means of GSD (Revision 4):

- F-parameter assignment of submodules
- Diagnostic messages for interrupt blocks
- Constant mode (isochronous mode)

Special Features:

Please note that the optional package COM PROFIsafe must be installed to edit F-parameters. If this package is not installed, the parameters will not be visible and thus cannot be changed. The F-parameters will, however, remain (default values are taken from the GSD file or are values changed with COM PROFIsafe) and will be taken into account when generating a configuration (For further information, see the documentation on distributed F-systems).

The following functions, although possible in GSD Revision 4, are not supported:

- A second parameter frame (extended parameter assignment)
- Subscriber functions (receive capability for direct data exchange/lateral communication)
- Keywords for HART parameter assignment

STEP 7 V5.3

This version of STEP 7 can interpret GSD files up to GSD Revision 5. This means that you can use new functions also for DP slaves that are installed using GSD (Revision 5):

- Using a DP Slaver as a receiver (subscriber) for direct data exchange (lateral communication).
- Redundant configuration of a DP slave in an H station.
  A DP slave can be redundantly configured if the GSD entry "Slave_Redundancy_supp = 8" is present.
- Clock Synchronization for DP slaves.
  A DP slave with the GSD entry "Time_Sync_supp = 1" provides the "Clock Synchronization" tab for configuring this function.

STEP 7 V5.3 Service Pack 1

In addition to GSD files for DP slaves (up to Revision 5), this version of STEP 7 can also interpret GSD files for PROFINET IO devices in XML format (Generic Station Description). Both types of GSD files are handled identically.

There are special considerations with respect to naming these new GSD files and specifying their versions. STEP 7 V5.3, Service Pack 1, interprets GSD files with the GSDML schema V1.0.

3.11.3 What You Should Know about GSD Files for PROFINET IO Devices

Basic information

The properties of PROFINET IO devices are not stored in a keyword based text file (as for PROFIBUS DP slaves) but in an XML file whose structure and rules are determined by a GSDML schema.

The language used to described GSD files is GSDML (Generic Station Description Markup Language). It is defined by the GSDML schema.

A GSDML schema contains validity rules that, among other things, allow the syntax of a GSD file to be checked. Manufacturers of IO devices get the GSDML schemas (in the form of schema files) from PROFIBUS International.

Expanded functionalities in the area of PROFINET IO also effect the GSDML specification and the associated schema. Expanded functionalities result in a new version of the GSDML specification and the schema.

Names of GSD files for IO devices

The structure of the names for a GSD file is explained based on the following example:

"`GSDML-V1.0-Siemens-ET200S-20030616.xml`"

Part of name	Explanation
GSDML	Every GSD file for IO devices starts with this character string
V1.0	Version of the GSDML schema
Siemens	Manufacturer
ET200S	Name of device
20030616	Version identifier (date)
.xml	File extension

If the Hardware Catalog happens to contain duplicate names, then the GSD file with latest version number or date is the one used.

Versions of GSD files for IO devices

The version information for GSD files consists of two parts:

One of these parts indicates the version of the GSDML schema. This determines what languages that a GSD file uses.

The other part indicates the file version (based on date). A GSD file receives a new, later version number whenever it is improved (e.g. errors corrected) or its functions are enhanced.

Enhanced functions may lead to a new version of the GSDML schema. STEP 7 supports a new version of the GSDML schema in the relevant current version, however, possibly with some restrictions.

3.12 DPV1

3.12.1 What You Should Know About PROFIBUS DPV1

Below you can find information relating to the topics:

- New mechanisms introduced by the DPV1 Masters/Slaves
- The changes you meet when configuring and programming those components

Additional Information

Our Internet Customer Support Pages contain a FAQ publication relating to this topic under article ID 7027576. (Title: "Changing over to DPV1"; refer to **Automation Systems > SIMATIC Decentralized Peripherals > PROFIBUS > General**)

How do you recognize a DPV1 Master/Slave?

CPUs of the S7-400 family with integrated DP interface support DPV1 Master functions as of Firmware Version 3.0.

DPV1 Master functions are also available for the new CP 443-5 (DX03).

DP Slaves listed in the Step 7 hardware catalog under their family name can be recognized in the Info text as DPV1 Slaves.

DP Slaves implemented in STEP 7 via **GSD Files** support DPV1 functions as of GSD Revision 3.

Additional Functions DPV1 Devices (Master/Slaves)

DP Masters and DP Slaves that support DPV1 are, contrary to the "old" devices (often referred to as "Standard Masters" or "Standard Slaves"), equipped with the following additional functions:

- Acyclic data exchange between Masters and Slaves is supported (reading/writing a data record to reconfigure an operational slave). The module data records and their structure are explained in the documentation for the respective module.
- A DPV1 Slave can provide interrupts to insure that the Master CPU handles the event triggering the interrupt. Interrupt data is also evaluated in CPU STOP mode (updating the diagnostic buffer and module status); OBs are, however, not processed in STOP mode.
  In addition to the interrupts known in SIMATIC (for example, diagnostic interrupt with ET 200M), the new Status/Update/Manufacturer specific interrupts are now also supported.

Comment: DP Slaves that were up to now fully integrated in STEP 7 (that is, Slaves not configured via GSD File, but rather via internal STEP 7 module information) also supported a part of those functions, however, with an S7 specific significance of, for example, data record contents. New is here that those functions are now available manufacturer independent (for example, for DP Slaves with GSD Files Revision 3).

3.12.2 Configuring DPV1 Devices

Changing the DP Master Interface and Configuring DP-Slaves

DPV1 operating mode is set by default in your hardware configuration with STEP 7 when you insert a DP Master with DPV1 functions in the master module rack. To change this operating mode, proceed as follows:

1. Double-click on the "DP Master" row of the CPU in the configuration table. This row represents the DP interface..
2. In the properties dialog, click on the dropdown menu "DP-Mode" and select the desired operating mode:
   - "S7 compatible", if you do not use DPV1 functions
   - "DPV1", if you use DPV1 functions.
3. Implement all DP Slaves in the DP Master system.
   Valid is:
   - On a DP interface with DPV1 operating mode you can, generally seen, also operate DP Slaves that do not support those functions (for example, DP Slaves with GSD Revision < 3).
   - You can also, on principal, operate DPV1 Slaves on a DP interface in operating mode "S7 compatible". In this case, DPV1 functions are automatically switched off. However, manufacturer specific configuration rules for certain DP Slaves possibly force DPV1 mode, thus preventing you from connecting them to the DP Master system (this is automatically checked during the configuration process)!

Consequences of Switching Over a DP Master Interface

Case 1: The DP Master Interface is to be changed to "DPV1":

On this interface you can maintain operation of currently connected DP Slaves that do not support DPV1 functionality.

Case 2: The DP Master Interface is to be switched over from "DPV1" to "S7 compatible":

STEP 7 checks whether all DP Slaves can be switched over to this operating mode. If a DP Slave forces DPV1 functionality, for example, if an interrupt must be activated, you cannot use this DP Slave on the DP Master in "S7 compatible" operating mode.

Changes in the Structure of a DPV1-Slave

DPV1 Slaves are equipped with a slot model design, which is new compared to the ones known up to now. However, the consequences involved for you as STEP 7 user are minute.

As a rule, you address the distributed I/Os via logic addresses in the same way as you used to do. The slot address <-> logic address conversion is carried out automatically or controlled in a dialog when a DP Slave is being configured. Here, the slot and address assignment of the configuration corresponds with the assignment you can determine via address conversion in the user program (conversion of the physical address into a logic address and vice versa - via SFC 5 and SFC 49).

As of STEP 7 V5.1, Service Pack 2, the slots in the detail view of a DP1 Slave always start with slot number 1. This has the consequence that, when using DP Slaves not configured via GSD file, the DP connection (for example, an IM 153) is visible on slot number 2.

Diagnostic Address

The diagnostic address of a DP Slave is not altered by the switchover. With DPV1 Slaves it is automatically assigned to the "virtual" slot number "0" as station representative.

Generally, the following assignment applies:

- Diagnostic data and interrupts which can only be assigned globally to the DP Slave are assigned to virtual slot 0 and its diagnostic address: for example, interrupts from modules inserted in slots which have not been configured, station failure/station return (OB 86)
- The remaining slots and their respective start addresses are configured with diagnostics and interrupts triggered by the module (for example, by a DP connection IM 153-2 in slot 2).

3.12.3 Programming DPV1 Devices

New Interrupt OBs for DPV1 Events

DPV1 Slaves can trigger interrupts. You can use the respective OBs the S7 CPU operating system provides up to now for Diagnostic/System/Extraction/Insertion Interrupts.

New are the OBs for the following interrupts:

DPV1 Interrupt	OB	Explanation
Status Interrupt	OB 55	A Status Interrupt can be triggered by the operating state transition of a module, for example, from RUN to STOP mode. Refer to the respective DPV1 Slave manufacturer's documentation for detailed information on events that can trigger a Status Interrupt.
Update Interrupt	OB 56	An Update Interrupt can be triggered after a slot was reconfigured. For example, this might occur as a result of local or remote access to the parameters. Refer to the respective DPV1 Slave manufacturer's documentation for detailed information on events that can trigger an Update Interrupt.
Manufacturer Specific Interrupt	OB 57	The event triggering a Manufacturer Specific Interrupt can be specified by the manufacturer of a DPV1 Slave

New SFBs and SFCs for Accessing DPV1 Slaves

In order to make the topic more comprehensive, the table below shows - wherever possible - the new interfaces and their functions compared with previous interfaces. You can find detailed information in the descriptions of the SFBs/SFCs and new OBs. A conversion of existing configurations to the new SFBs/SFCs is not necessarily required. However, you should use the new SFCs/SFBs when you create new projects with a DPV1 configuration to be able to use all DPV1 functions.

Function	Previous Interface	New Interface (DPV1)	Comments
Read Data Record	SFC 59 RD_REC	SFB 52 RDREC	-
Write Data Record	SFC 58 WR_REC	SFB 53 WRREC	-
Receive interrupt from a DP Slave	-	SFB 54 RALRM	The SFB must be called in the OB that is triggered by the interrupt.

Note

If a DPV1 slave is configured using a GSD file, (GSD as of Rev. 3) and the DP interface of the DP master is set to "**S7-compatible**", no data records may be read from or written to the I/O modules with SFCs 58/59 or SFBs 53/52 in the user program. In this case, the DP master would address the wrong slot (configured slot + 3).

Workabout: Set the interface of the DP master to "DPV1".

Checklist for Testing Existing User Programs

The following sections of your existing user program must be checked if you have edited the configuration with STEP 7 V5.1, Service Pack 2 and if you have switched to "DPV1":

Function	What is to be checked?
Address conversion	For DP Slaves configured via GSD files you must check the assignments Slot <-> Logic Start Address, if you have used address conversion in the user program (SFC 5, SFC 49, SFC 50). Slot 0 has an additional address. • DP Slave implemented via GSD File: Previously the first I/O module of the DP Slave was assigned to slot 4. However, the first I/O module is now assigned to slot (as you can see in the hardware configuration). • DP Slave integrated in STEP 7 (for example, ET 200M): the interface module (slot 2) has its own address.
Reading Diagnostics with SFC 13	The originally assigned diagnostic address still functions. STEP 7 assigns this address internally to slot 0. However, the diagnostic data record of DPV1 Slaves has a different structure (refer to the description of DP Slaves. With ET 200M, for example, also refer to the keyword "Extended Diagnostics").
Reading/Writing Data Records	If you transfer data records to a DPV1 slave with SFC58 "WR_REC" or if you fetch data records from a DPV1 slave with SFC59 "RD_REC" and if this DPV1 slave operates in DPV1 mode, the DP Master evaluates the error information it received from the Slave as follows: If the error information lies within the range from W#16#8000 to W#16#80FF or W#16#F000 to W#16#FFFF the DP master passes the error information to the SFC. If it lies out of this range, the CPU passes the value W#16#80A2 to the SFC and suspends the slave. For a description of the error information received from DPV1-Slaves, see Receiving an Interrupt from a DP-Slave with SFB 54 "RALRM" STATUS[3]. See also: Jumps to Language Descriptions and Help on Blocks, System Attributes
Reading the System Status List	If you use SFC 51 (RDSYSST), for example, to read out information on module status or module rack/station status, you must take into consideration the changed significance of the slots and of the additional slot 0 (see above).

Example 1: Evaluating Interrupt Info from OB 40 with SFB 54 "RALRM"

A distributed S7 digital input module (Start address 288) triggers a Hardware Interrupt. The supplementary interrupt information on this module is to be read from OB 40 via call of SFB 54 "DP_ALRM". It is checked whether the first channel has triggered a hardware interrupt.

You could also read out additional interrupt information with S7 modules directly from the start information of OB 40. However, the DPV1 standard generally permits up 59 bytes of additional interrupt information - too much for the OB 40 start information.

For information on the SFB 54 and the structure of additional interrupt information for the various interrupt types please refer to the manual "System Software for S7-300/400 System and Standard functions" or to the corresponding Online Help.

```
// ...

// ...

//Switch for address that triggered the interrupt (288)

     L    DW#16#120

     T    "MD10"


     CALL  "RALRM" , "DB54"

      MODE  :=1                 //function mode: 1 = set all output parameters (that is, F_ID
                                //has no effect)

      F_ID  :="MD10"            //start address of the slot from which an interrupt is
                                //permitted

      MLEN  :=8                 //max. length of supplementary interrupt info in bytes
                                //(for example, for module channel status)

      NEW   :="Alarm_neu"       //receive interrupt ? (yes = 1)

      STATUS:="DP_RALRM_STATUS"  //Return value with function result/error message

      ID    :="Slotadresse_Alarm"//start address of the slot from which an interrupt was
                                //received

      LEN   :="Laenge_Alarminfo" // length of supplementary interrupt info (4 bytes header
                                //info + ,for example, 4 bytes with S7 I/O modules)

      TINFO :=P#M 100.0 BYTE 28  //pointer for OB start info + management info: 28 bytes
                                //as of MB 100

      AINFO :=P#M 130.0 BYTE 8   //pointer for target area of the header info + supplementary
                                //interrupt info (max. 59 bytes)

     U    M   124.0             //has input 1 (bit 0) triggered the interrupt?

     SPB  Alrm

     BEA

Alrm: S    A     0.0            // interrupt handling

// ...
```

Example 2: Evaluation of Diagnostic Data in OB 82 with SFB 54 "RALRM"

The target area for diagnostic data must be large enough to hold standard diagnostics (6 bytes), identifier specific diagnostics (3 bytes for 12 slots) and the evaluation of device-specific diagnostics (module status only, which requires 7 more bytes).

Extended evaluation (channel-specific diagnostics) would require the reservation of additional bytes, provided the DP Slave supports this function.

```
// ...

// ...

     L    120                        //determine the start address for the module/station,

     T    "Slotadresse_Diag"         //from which the diagnosis is to be fetched




     CALL  "RALRM" , "DB54"

      MODE  :="Alle_Params"          // 1 = all output parameters are set

      F_ID  :="Slotadresse_Diag"     //start address of the slot from which the diagnosis is to
                                     //be fetched

      MLEN  :=20                     //max. length of diagnostic data in bytes

      NEW   :="neu"                  //irrelevant

      STATUS:="RET_VAL"              //function result, error message

      ID    :="Slotadresse_Alarm"    //start address of the slot from which an interrupt was
                                     //received

      LEN   :="Laenge_Alarminfo"     //length of the supplementary interrupt info (4 bytes header
                                     //info+16 bytes diagnostic data

      TINFO :=P#M 100.0 BYTE 28      //pointer for OB start info + management info: 28 bytes as
                                     //of MB 100

      AINFO :=P#M 130.0 BYTE 20      //pointer to the target area in which the diagnostic data
                                     //is to be stored


// ...

//Structure of the stored diagnostic data:

//   MB 130 to MB 133: header info (length, identifier, slot)

//   MB 134 to MB 139: Standard Diagnostics (6 bytes)

//   MB 140 to MB 142: identifier specific diagnostics (3 bytes)

//   MB 143 to MB 149: module status (7 bytes)

// ...

     U    M   141.0                  //slot 1 error?

     SPB  stp1
     BE


stp1: L    MB   147                   //fetch module status slots 1 to 4
     UW   W#16#3                      //filter slot 1
     L    W#16#2                      //2-bit status 'wrong module', wrong module inserted
     ==I
     S    A     0.1                   //reaction to wrong module

     L    MB   147                    //fetch module status slots 1 to 4
     UW   W#16#3                      //filter slot 1
     L    W#16#1                      //2-bit status 'invalid data', invalid user data
     ==I
     S    A     0.2                   //reaction to invalid user data
//..
```

3.12.4 The Slot Model of DPV1 Slaves with I-Slaves

Subject of article below is the visualization of the assignment of addresses (I/O addresses and diagnostic addresses) to the slots within the DPV1 model. We shall pay close attention to the addresses that do not carry user data and especially to the configuration of those addresses.

The Slot Model for DPV1

With DPV1 (IEC 61158) a slave is built of slots in the same way as DP (EN 50 170). The slot numbers are 0, 1, ...n. Slot 0 - a new slot - is of high significance because it is representative of the complete DP Slave.

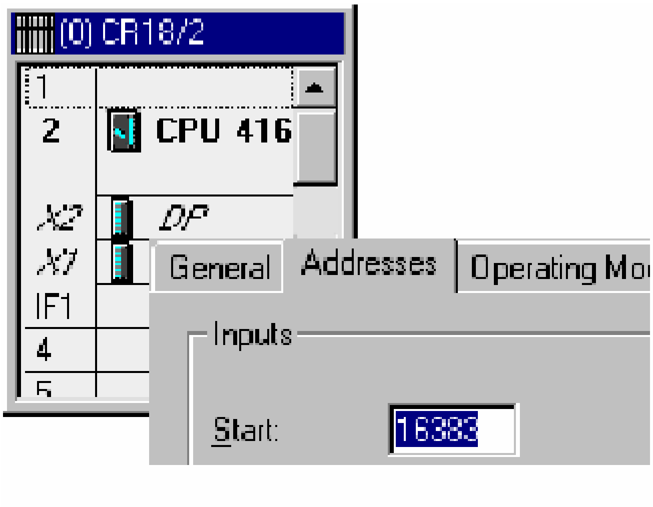
Representative means, for example, that interrupts triggered via slot 0 are assigned quasi to the global DP slave rather than to a certain slot within the DP Slave. Diagnostics output from this slot are globally assigned to the DP Slave rather than to any individual slot or module.

Excursion: Addresses for DP Interface

Seen from the CPU, a separate logical address is available for every one of your interfaces.

You can locate this address in the "Addresses" tab of the Master interface and of the Slave interface (double-click on the "DP" row in the configuration table).

These addresses have nothing in common with the slot model of DP Slaves. Rather, they are used by the CPU internally for the identification of, for example, the failure of an interface. This address is of little significance to the user program.



Start: 16383

Slots and Addresses for User Data

Generally speaking, the manufacturer of a DP Slave can freely choose what kind of data he assigns to any slot.

The first I/O module of DP Slaves (often referred to as "S7 Slaves") configured in Step 7 via the internal STEP 7 module knowledge base is always located in slot 4. In contrast, the DP Slaves that are installed in Step 7 via the GSD file can contain user data as of slot 1.

Distributed peripheral data are usually addressed in the same way as centralized peripheral data via their addresses. Therefore, for S7 Slaves the user data are always addressed as of the start address of slot 4.

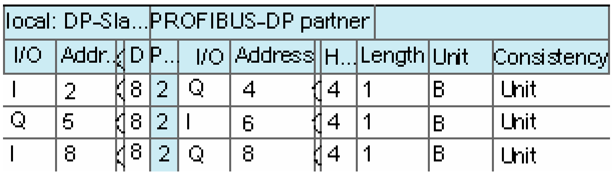
This is also valid for intelligent DP Slaves. For intelligent DP Slaves you can assign the I/O memory area of the Slave via a table ("Configuration" tab) to the I/O memory area of the Master. In operate state (cyclic data exchange) the data that you transfer in the user program of the intelligent DP Slave to these memory areas are transferred to the assigned Master memory areas.

However, the slot number remains hidden when you configure the addresses, because the slot limits are not formed by real modules (for example, with ET 200M). This is formed rather by a freely customizable length of the respective I/O area. In such cases we also speak of "virtual" slots.

Important for the assignment of addresses is:

- In addition to "real" slots, the memory area of an intelligent Slave also has "virtual" slots.
- Virtual slots are addressed in the same way as real slots, namely via their logical address. For "Standard" DP Slaves such as ET 200M this happens via module start address, and for an I-Slave via the address configured in the "Configuration" tab (I/O area).
- The addresses of the virtual slots are different from the DP Master point of view than from the DP Slave. The assignment is configurable. Therefore, the DP Master and the DP Slave as a rule use different addresses to address one and the same DP Slave slot.

Example of an Address Assignment for User Data



local: DP-Sla...				PROFIBUS-DP partner							
I/O	Addr.	(	D	P...	I/O	Address	(	H...	Length	Unit	Consistency
I	2	(	8	2	Q	4	(	4	1	B	Unit
Q	5	(	8	2	I	6	(	4	1	B	Unit
I	8	(	8	2	Q	8	(	4	1	B	Unit

Previously the assignment of "virtual" slots was as follows.

Example address from the view of the DP Slave	Significance (for DP Slave)	Slot (not visible when configuring)	Significance (for DP Master)	Example address from the view of the DP Master
		0		
		1		
		2		
		3		
E 2	Reading via input byte 2 that ...	4	... what the Master wrote into output byte 4.	A 4
A 5	What was written to output byte 5 in the Slave ...	5	... can be read in the Master as input byte 6.	E 6
E 8	...	6	...	A 8
		...		
		35		

Tip: the slot assignment is displayed in the address overview of the Master CPU or Slave CPU.

Slots and Addresses for System Information

Addresses for system information are used, for example, to handle diagnostic information or information on operating state transitions.

Addresses of the DP Slave

The system information of DP Slaves is also assigned to slots. Relevant for operating mode DPV1 in this context are the following slots:

- Slot 0 (Station representative):
  Via the address of this virtual slot, seen from the view of the **DP Master,** the DP Master diagnoses the failure or return of the intelligent DP Slave.
  Via the address of this virtual slot, seen from the view of the **DP-Slave,** the intelligent DP Slave diagnoses the failure or return of the DP Master.
- Slot 2 (for "Standard" DP Slaves the DP interface):
  Via the address of this virtual slot, seen from the view of the **DP Master,** the DP Master can detect an operating state transition of the DP Slave.
  Via the address of this virtual slot, seen from the view of the **DP-Slave,** the DP Slave can detect an operating state transition of the DP Master.
- Slots 1 and 3 are not relevant for intelligent DP Slaves.

In the table below you can find an assignment for slots 0 to 3 ("virtual" slots). The tab designations relevant for the configuration of the Master station and of the Slave station are listed below the table.

In STEP 7 addresses are automatically assigned "from top to bottom" in order to avoid conflicts with user data. You should apply the recommended addresses, even though you can edit them. Check whether or not the size of the addresses area matches the "smallest" CPU in case the user program is to run on different CPUs.

Example address from the view of the DP Slave	Significance (for DP Slave)	Slot (not visible when configuring)	Significance (for DP Master)	Example address from the view of the DP-Masters
8189	Station failure / station return of the DP Master (see **1**)	0	Station failure / station return of the DP-Slaves (see **3**)	16381
-	Not relevant	1	Not relevant	-
8188	Operating state transition of the DP Master (see **2**)	2	Operating state transition of the DP Slave (see **4**)	16380
-	Not relevant	3	Not relevant	-
	User data (see above)	4 ... 35	User data (see above)	

(1) Double-click on the DP interface of the intelligent DP Slave (e.g., CPU 414-3 DP) in the Slave station, "Configuration" tab; input is possible in the table, "Diagnostics" field.

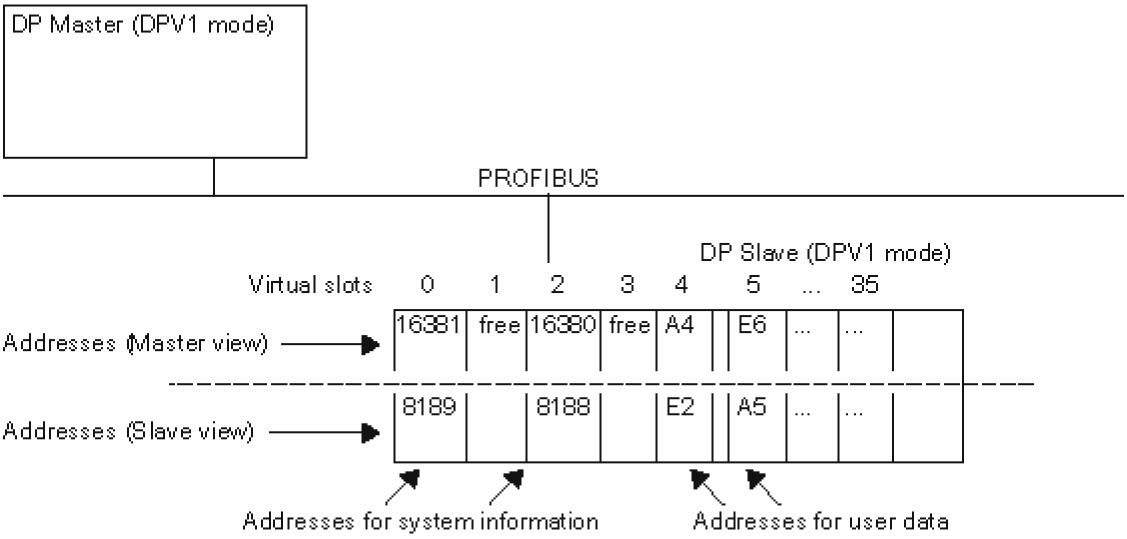
(2) Double-click on the DP interface of the intelligent DP Slave (e.g., CPU 414-3 DP) in the Slave station, "Operating Mode" tab; input is possible under the option "DP Slave" in the field "Address for virtual slot 2".

(3) Double-click on the DP Slave icon in the Master station, "General" tab; input is possible under "Addresses" in the "Diagnostic address" field.

(4) Double-click on the DP Slave icon in the Master station, "General" tab; input is possible under "Addresses" in the field "Address for virtual slot 2".

Summary

With openly presented virtual slots, the configuration of the intelligent DP Slave looks as follows:



Triggering Hardware Interrupts with SFC 7

With SFC 7 you can trigger a hardware interrupt for any configured address via the user program of the I-Slave CPU. This also applies to user data addresses of the I/O range as well to the address of the virtual slot 2.

In the I-Slave user program, for example, use the I/O addresses configured in the "Local..." column for the SFC 7.

A hardware interrupt is then triggered in the user program of the Master. In the start information of the hardware interrupt OB (e.g., OB 40) the address you have configured in the column "PROFIBUS-DP-Partner" is passed on as the address that has triggered the interrupt.

3.13 Diagnostic Repeater

3.13.1 Configuring and Commissioning the Diagnostic Repeater

This Diagnostic Repeater can monitor the segment of an RS485 PROFIBUS subnet (copper wire) during operation and report cable errors per diagnostics telegram to the DP Master. Via an HMI you can display the error location and the cause in plain text.

The Diagnostic Repeater with its line diagnostic feature offers the possibility to recognizing and localizing line errors at an early stage and while in operation, thus reducing system standstill.

Configuring the Diagnostics Repeater

You can locate the Diagnostic Repeater in the hardware catalog under "PROFIBUS DP\Network components\Diagnostic Repeater". The Diagnostic Repeater must be configured like any "Standard Slave" (it is connected to the master system of a DP Master).

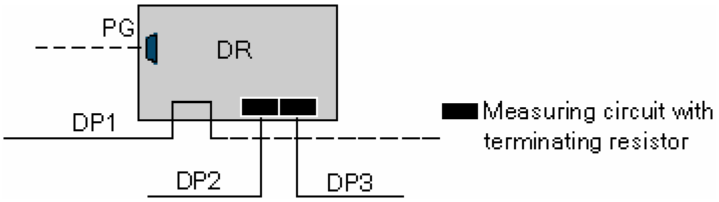
Function of the Diagnostic Repeaters

To locate anomalies during operation, the Diagnostic Repeater must know the topology of the PROFIBUS subnet to which it is connected. The Diagnostic Repeater measures the distance to all partners via "Prepare line diagnostics " function.

The Diagnostic Repeater measures the partner distances and stores them internally in a table. The Diagnostic Repeater also memorizes in which segment it has detected the partner.

After it has measured the distance to a segment error during operation, the table entries can be used to determine between which of the partners the segment fault is pending.

The Diagnostic Repeater interconnects 3 segments. During operation the Diagnostic Repeater can determine the topology and localize segment errors only in **Segments DP2 and DP3**, because they are the only ones equipped with measuring lines. The figure below displays the Diagnostic Repeater (DR) and its connections.



Prerequisites for Commissioning

The following prerequisites must apply:

- To be able to start capturing the topology, the PG must be connected to the PROFIBUS network.
- The structure of the PROFIBUS subnet with connected Diagnostic Repeater corresponds with the specifications and rules of the Diagnostic Repeater documentation.

Commissioning the Diagnostic Repeater with STEP 7

In order to be able to locate a disturbance during operation, the Diagnostic Repeater must know the topology of the PROFIBUS subnet to which it is connected. The Diagnostic Repeater measures the distance to all partners via "Prepare line diagnostics " function.

The Diagnostic Repeater measures the partner distances and stores them internally in a table. The Diagnostic Repeater also memorizes in which segment it has detected the partner.

After it has measured the distance to an error during operation, the table entries can be used to determine between which of the partners the disturbance is pending.

You must explicitly instruct the Diagnostic Repeater to determine the distance between PROFIBUS partners when you configure your hardware or network:

1. Highlight the Diagnostic Repeater or the DP Master System to which it is connected (configure the hardware), or
   highlight the PROFIBUS subnet to which the Diagnostic Repeater is connected (network configuration)
2. Select the menu command PLC > Prepare line diagnostics.
3. Start the measurement via the subsequently opened dialog.

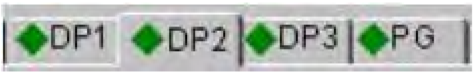
Locating the Error During Operation

During operation the Diagnostic Repeater reports the event "Found Error" to the DP Master CPU.

You can view detailed information on pending diagnostic events in the dialog for the module status of the Diagnostic Repeater. The error is displayed visually in the dialog with additional information, for example, showing the error cause (provided it can be detected by the Diagnostic Repeater).

Example of Visual Presentation in the Dialog "Module Status"

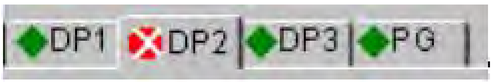
If the functions of all segments to which the Diagnostic Repeater is connected are free of error, the corresponding tabs in the dialog "Module Status" are displayed as follows:



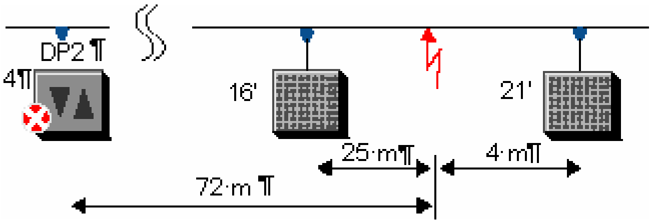
If a segment is switched off (that is, it is not possible to diagnose it), the following symbol will appear at the side of the register title:

▽ Symbol for a switched off segment

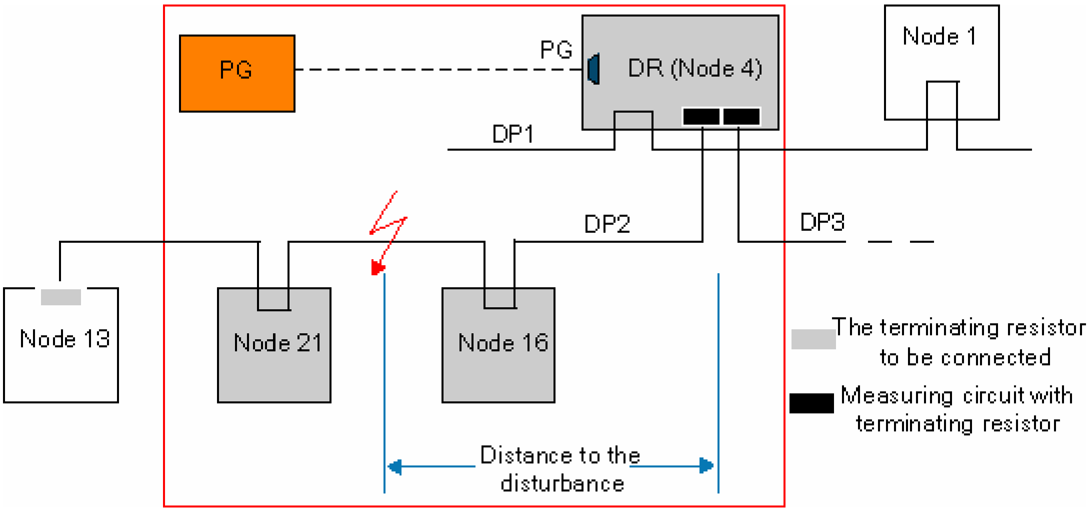
An error in segment "DP2" results in an error symbol at the side of the "DP2" tab identifier. The remaining segments are free of error:



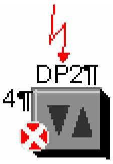
The "DP2" tab could show the disturbance in the following view: The Diagnostic Repeater is assigned to PROFIBUS address 4, the error is located between the partners assigned to the PROFIBUS addresses 16 and 21. In addition, the view shows the distance to the neighboring DP Slaves.



In the figure below you can find an example of a simplified detailed presentation of the arrangement illustrated above.


PG
PG
DR (Node 4)
Node 1
DP1
DP2
DP3
Node 13
Node 21
Node 16
Distance to the disturbance
The terminating resistor to be connected
Measuring circuit with terminating resistor

If STEP 7 does not localize the error in segment "DP2" or if, for example, this segment contains more than 32 partners and the Diagnostic Repeater cannot operate correctly anymore, you are shown the following image:



Summary of all Symbols

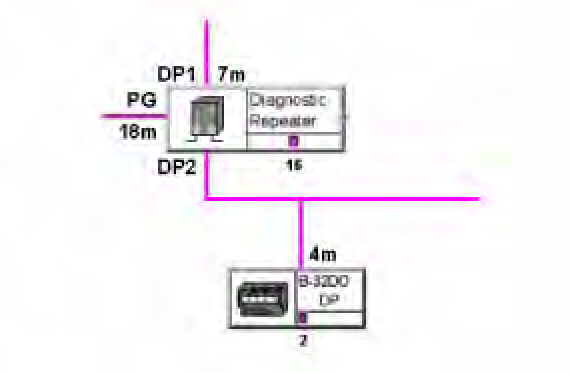
The symbols o the registers can be of the following forms:

◆	No segment error
✖	Segment error
▼	Segment is switched off
■	Information cannot be retrieved from the segment

3.13.2 Displaying the Topology With the Help of Diagnostic Repeaters

As of STEP 7 V5.2, in combination with a PROFIBUS configuration with Class 2 diagnostic repeaters 972-0AB01 it is not only possible to perform line diagnostics, but also to display the PROFIBUS DP network topology.

In contrast to the network view under NetPro, this function does not display the "Logic" view of a PROFIBUS subnet, but rather the physical arrangement of the PROFIBUS nodes in their actual order as well as the distances to the nodes - provide the diagnostic repeater was able to determine these data. The actual nodes are displayed same as under NetPro.



Function

Prior to displaying the topology, the "Prepare line diagnostics" function must be called initially after each modification of the hardware structure, in order to enable the diagnostic repeater to measure the PROFIBUS subnet and generate internal distance tables.

The "Display the PROFIBUS Network Topology" function is used to visualize these data.

If you select a subnet in an opened project and then display the topology, the nodes in the subnet are displayed along with their configured names.

In addition to the visualization, the entries in the diagnostics buffer of the diagnostic repeater as well as statistical data can be read and displayed.

You can write these data to a file and print it.

Requirements

The diagnostic repeaters must support the "Display the PROFIBUS Network Topology" function (i.e. as of order no. 6ES7 972-0AB01).

The structure of the PROFIBUS network must be in compliance with the guidelines in the diagnostic repeater manual, to be able to determine distances correctly. Cascaded diagnostic repeaters, for example, may only be connected to a master diagnostic repeater via a DP1 interface.

To call the "Prepare line diagnostics" function, the PG must be connected directly to the same PROFIBUS as the diagnose repeater. The "Prepare line diagnostics" function can also be called without open project.

To enable the "Display the PROFIBUS Network Topology" function, you can also connect the PROFIBUS network with its diagnostic repeaters via a "Data record router" (z. B. CP 443-5 Ext V3.2) to the PG. The PG must have been assigned in the STEP 7 project (Under NetPro, use menu command **PLC > Assign PG/PC** to configure the "PG/PC" object). To enable the display of a network topology via routed diagnostic repeater, you need to open the corresponding project and select the participating PROFIBUS subnet.

Procedure

Do one of the following:

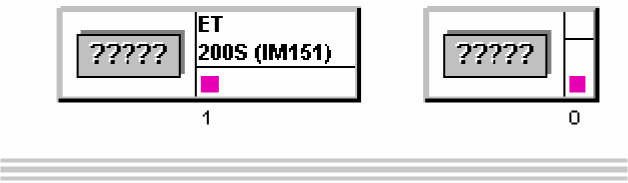
1. In NetPro or in HW Config, select the menu command **PLC > Prepare Line Diagnostics**.
2. In the SIMATIC Manager, select the menu command **PLC > PROFIBUS > Display Network Topology**; or in NetPro, select the menu command **PLC > Display PROFIBUS Topology**.

Alternative: In the user program, use SFC 103 ("DP_TOPOL") to determine the topology.

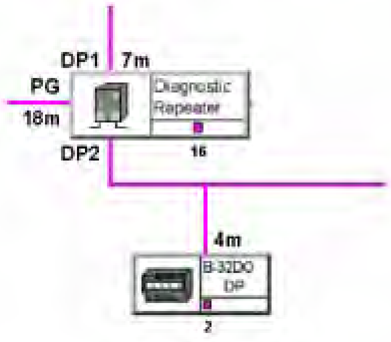
3.13.3 Working With the Topology View

Display of the Nodes

The upper section of the "Topology display PROFIBUS DP" window shows the nodes that cannot be assigned.

Representation	Meaning
	Nodes cannot be assigned. The possible causes are indicated in a message in the window: Nodes have been added, node addresses have been changed and afterwards the function "Prepare Line Diagnostics" was not started. If diagnostic repeaters are connected that do not support the reading of topology data, this is also reported in the upper section of the window. An unknown node is indicated by a series of question marks.

The lower section of the window shows the nodes that can be assigned as networked nodes along with the distance information determined and, if necessary, other information.

Representation	Meaning
 DP1 7m PG 18m Diagnostic Repeater 16 DP2 4m B-32DO DP 2	Nodes can be assigned and represented in the PROFIBUS topology. Other information such as a faulty configuration (such as when the measuring segments of two diagnostic repeaters are directly connected) will be reported in a message. Representation of the cable lengths (in the example): The cable length between the DP slave with the PROFIBUS address 2 and the diagnostic repeater (PROFIBUS address 16) is 4 meters. The DP slave is connected at segment DP2.
⊘	Node can be assigned, but cannot be reached by the diagnostic repeater at this time.
⊗	Node can be assigned, but has been detected as faulty by the diagnostic repeater

How to find nodes in the topology view

In larger configurations you can locate a required node via the menu command **Options > Go To**. The next dialog box "Go To" displays all nodes of the PROFIBUS network:

1. Select the required node (e.g. a DP slave)
2. Click on the "Node" button to display the node in the center of the window.
   Click on the "Diagnostic repeater" button to display the assigned in the center of the window.

Display of the topology in tables

If you prefer to display the topology in a table rather than graphically, call menu command **View > Table > Topology**.

Preparing line diagnostics

Use the same procedure as in HW Config or under NetPro. In the topology display, call menu command **PLC > Prepare Line Diagnostics**.

Calling the module status

Use the same procedure as in HW Config or under NetPro. In the topology display, call menu command **PLC > Module Status**.

How to save and open topology data

Select the menu command **File > Save** or **File > Save As** to save the current display. With this function, it is easy to save the data you have acquired online for the purpose of later diagnostics and error evaluation.

How to export topology data

You can use the following view to export topology data:

- The "Table" view (after calling menu command **View > Table**)
- The "Statistics" dialog box (after calling menu command **PLC > Statistics**)
- The "Diagnostics buffer" dialog box (after calling menu command **PLC > Diagnostics Buffer**)

The CSV (ASCII) export format can be read and edited with other applications.

The topology display can no longer read exported data.

How to find data on reflection errors and message frame errors (statistical data)

Reflection errors occur in cases such as if a line is interrupted or faulty or if there is no terminating resistor or there is an accumulation of too much terminating resistance.

Frame errors occur in cases such as when at least one bit (i.e. the parity bit) is corrupted due to faulty hardware.

You can log reflection errors and message frame errors detected by the diagnostic repeater in a window. You can then print or export the logged data, for example.

1. In the topology display, select the diagnose repeater whose data you want to read.
2. Start this function with the menu command **PLC > Statistics**.

The values will be displayed for 60 seconds, starting at the time the dialog is being opened. Further values will be accumulated internally across this interval. You can click on the "Export" button to export these values in CSV format.

The color-coding also helps you in determined the severity of the error. The color-coding shown is determined by an evaluation of the statistical data.

Click on "Print" to print the visible graphic object.

Fetching data from the diagnostics buffer

Similar to the diagnostics buffer function of the CPU, you can use this function to record a history of error events on the PROFIBUS. Select menu command **PLC > Diagnostics Buffer** to start this function. The next dialog displays the last 10 events. Click on an event to display details in the lower section of the dialog box.

If a "DPx" tab (i.e. the "DP2" tab) in the "Diagnostic Buffer" dialog box indicates that the segment is faulty, then there is an incoming error. Under certain conditions this error is no longer contained in the diagnostic buffer.

To display the current status, select the men command **PLC > Module Status**.

Printing the topology display

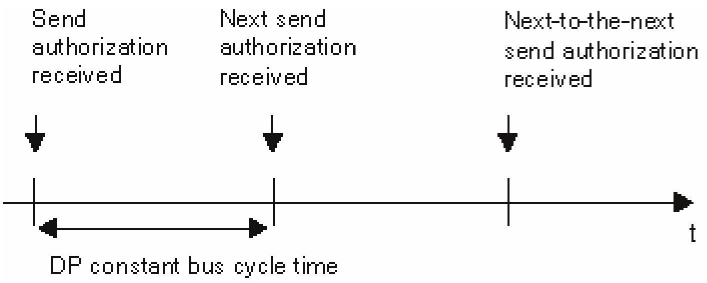
Select **File > Print** to print the topology data. In the dialog box that is then displayed, you can set up the printer, select the area to printed and specify a notes field.

3.14 Setting Constant Bus Cycle Times for PROFIBUS Subnets

Introduction

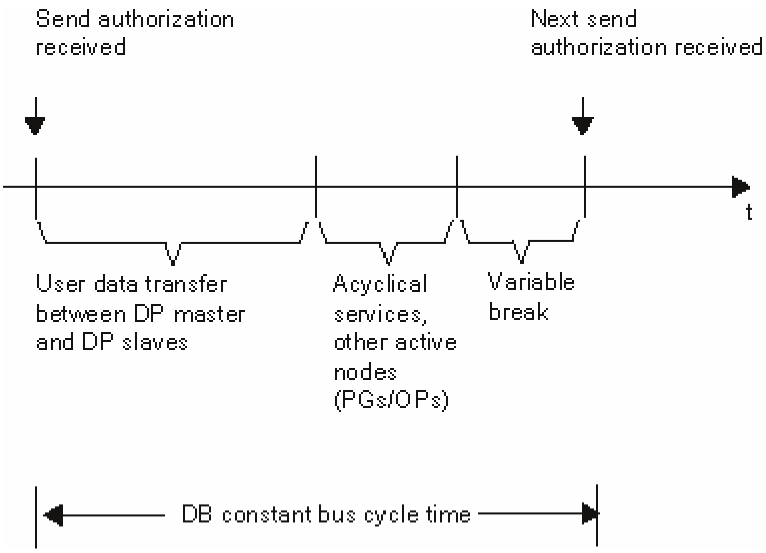
For PROFIBUS subnets you can set constant (of equal length) bus cycle times in STEP 7.

Constant bus cycle time is the property of the PROFIBUS-DP that guarantees bus cycles that are exactly the same length. "Bus cycles of the same length" means that the DP master always begins the DP bus cycle after the same time interval. From the viewpoint of the connected slaves, this means that they also receive their data from the master in time intervals of exactly the same duration.


Send authorization received
Next send authorization received
Next-to-the-next send authorization received
t
DP constant bus cycle time

Bus Cycle Time

The following figure shows how the time for a bus cycle is made up.


Send authorization received
Next send authorization received
t
User data transfer between DP master and DP slaves
Acyclical services, other active nodes (PGs/OPs)
Variable break
DB constant bus cycle time

The "variable pause" shown in the figure is always minimal if communication jobs, for example, for other active nodes are still pending. The master (also known as the constant-bus-cycle-time master) controls the communication parts so that the same duration for a bus cycle is always achieved.

Requirements

- The constant-bus-cycle-time master must support the "Constant bus cycle time" function (see info text in the hardware catalog).
- The constant-bus-cycle-time master must be a class 1 DP master. This means a PG/PC can never be a constant-bus-cycle-time master.
- The constant-bus-cycle-time master is the only active station on the PROFIBUS-DP. Only one DP master system may be present on the PROFIBUS subnet. Programming devices or PCs can also be connected.
- Constant bus cycle time is possible only for the "DP" and "User-Defined" bus profiles.
- CiR must not be configured.
- No H-CPU may be connected to the PROFIBUS subnet.
- The PROFIBUS subnet must not be a cross-project one.

Time for DP Constant Bus Cycle Time

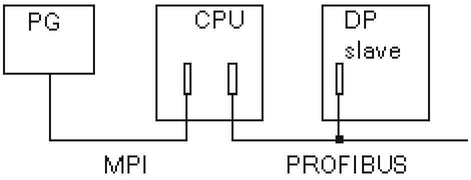
STEP 7 calculates a recommended time for the "DP constant bus cycle time(ms)" based on:

- The PROFIBUS configuration (number of configured nodes, number of programming devices etc.)
- Other information for the calculation which can be specified as an option (for example, any additional not configured programming devices to be taken into account)

You can correct this time but not below the calculated and displayed minimum value.

Influence of Connected Active Nodes (PGs/PCs and I Slaves)

A PG/PC must only be taken into account if it is connected directly to the PROFIBUS via its PROFIBUS interface. It does not need to be taken into account if it is connected via the multipoint interface of the CPU, as shown in the following figure.



If intelligent DP slaves (for example, CPU 315-2DP) are connected, the time for the DP constant bus cycle should be calculated generously.

Constant Bus Cycle Time Behavior

When recalculating times, STEP 7 recommends a value for the constant bus cycle time. This value is based on the configuration in question. However, you can change this value.

When STEP 7 calculates this value for the constant bus cycle time, it includes in its calculation the user data traffic of the DP master as well as an allowance for a few errors that may possibly occur.

STEP 7 also calculates a minimum value for the constant DP bus cycle time. The cycle time cannot fall below this value. When STEP 7 calculates the minimum value, it considers only the normal message frames for each bus cycle. If errors occur, there may be violations of the constant bus cycle time.

Times that are longer than the recommended times are possible without any problems.

⚠ **Caution**

If you select times that are shorter than the time that the system recommends, under certain circumstances the communication of the additional active nodes that are connected to the PROFIBUS subnet is delayed, or, in the worst case, comes to a standstill. If you set values close to the minimum possible constant bus cycle time that is displayed, bus faults can cause the entire PROFIBUS subnet to be shut down in certain cases.

Relationship: Constant Bus Cycle Time and SYNC/FREEZE

For PROFIBUS-DP, if you configure both "constant bus cycle time" and SYNC/FREEZE groups, note the following:

- Group 8 must not be used (reserved for constant bus cycle time clock). If you configure the group assignment first and have assigned group 8, you can no longer set constant bus cycle time.
- If you configure group 7 when the constant bus cycle time is set, you cannot use the SYNC or FREEZE functions for the slaves in this group.

Procedure

1. Configure a PROFIBUS subnet with a DP master that supports the "constant bus cycle time" function (see the info text in the "Hardware Catalog" window of Hardware Configuration).
2. In the Network view, double-click on the PROFIBUS subnet.
3. In the Properties dialog box ("Network Settings" tab), select the "DP" profile and click the "Options" button.
4. In the "Constant Bus Cycle Time" tab, set the constant bus cycle time behavior that is appropriate for your application and, where necessary, adapt the times to be considered and the connected programming devices/operator panels. You can find detailed information on the possible settings by clicking the Help button in the dialog box. If the "Constant bus cycle time" tab is not displayed, this means that not all the requirements for operation with constant bus cycle time have been met (see above).

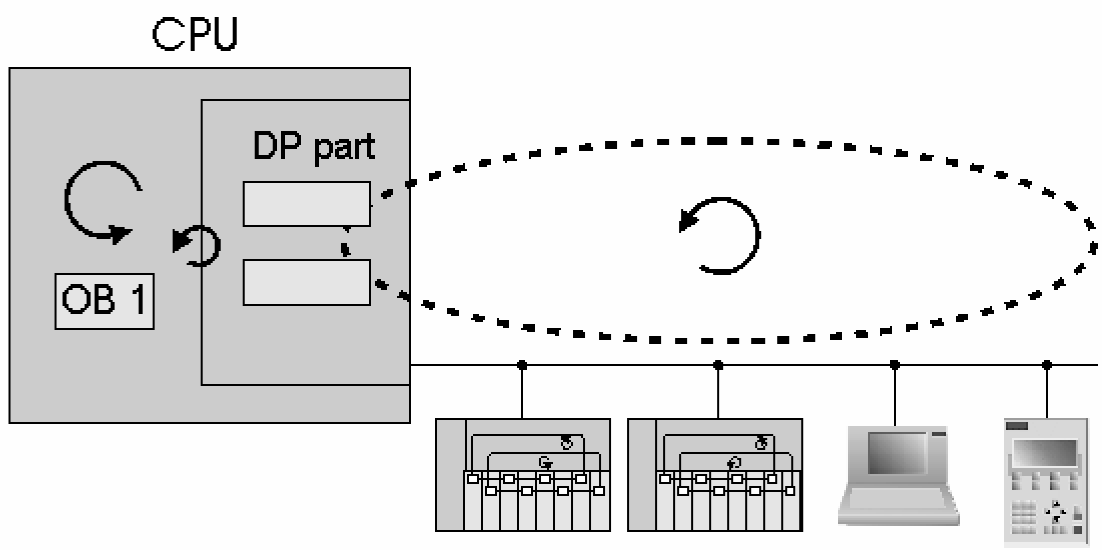
Additional Information

You can find more detailed information on setting the constant bus cycle time in the Help for the tab dialogs.

3.14.1 Configuring Short and Equal-Length Process Reaction Times on PROFIBUS DP

Process Reaction Times Without Constant Bus Cycle Time and Isochronous Mode

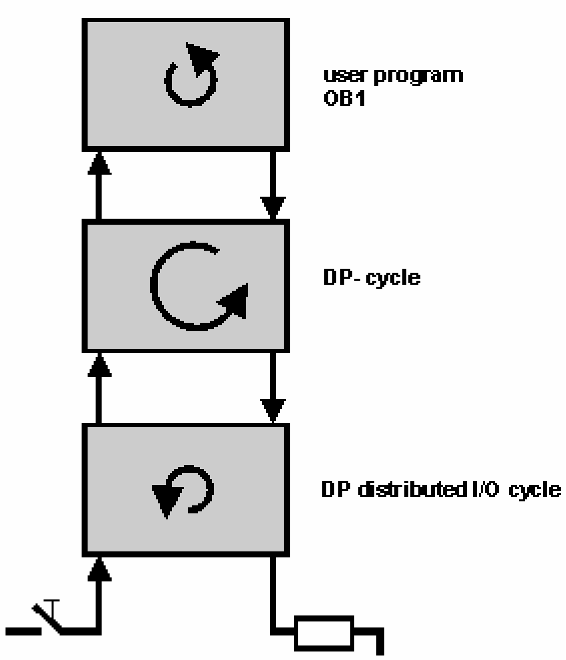
If a drive engineering or other application requires short and reproducible (that is, repeatable, of equal length) process reaction times, then the individual free cycles of sub-components can have a negative effect on the overall reaction time.



In the previous example, the behavior is depicted without constant bus cycle time and cycle synchronization by using a model structure with a DP master, two DP slaves, a programming device (PG) and an OP. This configuration yields the following subcycles with their own cyclic and acyclic portions:

- Free OB 1 cycle of the user program. A cyclic program branching can cause variations in cycle time length.
- Free, variable DP cycle on the PROFIBUS subnet consisting of:
  - Cyclic master-slave data exchange, DP slave 1.
  - Cyclic master-slave data exchange, DP slave 2.
  - Acyclic portion for interrupts, bus acceptances or diagnostic services.
  - Forwarding the token to a programming device (PG), followed by its processing.
  - Forwarding the token to an OP, followed by its processing.
- Free cycle on the DP slave backplane bus.
- Free cycle for signal preparation and conversion within the electronic submodules on the DP slave.

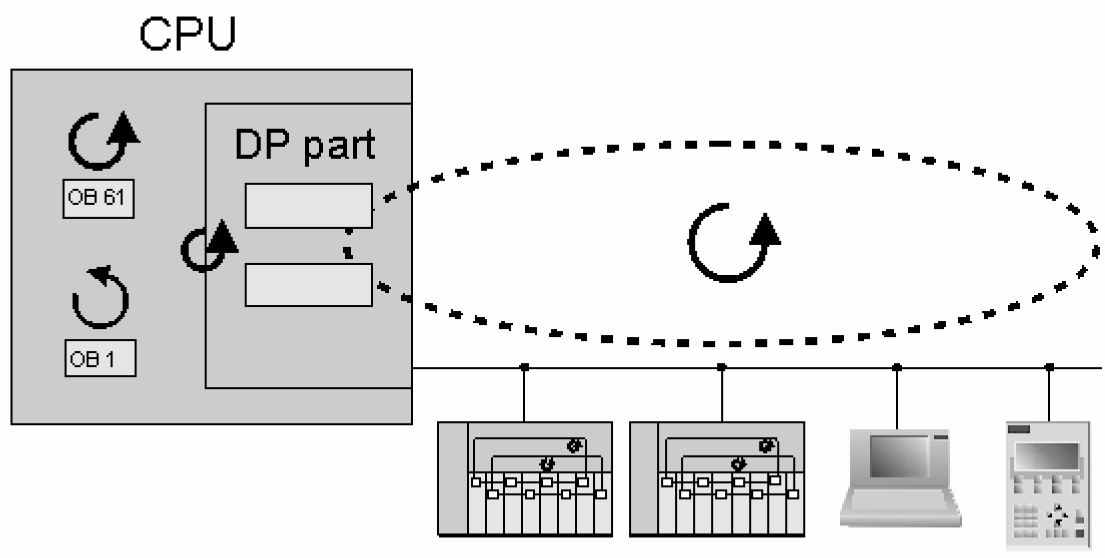
If especially short and secure process reaction times are desired, then free cycles with different lengths have a definite effect on process reaction times.



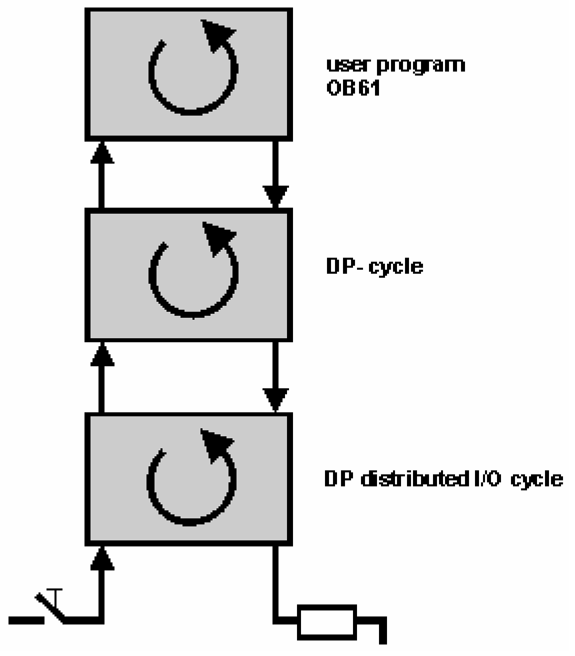
With regard of the individual cycling of the input electronic module, signal or data exchange occurs through the DP slave backplane bus, the master-slave data exchange on the PROFIBUS subnet on to the OB 1 user program on the CPU. The process reaction is determined in the OB 1 user program and is then sent back over the same route to the output electronic submodule. The different lengths and the "random" position of individual cycles have a pronounced effect on process reaction time. Depending on the position of the individual cycles, information transmission can either occur immediately or after two cycles.

Process Reaction Times with Constant Bus Cycle Time and Clock Synchronization

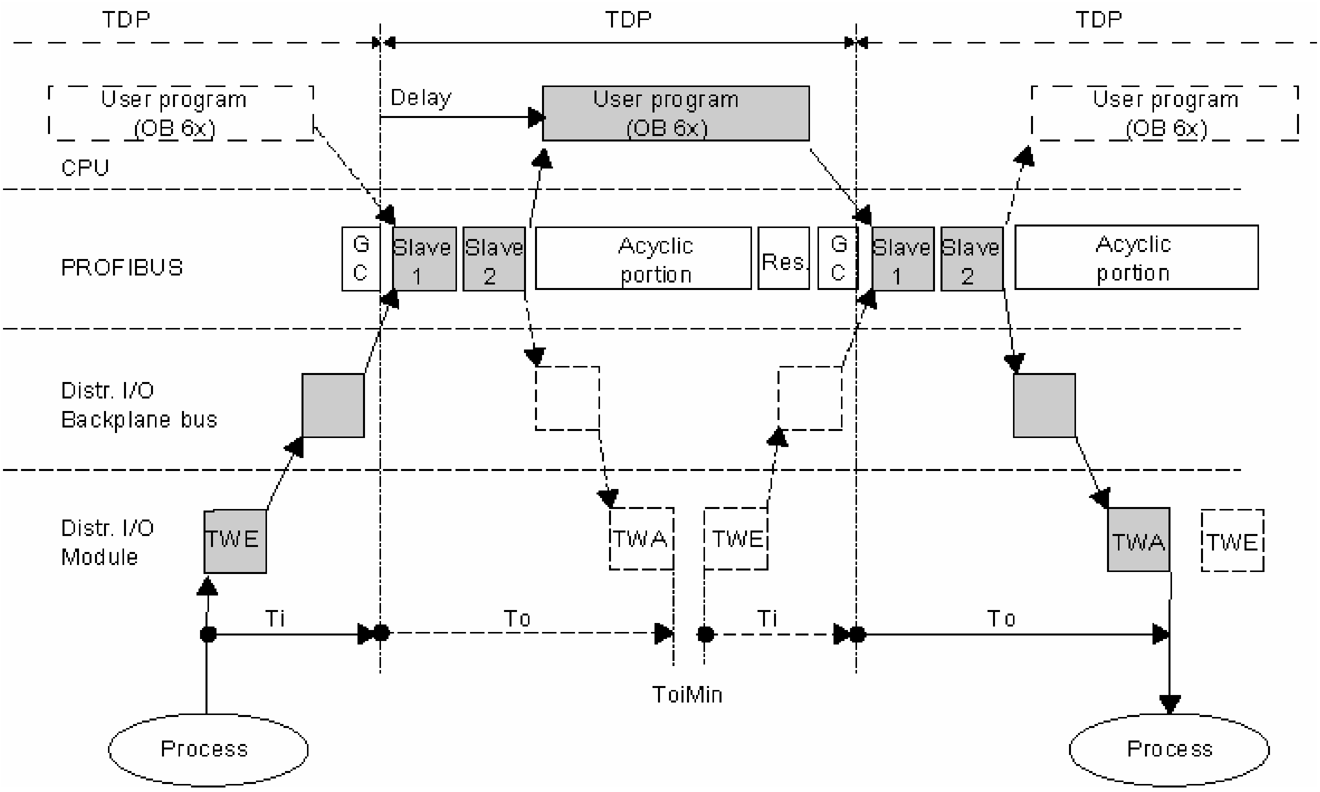
SIMATIC produces reproducible (that is, repeatable, of equal length) reaction times by means of a constant (isochronous) DP bus cycle and synchronization of the individual cycles previously listed.



In this case, the situation corresponds to the example given above with the difference that all cycles (up to the OB 1 cycle) are of equal length and synchronously cycled. The clock pulse generator is comprised of the DP master constant bus cycle time clock that is sent as the global control frame to the DP slaves. A synchronous cycle interrupt OB 61 (or OB 61 to OB 64) ensures that it is synchronized with the user program.



With constant bus cycle time and cycle synchronization, all cycles concerned have the same cycle time and length. This keeps the process reaction times of equal length and, because there are no cycle jumps, they are also shorter. This means that the previously described case in which information transmission can occur in the first or second cycle, depending on the position of individual cycles, now no longer applies.



In the previous example, the DP master handles the cyclic master-slave data exchange with slaves 1 and 2. After this comes the processing of the acyclic portions for interrupts, bus acceptances or diagnostic services. The DP master then retains it for a reserve time until the configured constant DP bus cycle time has expired in order to compensate for possible network disturbances and retrieve possible message frame repeats. At this point, a new DP cycle starts with the global frame (GC).

To ensure that consistent status information for the DP inputs can be read at the start time of a new DP cycle, the read process must be moved up by the specified time Ti. This time Ti includes the time for signal preparation and conversion at the electronic submodules and the time for processing at the inputs on the DP slave backplane bus.

When using a SIMATIC WinAC RTX (as of V3.1) the following applies: After the input data for all DP slaves have been read by the DP master, the cycle-synchronized user program is started automatically (OB 6x).
When using a SIMATIC S7-300/400 the following applies: The start of the cycle-synchronized user program is influenced by a configured "time delay".

The time To ensures that the process reactions for the user program are sent consistently and in the equal amounts of time to the terminals of the DP I/O devices. This time To includes the time for the cyclic master-slave data exchange for all DP slaves, the time for signal preparation and conversion at the electronic submodules and the time for processing at the outputs on the DP slave backplane bus.

The period starting at the point when an input is detected at the electronic module to the reaction at an output results in a constant processing time of **Ti + TDP + To**. This condition ensures a constant process reaction time described by: **TDP + Ti + TDP + To**.

Prerequisites and General Conditions

- H-systems (redundant/fault-tolerant) do not support cycle synchronization.
- In F-systems, cycle synchronization cannot be used for non-failsafe I/O devices/peripherals.
- Cycle synchronization cannot be used on optical PROFIBUS networks.
- Constant bus cycle time and cycle synchronization are only possible with the "DP" and "User-defined" bus profiles. However, using the "User-defined" profile is not recommended.
- Cycle synchronization is only possible with DP interfaces integrated in the CPU.
- At a cycle-synchronized PROFIBUS-DP, only the constant-bus-cycle-time master can be the active station. OPs and programming devices(PG) (or PCs with PG functionality) influence the timing of the constant-bus-cycle-time DP-cycle. For this reason, they are not recommended.
- Cycle synchronization among chains is not possible at this time.
- Cycle-synchronized I/O devices can only be processed in process image partitions (part process images). Process image partitions are required to achieve consistent, cycle-synchronized data transmission. Without them, consistent, cycle-synchronized data transmission is not possible. To ensure that a process image partition remains consistent, STEP 7 monitors the quantity of data (the number of slaves and number of bytes per process image partition for the DP master system are limited). In addition, please observe the following points:
  - Within a station, input addresses must not be assigned to different process image partitions.
  - Within a station, output addresses must not be assigned to different process image partitions.
  - A common process image partition can be used for both input and output addresses.
- In HW Config, the address of the cycle-synchronized analog I/O devices must be located in the address area of the process image partition.
- Cycle synchronization is only possible with ET 200M and ET 200S devices; synchronization with centralized I/O devices is not possible.
- Full cycle synchronization "from terminal to terminal" is only possible if all components in the chain support the "Isochronous mode" system property. When selecting devices in a catalog or in the hardware catalog, make sure that the information field for the modules contains the entry "Isochronous mode". The latest updated list is available in the Internet at http://support.automation.siemens.com/WW/view/de/14747677.

3.14.2 Assigning Parameters for Constant Bus Cycle Time and Isochronous Mode in HW Config

Introduction

A station consists of the following isochronous components that you have to set up in HW Config:

- CPUs with integrated DP interfaces (i.e. CPU 414-3 DP, V3.1)
- DP interface modules (i.e. ET 200S interface module IM 151-1 High Feature)
- Distributed I/O modules (i.e. DI 2xDC24V, High Feature [131-4BB00], DO 2xDC24V/2A, High Feature [132-4BB30])

The latest updated list is available in the Internet at "http://www.siemens.com/automation/support", Entry ID 14747353.

The following sections contain information on the special aspects of configuring these components for isochronous mode.

Setting CPU Properties

1. Select the "Synchronous Cycle Interrupt" tab.
2. The following settings have to be made for each synchronous cycle interrupt OB:
   - Specify the DP master system in use.
   - Specify the desired process image partition(s)
   - For S7-400-CPUs: Set the time lag. This time lag is the time between the global control frame and the start of OB 6x. This is the time period in which the DP master completes the cyclical data exchange with the DP slaves.
     Tip: After you have finished assigning parameters to the distributed I/O devices, be sure to let STEP 7 calculate the default value.

Settings at the DP Master System

To activate the constant bus cycle time at the DP master system:

1. Double-click the DP master system.
2. In the "General" tab, click on the "Properties" button.
3. In the "Properties - PROFIBUS" dialog box, select the "Network Settings" tab.
4. Select the profile allowed (i.e. "DP")
5. Click the "Options" button.
6. In the "Options" dialog box, select the "Constant Bus Cycle Time" tab and make the following settings:
   - Select the "Activate constant bus cycle" check box. This step activates the constant DP cycle as the basis for maintaining isochronous mode.
   - Select the "Times Ti and To same for all slaves" check box.
   - For the time being, leave the other parameters in their default settings.
7. Close this dialog box and any other open dialog boxes as well by clicking "OK" in each case.

Settings at the Modules in the DP Slave

The address space for each module that is involved in isochronous processing must be assigned to a process image partition. Isochronous read-in and output is can only be done by means of process image partitions.

1. Double-click the module.
2. Select the "Addresses" tab.
3. In the drop-down list, select the process image partition that you assigned to the synchronous cycle interrupt OB when you were assigning parameters to the CPU.
   If the addresses for the modules are outside the address range (i.e. for analog modules), then you can either select lower address that lies within the prescribed range of process image partition or change the size of the process image partition so that the address space for the module now lies in the process image partition. If choose to do the latter, go to the "Cycle/Clock Memory" tab and change the "Size of the Process Image" parameter. The value that you set here apples to all process image partitions.
4. To the extent reasonable, keep the "Input delay" parameter for digital input modules as low as possible. This is because short input delays result in a shorter Time Ti and thus in shorter overall reaction times. The critical setting here the longest input delay time for the DP slaves.

Settings at the DP Slave (DP Interface Module)

Isochronous input and output modules have to be made known to the DR interface modules (i.e. IM 151-1 High Feature) as isochronous components. To do so, proceed as follows:

1. Double-click the icon for the DP slave (i.e. IM 151-1 High Feature).
2. In the "Properties - DP Slave" dialog box, select the "Isochronous Mode" tab and make the following settings:
   - Select the "Synchronize DP slave to constant bus cycle time for DP cycle…" check box.
   - Select the modules desired for "isochronous operation." Modules that do not support isochronous mode or for which this option was not selected will not be included in a calculation of the Ti (read in process values) times and the To (output process values) times.
3. Confirm your entries and close the dialog box by clicking "OK".

After this, a message will appear to remind you that the Ti and To times in the configuration of the DP master system are not yet updated.

Updating the Times (Ti, To and the lag time)

To update the Ti and To times, go to the "Options" dialog box and select the "Constant Bus Cycle Time" tab, as previously described in the section "Settings at the DP Master System". Then click the "Recalculate" button.

The calculation process will enter a cycle time in the "Constant DP Cycle" field. This cycle time is one that will ensure adherence to the DP cycle time even in the case of strong interference (i.e. EMC related disturbances). Under very stable conditions, this value can be reduced down to the minimum value. The system requires that new values be changed in terms of the specified interval. For this reason, use a stepping switch to change this value. A greater DP cycle time may be necessary to ensure that OB6x has enough calculation time available to it.

During automatic calculation, the values for Ti and To are set to their minimum values. These values can also be changed and set within the limits shown. The maximum values for Ti and To can be extended by setting a greater constant DP cycle time.

To update the lag time between the global control frame and the call of the synchronous cycle interrupt OB, open the properties sheet for the CPU, select the "Synchronous Cycle Interrupts" tab and then click the "Default" button to have the value recalculated. In some isolated cases, it may be necessary to move up the start of OB6x. In this case, correct the calculated value manually. The value entered is understood to be in milliseconds.

Optimizing the Configuration

To assist you in optimizing the configuration, the "Isochronous Mode" dialog box provides an overview of all clocking-related parameters. To open this dialog box, go to HW Config and select the **Edit > Isochronous Mode** menu command.

The dialog box is divided into a hierarchy containing "PROFIBUS", "Slave" and "Module" display areas. When you select a master system in the "PROFIBUS" area, the "Slave" area will automatically display the associated slaves. Similarly, when you select a DP slave, the "Module" area will automatically display the associated modules. For a detailed description of the columns displayed in the dialog box, see the relevant online help section.

Creating a User Program

Create the required synchronous cycle interrupt OBs (i.e. OB 61).

At the start of the synchronous cycle interrupt OBs, the SFC 126 'SYNC_PI' must be called to update the process image partition for inputs, and at the end of OB 61, SFC 127 'SYNC_PO' must be called to update the process image partition for outputs. The process image partition to be used here is the one configured in the CPU ("Synchronous Cycle Interrupts" tab).

Notes

In particular, the following situation can arise with very short DP cycle times: The runtime of the user program (OB6x with SFC 126/127 called) is greater than the smallest cycle (see the technical data for the CPU, Section "Isochronous Mode"). In this case, you will have to manually increase the DP cycle time that was automatically calculated by STEP 7.

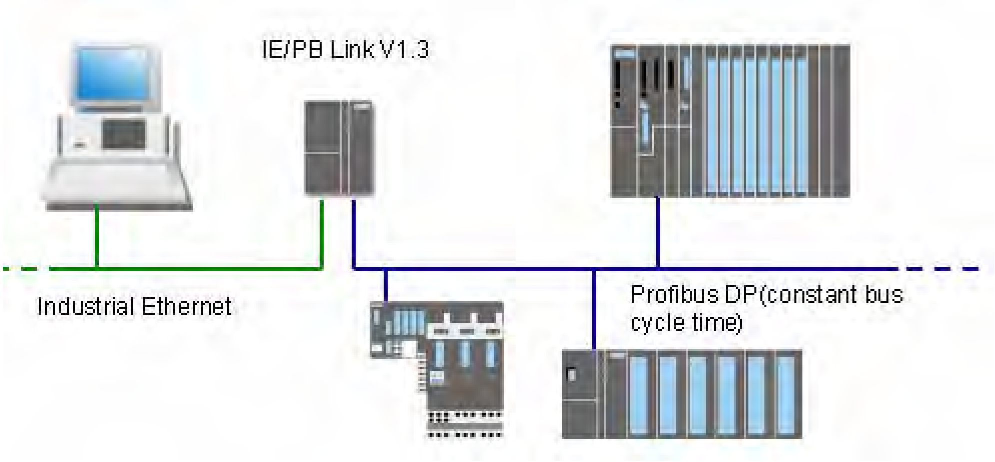
The runtime for individual OBs can be determined for different periods of time with SFC 78 'OB_RT' (WinAC RTX only).

To prevent inconsistent data being returned for OB6x, you should avoid using SFC14/15 (direct data access) in the isochronous mode OB.

3.14.3 Connecting a PG/PC to a Constant-Cycle PROFIBUS network via Industrial Ethernet and an IE/PB Link

The IE/PB link (Version 1.3) can be connected with the DP interface to a constant-cycle PROFIBUS-DP.

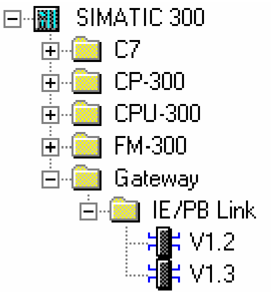
This configuration allows you to use the programming device/PC connected to the Industrial Ethernet to access stations on the constant-cycle PROFIBUS-DP (routing).



Configuring the IE/PB Link as the S7 Router

To configure the IE/PB link, proceed as follows:

1. Create a station of type SIMATIC 300.
2. Using drag & drop, add the IE/PB-Link (V1.3) to the station.



3. When adding the link, you will have to edit dialog boxes as follows:
   - To set the properties for the Industrial Ethernet interface, and
   - To set the properties for the PROFIBUS interface.
   After the IE/PB link has been added, it is in "DP master" operating mode.
4. Double-click the "PROFIBUS/DP" line for the IE/PB link.
5. Select the "Operating mode" tab.
6. Select the "No DP" option.
   In this operating mode, the IE/PB link at the PROFIBUS behaves like a programming device/PC.

3.14.4 Shortening the Process Reaction Time by Overlapping Ti and To

If, in your configuration, you select DP slaves that allow overlapping of Ti and To, you can shorten the DP cycle and so further reduce process reaction time.

For example, the IM 153-2 (as of 6ES7 153-2BAx1) supports overlapping of Ti and To.

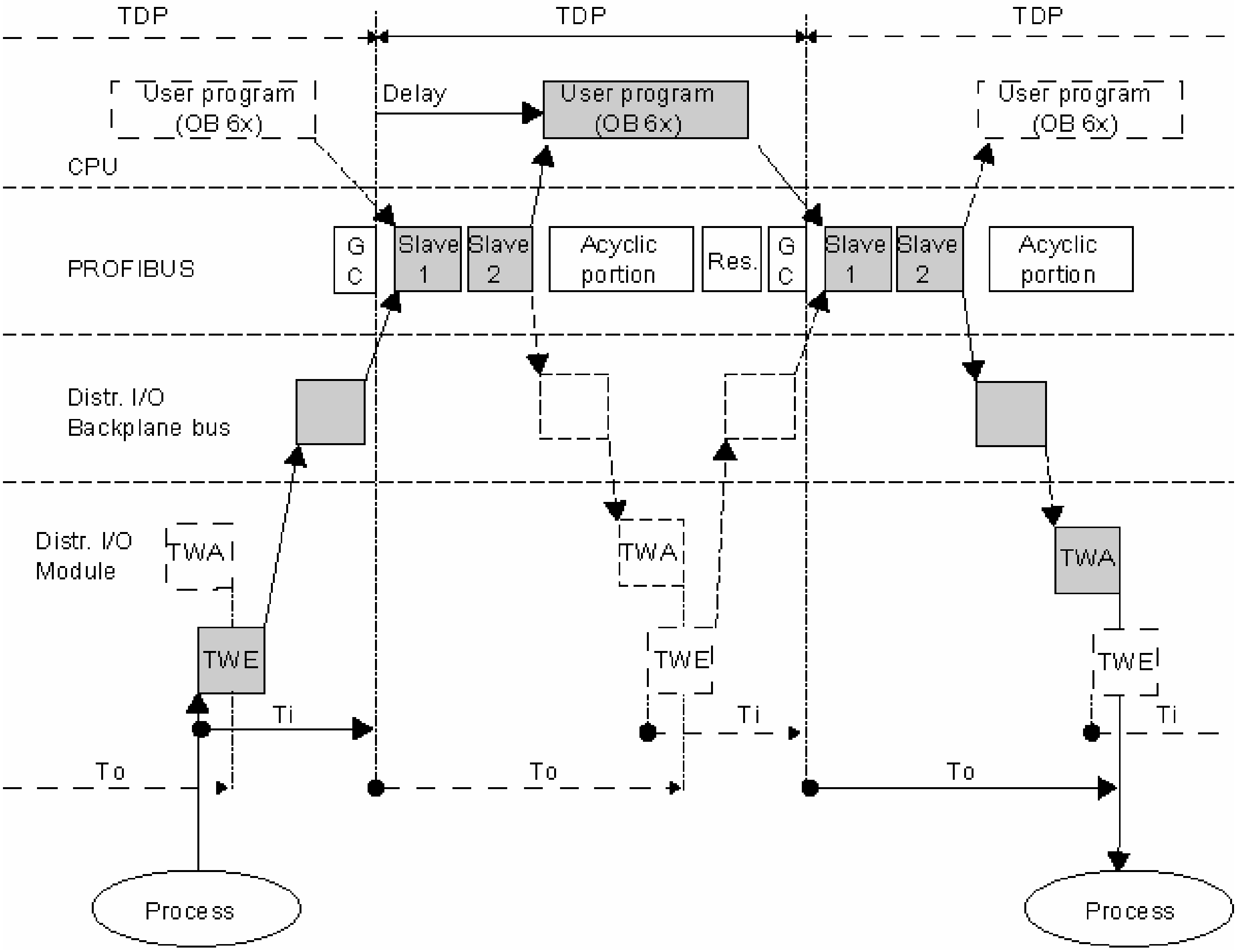
This has no influence on the procedure, since STEP 7 automatically determines the times and, based on the configuration selected, calculates the shortest possible DP cycle time.

Please note the following for the configuration:

- In the "Constant Bus Cycle Time" tab, uncheck the option "Times Ti and To same for all slaves" and set these times for each slave.
- If you want to operate modules that use both inputs and outputs in isochronous mode, overlapping Ti and To times is not possible.

Overlapping Processing

The principle for overlapping Ti and To functions is as follows: The peripheral input module is already reading the inputs while the peripheral output module is sending the process reaction from the user program to the outputs.



4 Configuring PROFINET IO Devices

4.1 What You Should Know about PROFINET IO

4.1.1 What is PROFINET IO?

PROFINET is the Ethernet based automation standard of the German PROFIBUS User Organization [PROFIBUS Nutzerorganisation e.V. (PNO)] in Karlsruhe, Germany. It defines a communications, automation and engineering model that applies to all manufacturers.

Objectives

The objectives of PROFINET are:

- Consistent, uniform communication over field bus and Ethernet
- Open, distributed automation
- Use of open standards

Architecture

The PROFIBUS User Organization (PROFIBUS International) specifies the following characteristics for PROFINET architecture:

- Communication between controllers as components in distributed systems
- Communication between field devices such as peripheral (I/O) devices and drives

Implementation by Siemens

The requirement for "communication between controllers as components in distributed systems" is implemented by means of "Component-based Automation" (CbA). With "Component based Automation", you create a distributed automation solution based on prefabricated components and parts solutions. You can use SIMATIC iMap as a configuration tool.

The requirement for "communication between field devices" is implemented by Siemens with "PROFINET IO". As with PROFIBUS DP, STEP 7 can be used to fully configure and program the components involved.

The following sections discuss configuring the communication between field devices with PROFINET IO.

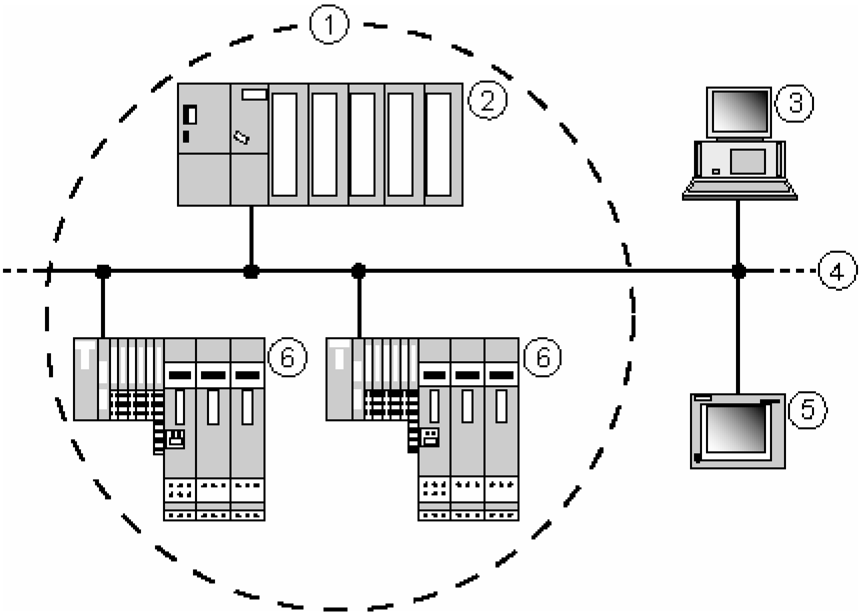
4.1.2 PROFIBUS DP and PROFINET IO: Similarities and Differences

Compatibility and continuity – and, in doing so, protecting system investments – characterizes the further development of field technology from PROFIBUS DP to PROFINET IO.
The following section contains an introduction to the new concepts and terminology. It concludes with information on the similarities and differences between PROFIBUS DP and PROFINET IO.

For more detailed information, please refer to the brochure "From PROFIBUS DP to PROFINET IO".

Comparison of the designations used with PROFIBUS DP and PROFINET IO

The following graphic shows the general designations of the most important devices used for PROFIBUS and PROFINET. The table after the graphic lists the names of the individual components used in PROFINET and PROFIBUS.



No.	PROFINET	PROFIBUS	Comment
(1)	IO system	DP master system	All IO devices (DP slaves) assigned to an IO controller (DP master)
(2)	IO controller	DP master	Controller in which the user program runs
(3)	IO supervisor (PG/PC)	PG/PC	Initial start up, HMI and diagnostics
(4)	Industrial Ethernet	PROFIBUS	Subnet type
(5)	HMI	HMI	Device for operator control and monitoring
(6)	IO device	DP slave	Distributed field device assigned to a controller (i.e. remote IO, valve terminal, frequency converter)

Note: PG = "Programmiergerät" = "programming device"

Similarities and Differences

The following table contains key words and phases pertaining to the fundamental characteristics of field bus systems along with explanations of the similarities and differences between PROFIBUS DP and PROFINET IO from the standpoint of PROFINET IO.

Function	Explanation
Real-time communication	Deterministic update times that can be calculated by STEP 7 based on the hardware configuration. In PROFINET IO, STEP 7 calculates the shortest update time resulting from the hardware configuration. You can change this setting manually. In contrast to PROFIBUS DP, the update time can be set individually for each IO device in PROFINET IO. PROFINET IO is based on a different communication procedure compared with PROFIBUS DP and you do not need to worry about profiles and bus parameters.
Integration of field devices	Done through installation of GSD files for both PROFIBUS DP and PROFINET IO. With PROFINET IO, the GSD files are in XML format, but the files themselves are handled in the same manner as for PROFIBUS DP.
Configuration	PROFINET IO is configured similarly to a DP master system. The only difference has to do with address assignment (due to Ethernet specifications). For detailed information on address assignment, refer to the section in which this topic is discussed.
Slot model	PROFINET IO is based on the slot model of PROFIBUS DP (DPV1): PROFINET interface modules plug into slot "0" of the IO device; the modules or submodules with user data start with slot "1".
Download or download to programming device (PG)	No difference in configuration between PROFINET IO and PROFIBUS DP
Diagnosis	Same diagnostic paths as for PROFIBUS DP (e.g. via station online, via accessible nodes) and options (i.e. module status). Diagnostic extent similar to that for PROFIBUS DP (only the structure of the diagnostic data is somewhat different; only channel diagnosis is possible). The structure of the diagnostic data records is, as with PROFIBUS DP, documented at the field devices (IO devices).
Blocks for the S7 user program and the system status lists (SSL)	Due to the larger data volumes for PROFINET IO, the system function blocks and the standard function blocks have to be adapted or re-implemented. Similar to the situation for the blocks, the system status lists (SSL) were also adapted. The new blocks and SSLs are also available for PROFIBUS DP. The list of the affected blocks and SSLs is in the *From PROFIBUS DP to PROFINET IO* programming manual.

Note: PG = "Programmiergerät" = "programming device"

4.1.3 Assigning Addresses and Names for PROFINET IO Devices

IP addresses

All PROFINET devices use the TCP/IP protocol; for this reason, they need an IP address when operated on the Ethernet.

To simplify configuration, you will only be prompted once to assign an IP address: when configuring the IO controller in HW Config.

At this point, STEP 7 displays a dialog for selecting the IP address and the Ethernet subnet. If the network is an isolated one, you can accept the default IP address and subnet mask assigned by STEP 7. If the network is part of an existing Ethernet company network, then you will have to ask you network administrator for this data.

The IP addresses of IO devices are generated by STEP 7 and normally assigned to the IO devices at the time the CPU is started up. The IP addresses of IO devices always have the same subnet mask as the IO controller and are assigned, starting from the IP address of the IO controller, in ascending order.

Device names

An IO device must have a device name before an IO controller can address it. PROFINET uses this method because names are easier to use and recall than complex IP addresses.

Assigning a device name for a specific IO device is comparable to setting the PROFIBUS address for a DP slave.

In their original, delivered condition, IO devices have no device names. IO devices can be addressed by an IO controller only after having been assigned a name by a programming device (PG)/PC, such as for transmitting configuration data (including the IP address) at startup or for the exchanging user data in cyclic operation.

To assign the configured device name to an IO device online, select the menu command **PLC > Ethernet > Assign Device Name**.

You can also assign a device name with the menu command **PLC > Ethernet > Edit Ethernet Node**; however, the device name should be set in the Properties dialog of an IO device so that it is stored in the configuration data.

Devices on an Ethernet subnet must have unique names.

If an IO controller in another station is going to be operated at the same time as the IO device (e.g. CP 1616), then in the project the name assigned to this IO device must be the same as the name assigned to the IO controller on the hardware side. This is the only case in which two nodes will have the same device names on the configured Ethernet subnet.

The device names must satisfy DNS conventions:

- Names are limited to a total of 127 characters (letters, numbers, dashes or periods).
- Any component part (that is, a character string between two dots) of the device name may only be up to 63 characters long.
- Names cannot contain any special characters such as umlauts, parentheses, underscores, forward or backward slashes, empty spaces, etc. The dash is the only special character allowed.

- The device name must not begin or end with the "-" or ".".
- The device name must not begin with numbers.
- The device name must not have the format n.n.n.n (n = 0...999).
- The device name must not begin with the character string "port-xyz-" (x , y, z = 0...9).
- Device names are assigned to the PROFINET IO devices in the commissioning phase.
- If several devices of the same type are connected to the same PROFINET IO system, STEP 7 automatically adds a consecutive number to the name from the GSD file. The second device has the suffix "-1" added, the third device the suffix "-2" etc.
- STEP 7 allows you to add the IO system to the device name automatically. If you require this, select the "Use name in device/controller" option in the properties of the IO system. The device name has the format:
- [Name from GSD file].[Name of IO system]

Structured device names

You can also structure the device name according to DNS conventions. To help you in structuring the names, use a period (".") as shown.

...<Subdomain Name>.<Domain Name>.<Top Level Domain Name>

STEP 7 supports you here by providing a prompt-driven dialog for using the name of the IO system in the device name:

<Name of specific device>.<Name of IO system>

You can set the name of the IO system at central location in the property dialog of the IO system.

When you copy an IO device to another IO system, STEP automatically applies the name of the IO system into which the device was inserted.

Device number

In addition to the device name, STEP 7 assigns a device number to the device when it is inserted. These numbers start with "1".

These device numbers are used to identify IO devices (e.g. SFC 71 "LOG_GEO") in the user program. Unlike the device name, the device number is not visible in the user program.

Special features

PROFINET IO: The maximum number of nodes in a PROFINET IO network is restricted to 512.

4.1.4 Use Different Method to Obtain IP Address / Device Name

Special applications such as mass produced machines or print feeders require new methods for obtaining the device names and IP addresses. The background here is the desire for a machine that can be commissioned locally at the customer's facility without needing STEP 7 or other tools.

Options for assigning IP addresses and device names

Apart from the known address and device name assignment in the "General" and "Parameters" tabs of the PN IO interface, there are three other ways in which the IP address and name can be obtained:

- Assignment by the user program.
- Assignment using the menu command: **PLC > Ethernet > Edit Ethernet Node**.
- Assignment when downloading the configuration to the target system in the "Select Node Address" dialog box.

Note

If neither the device name nor the IP address exist in the project, the user can select the target module with the IP address and/or the MAC address using **PLC > Display Accessible Nodes**.

Procedure

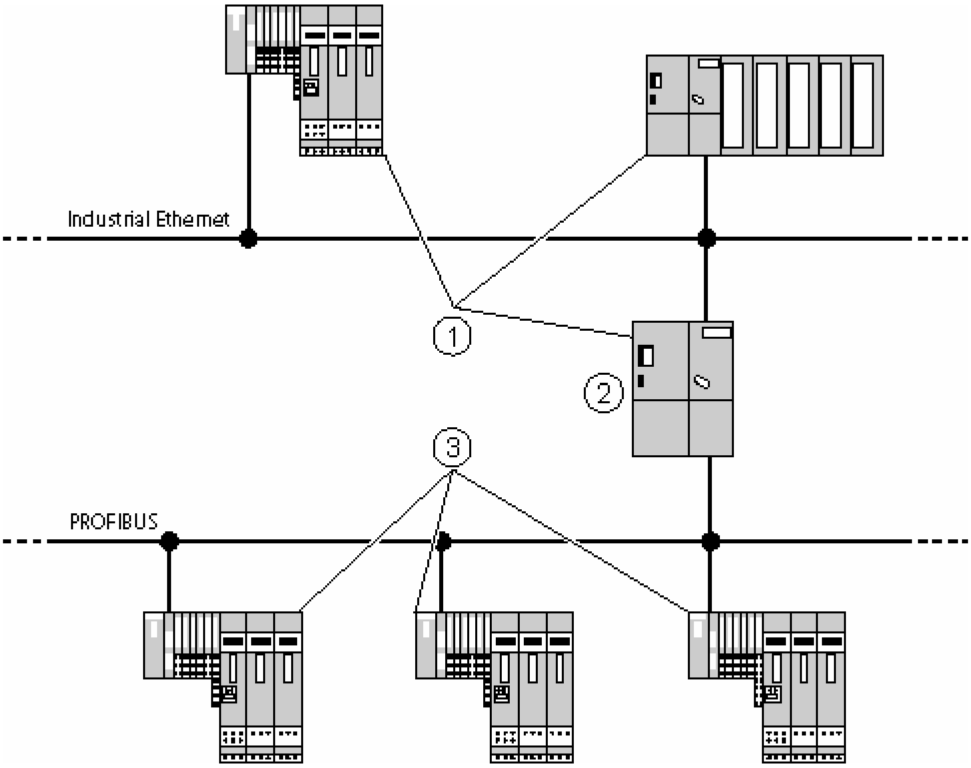
- Device name: Select the "Use different method to obtain device name" check box on the interface of the PROFINET device.
- IP address: Select the "Use different method to obtain IP address" check box on the "Parameters" tab of the Properties - Ethernet Interface dialog.

4.1.5 Integrating Existing PROFIBUS DP Configurations

Connecting PROFINET and PROFIBUS

PROFIBUS devices can be connected to the local PROFIBUS interface for a PROFINET device. In this way, you can integrate already existing PROFIBUS configurations into PROFINET.

The following illustration shows the network types supported for PROFINET: Industrial Ethernet and PROFIBUS.



No.	Description
(1)	PROFINET device
(2)	PROFINET device with proxy functionality (e.g. IE/PB link)
(3)	PROFIBUS devices

PROFINET devices with proxy functionality = Generic

The PROFINET device with proxy functionality is the substitute equivalent of a PROFIBUS device on the Ethernet. The proxy functionality allows a PROFIBUS device to communicate not only with its master but also with all nodes on PROFINET.

With this proxy functionality, there are no problems in integrating existing PROFIBUS systems into PROFINET communication.

For example, if you connect a PROFIBUS device to PROFINET by means of an IE/PB link, the IE/PB link establishes the communication via PROFINET as a substitute for the PROFIBUS-components.

4.1.6 Update Times For Cyclical Data Exchange

Because of the fully different kind of communication process used with PROFINET IO, you do not have deal with a series of bus parameters, unlike with PROFIBUS DP.

However, just like with PROFIBUS DP, STEP 7 automatically determines a time period within which a PROFINET IO device has exchanged its user data with the associated IO controller: this is known as the "update time".

Based on the hardware configuration in use and the resulting amount of cyclic data, STEP 7 automatically calculates an optimized update time for IO device in IRT mode. You can change these update times and set them individually for each IO device.

If you consider a single IO device in the PROFINET IO system, the IO controller provides this IO device with new data (outputs) within this update time, and the IO device has sent its latest data (inputs) to the IO controller within this time.

Properties of the update time

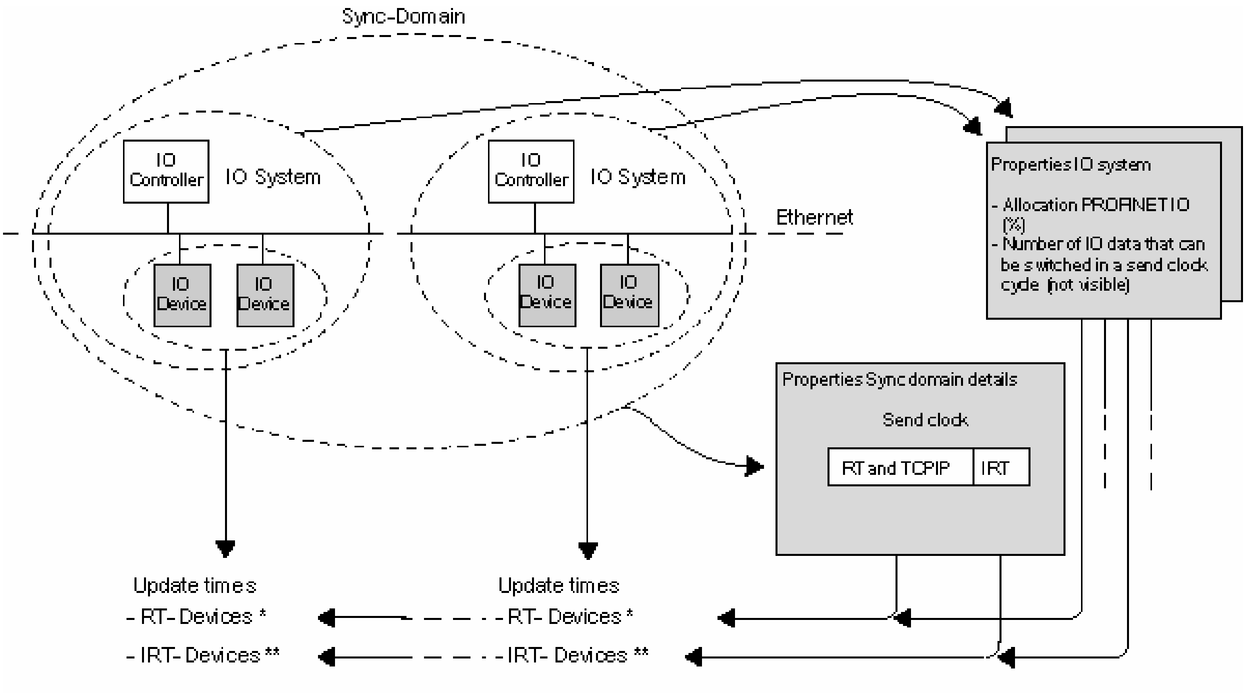
The update times can only be set in certain fixed intervals (timebases). STEP 7 calculates which values are possible from the properties of the IO devices involved (in other words, from the GSD files) and properties of the IO controller.

If you change the hardware configuration, such as by adding new IO devices, this may change the update time. If this is the case, the next time you open the dialog or during the consistency check, a message will inform you that a change has been made.

What factors influence the update time?

- The number of IO devices
- The number of inputs and outputs
- The percentage of time reserved for PROFINET IO
- The properties (performance) of the IO controller
- Properties of the relevant IO device
- The IRT time (if isochronous real time communication was configured)
- Send cycle of the IO controller or the sync master

The following schematic shows the factors that decide the update times. A distinction must be made between IO devices with IRT communication and IO devices with RT communication.



*	**RT:** If one of the displayed restrictions means that not all the IO data to be transferred could be transferred in one send cycle, the send clock is then automatically reduced, in other words, the data is transmitted over several send cycles. This increases the update time for individual devices.
**	**IRT:** Transfer of all the IO data during the update time calculated by the system is guaranteed. If you change the configuration later, the update time is **not** adapted automatically. Even a change to the "Communication allocation (PROFINET IO)" setting does not lead to automatic adaptation of the update time. A message is displayed during the consistency check if the set update time is not adequate. You will then need to adapt the update time in the "IO Cycle" tab of the Properties dialog of the interface or in the Properties dialog of the PROFINET IO system.

Reading and setting the update time from a central location

To open the dialog to read and change the update time, proceed as follows:

1. Double-click the Profinet IO system (the "railroad track").
2. Select the "Update Time" tab.
   - If, in addition to PROFINET IO, other cyclical services (i.e. from Component-based Automation, CBA) have to be taken into consideration:
     Go to the "Communication Component (PROFINET IO)" field and select a percentage that should be reserved for PROFINET IO.
     If at least one IO device is configured, then the default setting is 100%.
     If no IO device is configured, then the default setting is 0%.
   - The transmit clock is also displayed in this tab. If there is at least one synchronized PROFINET device in the IO system, the transmit clock is set by the sync master of the sync domain and cannot be changed here. If you want to change it, you will need to open the sync domain dialog (**Edit > PROFINET IO > Domain Management** menu command)
3. If required, you can increase the update time for one or more IO devices (multiple selection also possible).

Reading and setting the update time at the IO device

To open the dialog to read and change the update time for specific IO device, proceed as follows:

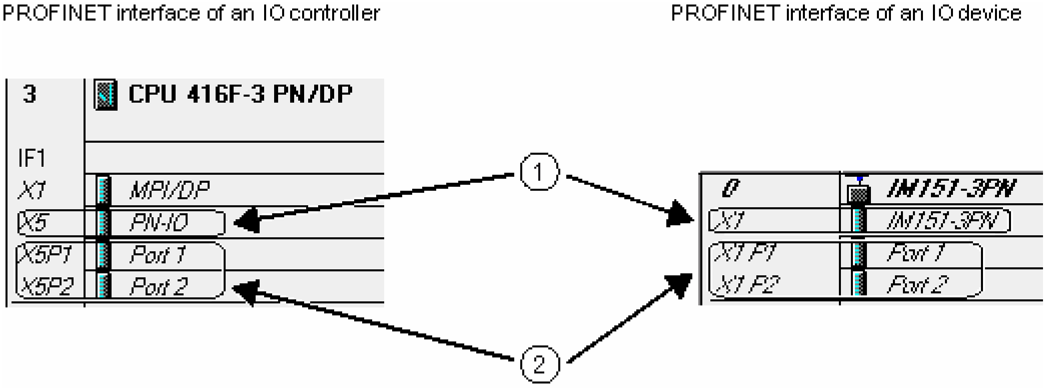
1. Double-click the IO device.
2. Select the "IO Cycle" tab.
3. Check the current setting for the update time. If necessary, change it and then confirm the settings with "OK".

Optimizing the update times of individual IO devices

The update times calculated by STEP 7 are already optimized, in other words, they are as short as possible for IO devices in RT or IRT mode. You can, however, extend the leeway for shorter update times of individual IO devices by increasing the update times of other IO devices that supply data that is not time critical.

4.1.7 Representation of the PROFINET Interface in HW Config

PROFINET devices with an integrated switch are shown in the configuration table of HW Config as follows:


PROFINET interface of an IO controller
PROFINET interface of an IO device
1
2

3	CPU 416F-3 PN/DP
IF1	
X1	MPI/DP
X5	PN-IO
X5P1	Port 1
X5P2	Port 2

0	IM151-3PN
X1	IM151-3PN
X1 P1	Port 1
X1 P2	Port 2

(1) This row represents the "Interface" of a PROFINET interface.

(2) These rows represent the "Ports" of a PROFINET interface.

Interface

In the "Properties" dialog of this row, you can access not only the name and diagnostic address, but also the following functions if they are supported by the component:

- Synchronization settings for PROFINET devices that support IRT communication, for example setting of the synchronization role and RT class.
  With IO controllers, you can also choose whether or not communication diagnostics leads to a diagnostic interrupt (OB82 call).
- PROFINET settings such as send clock, communication allocations for PROFINET IO and CBA for IO controllers.
- Time-of-day synchronization for IO controllers.
- Settings for isochronous mode applications for IO devices.
- Watchdog for IO devices.

Port

In the "Properties" dialog of these rows, you can access not only the name and diagnostic address, but also the following functions if they are supported by the component:

- Transmission medium
- Topology settings
- Settings for the end of domains (sync domains and time domains); known as boundaries.
  These settings set the boundaries for sync and time frames.

Diagnostics capabilities of the PROFINET interface

The interface and the ports also trigger diagnostic events that can be read out with system diagnostics (HW Config ONLINE) or using data records.

Examples:

- The interface can detect loss of synchronization.
- The port can diagnose a link down, for example, when the cable between the ports is interrupted.

Apart from the ability to recognize errors and faults, the PROFINET interface is also capable of reporting certain information relating to preventive maintenance (maintenance demand and requirement messages).

4.2 Steps For Configuring a PROFInet IO System

4.2.1 Overview: From Configuration To Cyclical Data Exchange

The following sections describe the basic steps required to place a PROFINET IO system in operation. It tells you how to configure PROFINET components, assign addresses and how the system starts.

Configuring an IO system in HW Config

1. Assign parameters to an IO controller (i.e. CPU 317-2 PN/DP) in HW Config. While assigning the interface parameters, assign an Ethernet subnet and an IP address.
2. Arrange all required IO devices in the IO system ("railroad track").
3. For each IO device: check the device name, and, if necessary, change it and set its parameters.

Assign addresses

1. Each IO device must have the configured device name assigned to it.
2. Download the hardware configuration while the CPU is in STOP mode.
   The configured IP address is automatically assigned to the PN interface (e.g. a CPU) when the hardware configuration is downloaded.

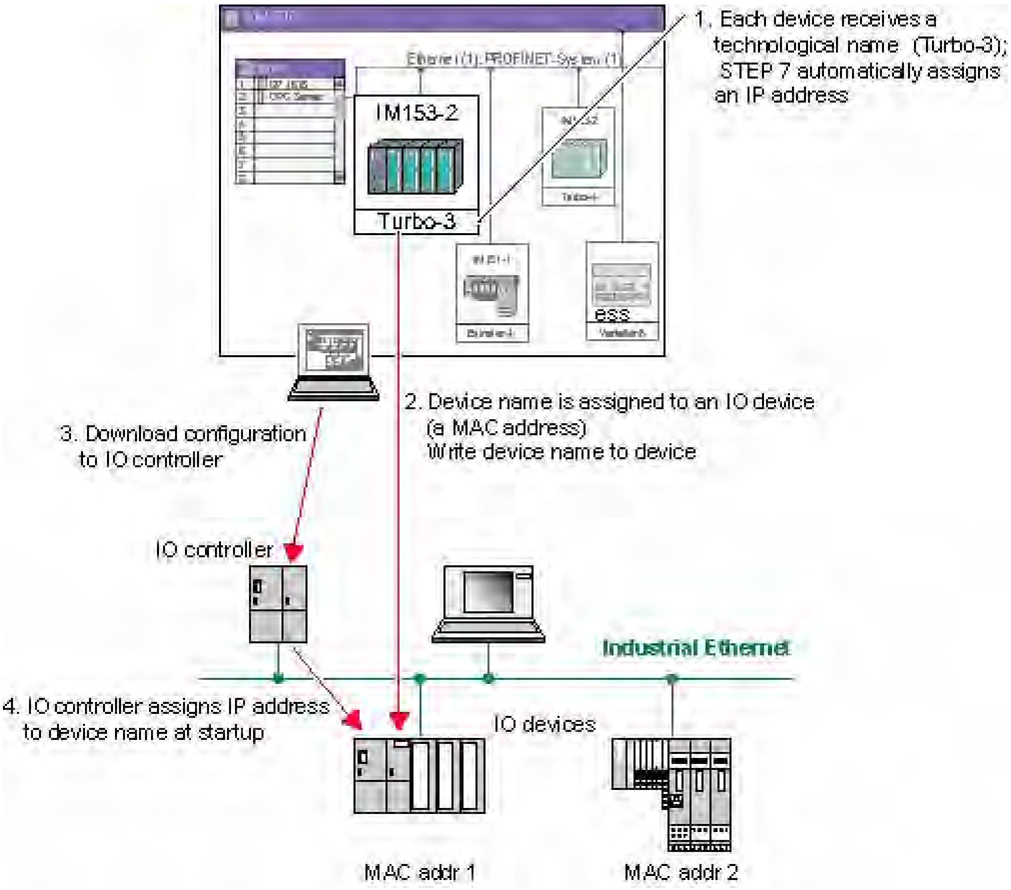
Startup

At startup, the CPU transfers the downloaded configuration via the PN interface to the IO devices. As with PROFIBUS DP, the "'Finished' message by modules" and "Transfer of parameters to modules" monitoring times that are assigned as parameters apply here as well.

At startup, the CPU can identify the IO devices by their device names and automatically transfer the associated IP addresses.

After the addresses and parameters have been successfully transferred within the monitoring times, the PROFINET devices start their cyclical data exchange.

If the addresses and parameters were not successfully transferred, the CPU goes into STOP or RUN mode, depending on the setting of the "Startup when expected/actual config. differ" parameter.


Ethernet(1): PROFINET-System (1)
IM153-2
Turbo-3
1. Each device receives a technological name (Turbo-3); STEP 7 automatically assigns an IP address
2. Device name is assigned to an IO device (a MAC address) Write device name to device
3. Download configuration to IO controller
IO controller
Industrial Ethernet
4. IO controller assigns IP address to device name at startup
IO devices
MAC addr 1
MAC addr 2

4.2.2 Basic Procedure for Configuring a PROFINET IO System

If you know how to configure a PROFIBUS DP system, then you already know how to configure a PROFINET IO system – the procedure is largely the same.

Station window as an image of the real IO system

When you have placed an IO controller (e.g. a CPU 317-2 PN/DP) in position, STEP automatically draws a line that represents the IO system. You then place the IO devices to be assigned to this IO controller in position by using drag-and-drop to move them there from the "Hardware Catalog" window in "PROFINET IO".

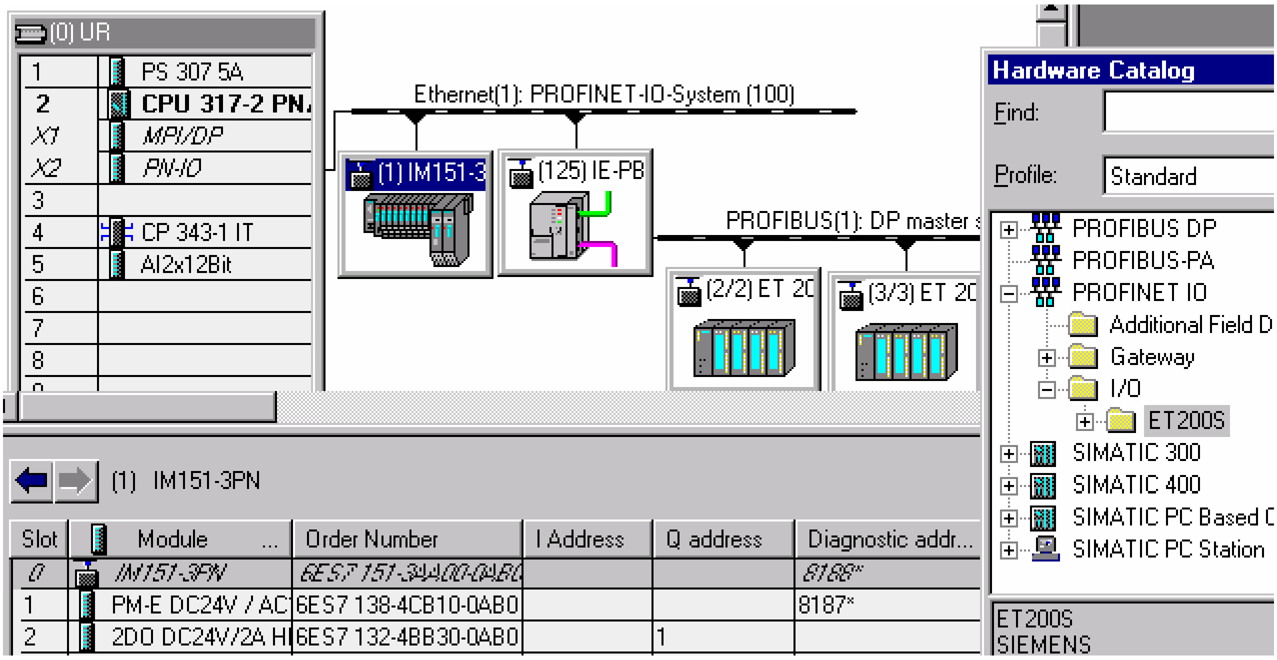
When an IO device is put in position, a name is automatically assigned to it (the default name is the name in the GSD file).

STEP 7 also automatically assigns an IP address. Starting with the IP address of the IO controller, STEP 7 searches for the next available IP address. However, this IP address is not relevant when handling the objects when configuring the system. Nevertheless, it is necessary because all nodes on Ethernet with TCP/IP protocol must have an IP address.

In addition, STEP 7 assigns device number that you can use to work with the device in the user program (e.g. with SFC 71 "LOG_GEO"). This number is also displayed in the symbol for the IO device.

The IO device does not appear in the "Hardware Catalog" window

If an IO device does not appear in the "Hardware Catalog" window, then you need to install the corresponding GSD file after STEP 7 has been started (menu command: **Options > Install GSD File**). Follow the dialog prompts to install the new GSD file. The newly installed IO device then appears in the "Hardware Catalog" window under "PROFIBUS IO", either under the name of the device family (for Siemens devices) or under "Additional Field Devices."



1	PS 307 5A
2	CPU 317-2 PN/
X1	MPI/DP
X2	PN-IO
3	
4	CP 343-1 IT
5	AI2x12Bit
6	
7	
8	

Find:

Profile: Standard

(1) IM151-3PN

Slot	Module ...	Order Number	I Address	Q address	Diagnostic addr...
0	IM151-3PN	6ES7 151-3AA00-0AB(			8188*
1	PM-E DC24V / AC	6ES7 138-4CB10-0AB0			8187*
2	2DO DC24V/2A HI	6ES7 132-4BB30-0AB0		1	

ET200S
SIEMENS

Configuring the IO device in the detailed view

When you select an IO device, then the configuration (slots with modules/submodules) and I/O addresses are displayed in the detailed view of the station window.

4.2.3 Creating a PROFINET IO System

Requirement

You have arranged a module rack in the station window, and it is now opened (slots in the rack are visible).

PROFINET IO Controller

You can use the following devices as an IO controller:

- A CPU with an integral or insertable PROFINET interface (integral, such as CPU 317-2 PN/DP)
- A CP along with a CPU (e.g. CP 443-1 Advanced connected to a suitable S7-400-CPU)
- A PC station (e.g. with CP 1612)

Procedure

1. In the "Hardware Catalog" window, select an IO controller (e.g. CPU 317-2 PN/DP).
2. Drag-and-drop the module to a permissible row in the module rack. The "Properties – Ethernet Node" dialog box then opens.
   Here you can do the following:
   - Create a new Ethernet subnet or select an existing one
   - Set the properties for the Ethernet subnet (such as the name)
   - Set the IP address of the IO controller
3. Confirm the settings with "OK"

   The following symbol appears for CPUs with an integral IO controller:

   This symbol is the "peg board" for the IO devices in the IO system.

External IO Controllers

CPs that can be used as external IO controllers support numerous options for communications. As such, they are not "predetermined" for use as IO controllers.

If you use an external IO controller (e.g. CP 443-1 Advanced), then you have to insert an IO system after inserting the IO controller (context menu command: "Insert PROFINET IO System").

Device name

The device name for the IO controller can be set in the "General" table of the Properties dialog of the interface (normally, the default device name is "PN-IO").

4.2.4 Selecting and Arranging IO Devices

The procedure for selecting and arranging IO devices is essentially the same as the one for PROFIBUS DP.

Requirement

An IO system must be available and visible in the station window.

Procedure

1. As with PROFIBUS DP, you will find the IO devices (correspond to the slaves used with PROFIBUS DP) in their own catalog section called "PROFINET IO".
   Open the folder desired under "PROFINET IO".
2. Drag-and-drop the IO devices to position them or double-click an IO system.
3. If you are dealing with a modular IO device, then insert the required modules or submodules into the IO device.

The IO devices are represented with symbols in the station window, much like the slaves on PROFIBUS. A symbol contains the device number (possibly abbreviated) and the device name.

Configuring IO devices

IO devices have property sheets in which you can change the address information (device number and device name) automatically assigned by STEP 7 when the device is inserted as well as the diagnostic address of the IO device.

In such a property sheet you can start the dialog for changing the interface and the subnet properties. To do this, click the "Ethernet" button in the property sheet. The dialog box that then appears contains the IP address, which you can then change.

Depending on the IO device, you can select a check box to disable the assignment of the IP address by the IO controller.

Parameters that apply to the entire IO device can be set in this dialog box.
The properties of an IO device are determined by its associated GSD file.

Navigation options

To locate IO devices quickly, you can select the **Edit > Go To > Ethernet Node** menu command. The dialog box that is then displayed lists all IO systems and the IO devices connected to them. For IO systems, the associated subnet is shown. For IO devices, the designation (device name), device number and the IP address are shown.

Assigning device names

You can change the default device names and the device number in the "General" tab of the IO device Properties dialog. Double-click on the icon of the IO device in the station window.

The configured device names must then be assigned to the IO devices online. To do this, select the menu command **PLC > Ethernet > Assign Device Name**.

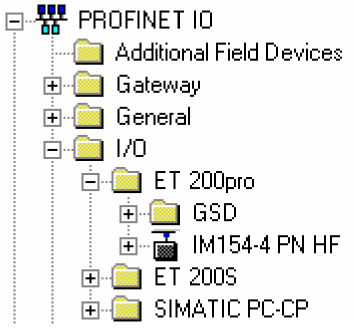
4.2.5 Where Are The IO Devices Located In The Hardware Catalog?

All IO devices are shown in the Hardware Catalog under "PROFINET IO".

The structure of the Hardware Catalog is determined only by the GSD files. If several GSD files for an IO device family are available in STEP 7 data management (e.g. caused by importing different file versions), then the last (that is, the most recent) version is always used to display the device properties.

IO devices made by Siemens appear directly in "PROFINET IO" below the folder with the name for the respective device family (e.g. ET 200S).

If other GSD files are available for the device family, they are also displayed in the Hardware Catalog under their family name in a "GSD" subfolder. The non-GSD variants offer easy and convenient options for configuring (such as easier packing of addresses, etc.).



Other IO devices installed with a GSD are displayed under "Additional Field Devices".

In the case of modular IO devices, the insertable modules/submodules are located below the IO device interface modules.

"Migration" folder

The current PROFINET IO devices (IO controller and IO devices) are devices with expanded PROFINET diagnostics. This includes functions such as diagnostics and parameter assignment of the integrated PROFINET interface, for example, fiber-optic diagnostics and topology configuration. These functions are accessible in the configuration table via additional submodules: interface and ports.

If you configure an IO controller that does not support extended PROFINET diagnostics, you cannot assign any IO devices with interface and port submodules.

For this situation, you will find an IO device of the same type in the "Migration" folder without extended PROFINET diagnostics - in other words, without interface and port submodules - that can be operated with this IO controller.

You will find detailed information on the Customer Support pages under entry ID 23678970.

4.2.6 Handling PROFINET IO Systems

Designations for IO Systems

The designations of IO systems are structured similarly to those used for DP master systems:

The first part of the name designates the Ethernet subnet. This part is followed by a colon (":").

The second part of the name consists of the designation "IO system", followed by the number of the IO system in parentheses. While the numbering for DP master systems starts with "1", STEP 7 starts numbering IO systems with "100".

Changing the settings of an IO system

In addition to the name and number, the changeable properties of an IO system also include the update time.

To change the settings, proceed as follows:

1. Double-click the IO system ("railroad track")
2. Select the appropriate tab
   (the "General" tab to change the name and the IO system number,
   or the "Update Time" tab to read or change the update time for the IO system).

Disconnecting an IO system

As with a DP master system, you can also disconnect an IO system. To do so, select the **Edit > PROFINET IO > PROFINET IO System > Disconnect** menu command.

The IO system then remains as an "orphaned" IO system and is visible in the station.

If the IO system has no IO devices on it, the IO system is deleted.

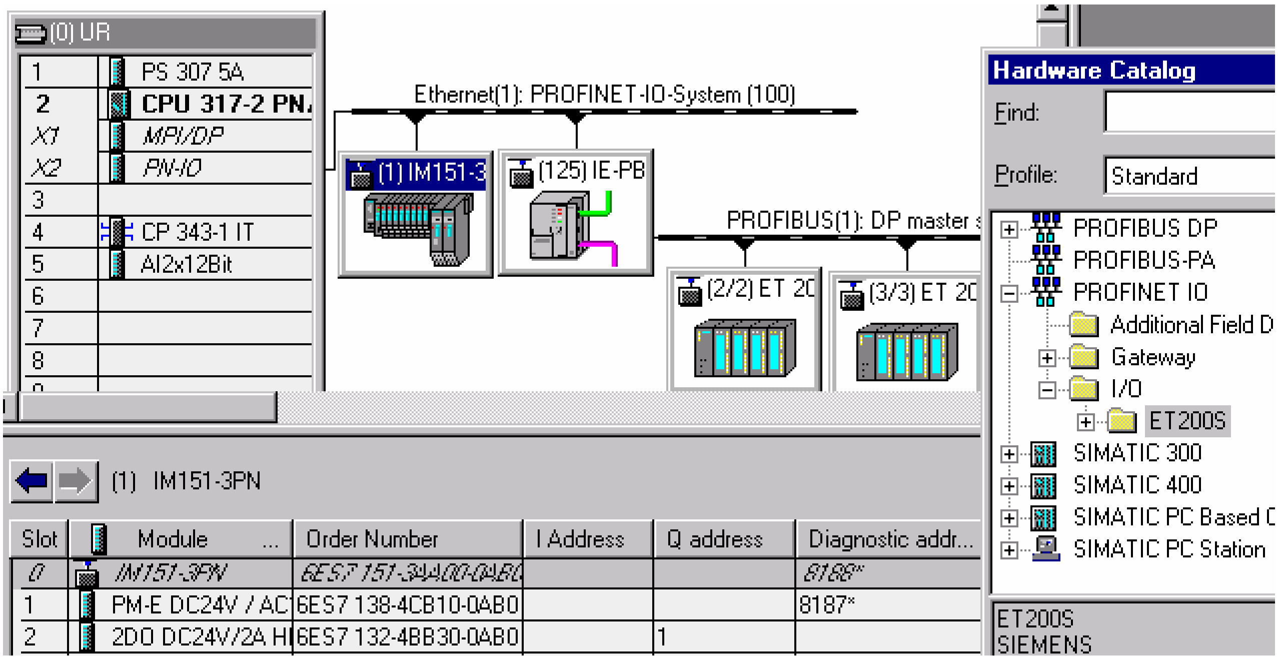
Inserting an IO system

If you have configured one or more IO systems and then disconnected them from the PROFINET interface, you can insert one of the "orphaned" IO systems back into the selected PROFINET interface. To do so, select the **Edit > PROFINET IO > PROFINET IO System > Insert** menu command.

4.3 Examples of Configurations with PROFINET IO

4.3.1 Configuration with an Integrated IO Controller

Whenever a CPU is inserted into an integrated IO controller (e.g. CPU 317-2 PN/DP), STEP 7 automatically creates an IO system. You can then drag-and-drop the desired IO devices from the Hardware Catalog to this PROFINET IO system.



(0) UR

1		PS 307 5A
2		**CPU 317-2 PN/**
X1		*MPI/DP*
X2		*PN-IO*
3		
4		CP 343-1 IT
5		AI2x12Bit
6		
7		
8		

(1) IM151-3PN

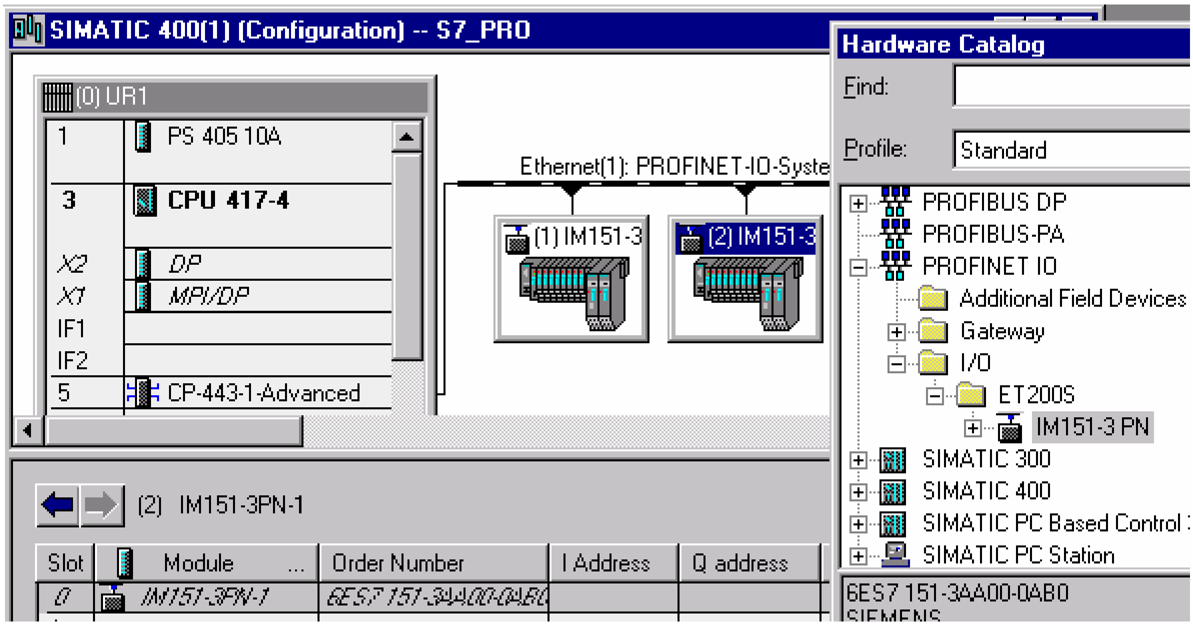
Slot		Module ...	Order Number	I Address	Q address	Diagnostic addr...
0		*IM151-3PN*	*6ES7 151-3AA00-0AB(*			*8188**
1		PM-E DC24V / AC	6ES7 138-4CB10-0AB0			8187*
2		2DO DC24V/2A HI	6ES7 132-4BB30-0AB0		1	

4.3.2 Configuration with an External IO Controller

CPs that can be used as external IO controllers support numerous communications options. For this reason, they are not "predetermined" to be used as IO controllers.

If you use an external IO controller (e.g. CP 443-1 Advanced), then after inserting the controller, you must insert an IO system (context menu command: "Insert PROFINET IO System").

You can then drag-and-drop the desired IO devices from the Hardware Catalog to this PROFINET IO system.



SIMATIC 400(1) (Configuration) -- S7_PRO

(0) UR1

1	PS 405 10A
3	CPU 417-4
X2	DP
X1	MPI/DP
IF1	
IF2	
5	CP-443-1-Advanced

Profile: Standard

(2) IM151-3PN-1

Slot		Module ...	Order Number	I Address	Q address
0		IM151-3PN-1	6ES7 151-3AA00-0ABC		

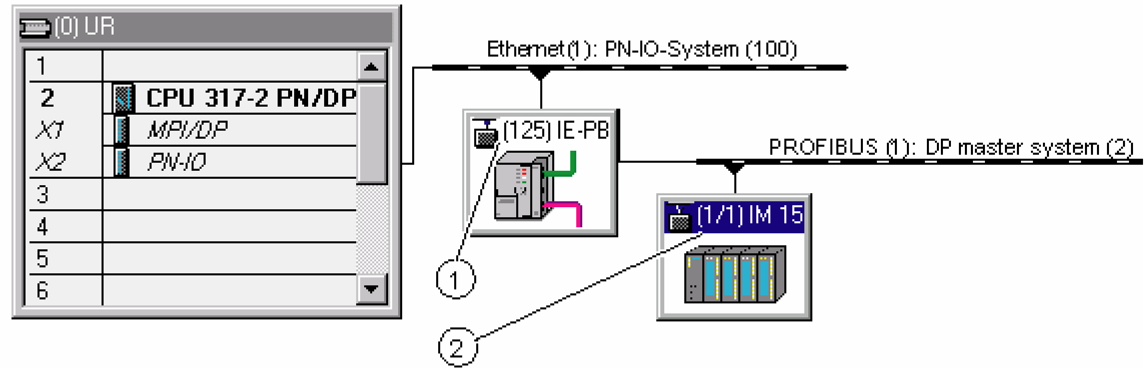
6ES7 151-3AA00-0AB0
SIEMENS

4.3.3 Configuration with IE/PB Link

You can use the IE/PB link to connect PROFIBUS DP configurations to PROFINET IO.

From the standpoint of the CPU, PROFIBUS DP slaves are on the same network as the IE/PB link. These slaves have the same device names and IP addresses as the IE/PB link, but do have different device numbers. In addition, they also still have specific PROFIBUS addresses.

In addition to the device number, the symbol for the IE/PB link also shows the PROFIBUS address, since this device has two address schemas.



(1) Device number of the IE/PB link

(2) Device number and PROFIBUS address of the DP slave

Handling device numbers and PROFIBUS addresses at the master system

When IO devices are positioned, STEP 7 assigns the same number for the device number and the PROFIBUS address.

To get an overview of the device numbers and the PROFIBUS addresses being used, double-click the symbol for the IE/PB link and select the "Device Numbers" tab.

Each line in the "Device Numbers" tab represents a DP slave. If you select a line and then click the "Change" button or double-click a line, a dialog box is displayed in which you can change the device number.

You can change the PROFIBUS address in the usual manner:

1. Double-click the symbol for the DP slave to display its properties dialog.
2. In the "General" tab, click the "PROFIBUS" button.
3. In the next dialog box, change the PROFIBUS address.

Restrictions

The configuration described above applies to DP slaves on the PROFIBUS subnet of an IE/PB link, subject to the following restrictions:

- No IE/PB link can be inserted
- No DP/PA link can be inserted
- No Y-link can be inserted
- No CiR compatibility
- No redundant slaves can be inserted
- No isochronous mode/constant bus cycle time cane be configured

4.3.4 Configuration with SIMATIC PC Stations

A "PC station" is a PC with communications modules and software components within an automation solution with SIMATIC.

By using appropriate communications modules and software components you can operate a PC stations as a PROFINET IO controller.

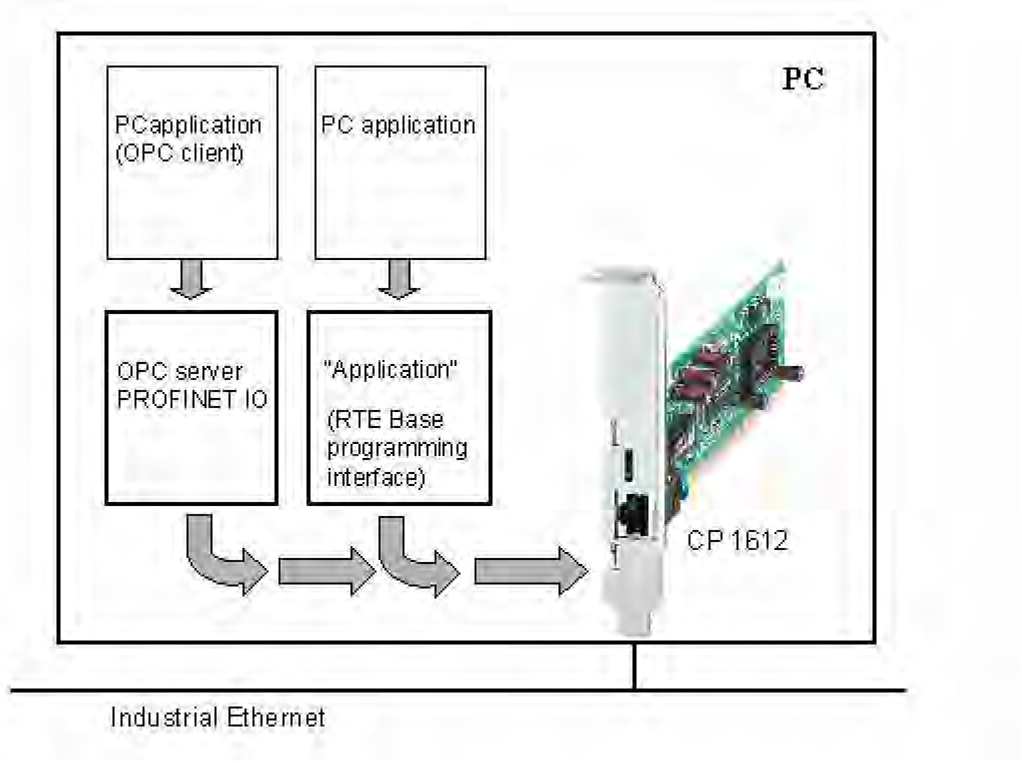
Your PC applications in the PC station can access the PROFINET IO controller in the following ways:

- As an OPC client over the PROFINET IO OPC server
- Directly over the PROFINET IO user interface (RTE base programming interface)

Only one of these access options can be used by PC applications at any one time (open/close sequence).

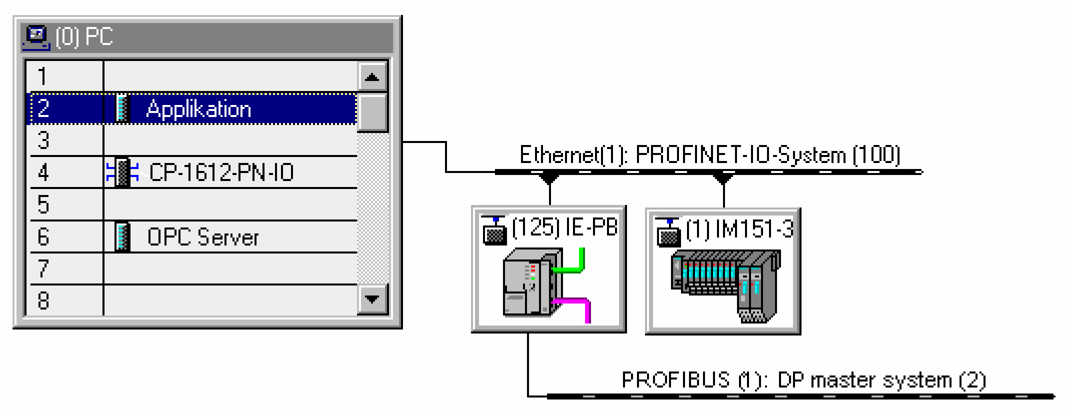
Functions	PROFINET IO OPC Server	RTE Base Programming Interface
Read and write IO data	Yes	Yes
Read and write data records	Yes	Yes
Receive and acknowledge interrupts	No	Yes

The following illustration shows a PC station with the components described. The lower illustration shows the corresponding configuration of the station in HW Config.


PC
PCapplication (OPC client)
PC application
OPC server PROFINET IO
"Application" (RTE Base programming interface)
CP 1612
Industrial Ethernet

Configuring in HW Config

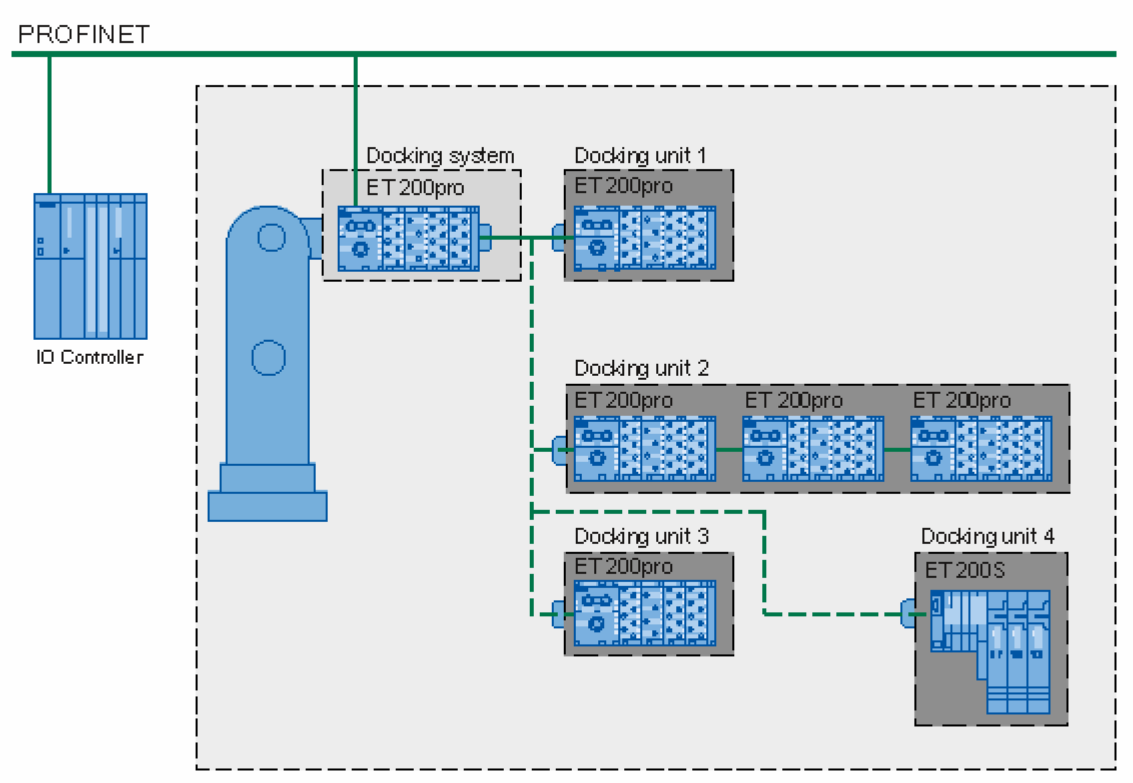
As with other types of stations, the IO devices to be addressed are assigned to the IO system of the IO controller (here CP 1612).



4.3.5 Configuring Docking Systems

Different docking units are often used alternately on robots and other machines.

Example of a welding robot:



The **docking system** corresponds to the alternating partner port of an IO device or switch. The docking units are connected alternately with the docking system or the "alternating partner port".

On a welding robot, **docking units** often correspond to the various welding tongs that the robots change and dock automatically. Depending on the application, other welding tongs may also be needed. The robot fits the required tongs as necessary. The docking units consist of one or more IO devices.

Configuration:

To configure multiple docking units for a docking system, follow the steps below:

1. Configure your hardware as usual; initially, handle the IO devices representing the alternating partners as you would normal IO devices.
2. Open the properties of the port you want to configure as an "alternating partner port".
3. As the partner port in the "Topology" tab, select "Alternating partner port".
4. Click the "Add" button and select a port for each device that will be part of the docking system in the "Add Alternating Partner" window. Confirm with "OK".
5. Close the properties dialog with "OK"

You can also perform steps 4. and 5. graphically by dragging and dropping. To do this, use the Topology Editor:

6. Right-click on the **PROFINET IO system** and then on **PROFINET IO Topology** in the menu. In the Topology Editor, select the "Graphic View" tab.
7. First click on the required **partner port** and hold down the left mouse button. The move the mouse pointer to the **alternating partner port** and release the mouse button. The interconnection is indicated by a broken line.

The startup time of the IO devices is particularly important when using docking systems; refer to the section: Tips for Achieving the Fastest Startup Times

---

Note

- A docking unit can also consist of several IO devices connected in series; this is known as the line depth and depends on the number of devices. To be able to implement docking units consisting of more than one IO device, you will need to set the partner ports used for the IO device being used (configure the topology).
- You can obtain a clearer view of your port interconnections in the graphic vie of the Topology Editor.

4.4 Commissioning PROFINET IO Devices

4.4.1 Accessing PROFINET IO Devices Online via Ethernet

Requirements

You have set up the programming device (PG)/PC interface to work with a TCP/IP interface card, which means that the Ethernet interface of your programming device (PG)/PC can be accessed from STEP 7.

Be sure to observe the guidelines for connecting your programming device (PG)/PC to the CPU. For CPU 317-2 PN/DP, both devices (CPU and PG/PC) must be connected to one switch.

It may be necessary to adjust the properties of the Ethernet interface in the Control Panel of the programming device (PG) (the properties of the Internet Protocol (TCP/IP)). If, in addition to the programming device (PG), you have only connected S7-CPUs and PROFINET IO devices, you must not enable the "Obtain an IP address automatically" option in Windows.

Online access as with MPI or PROFIBUS

The commands in the "PLC" menu used to establish online access to IO devices can be used in all applications that provide this menu, such as in HW Config, in the SIMATIC Manager and in NetPro.

In general, you have the same options via Ethernet (PROFINET) as you do with other online paths (MPI or PROFIBUS).

Exceptions

Special functions not available over PROFINET:

- The menus available through **PLC > PROFIBUS** only apply to PROFIBUS
- Prepare line diagnostics (only suitable for diagnostic repeaters on PROFIBUS DP)

4.4.2 Displaying Ethernet Nodes in the "Accessible Nodes" Window

Requirements

You have set up the programming device (PG)/PC interface to work with a TCP/IP interface card, which means that the Ethernet interface of your programming device (PG)/PC can be accessed from STEP 7.

For an online connection via the "Accessible Nodes" window, the programming device (PG)/PC and "Accessible Nodes" must be connected to the same physical Ethernet subnet.

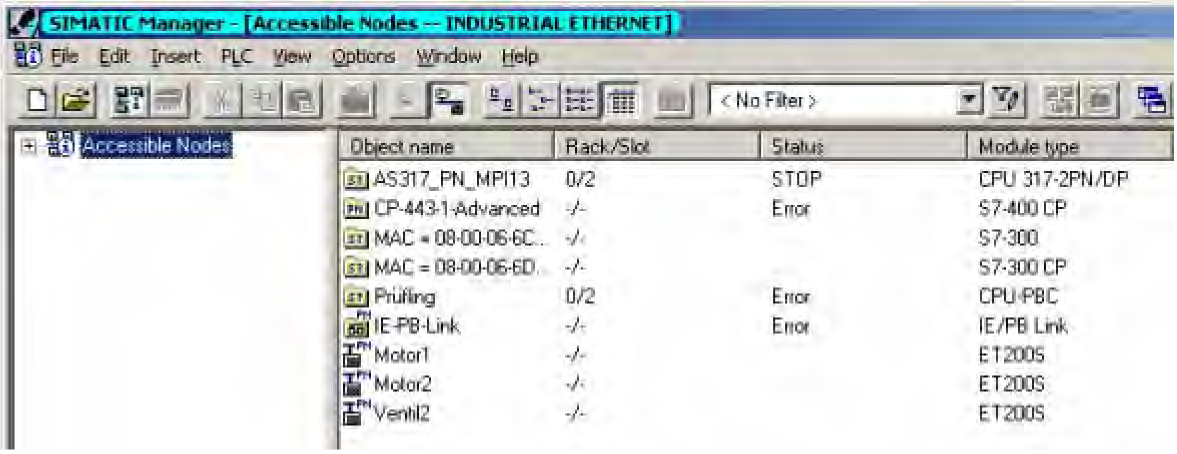
If nodes are only accessible via intermediate switches or routers (with the use of a protocol), they will not be displayed in the list of "Accessible Nodes".

Nodes displayed

The following components are shown in the "Accessible Nodes" window:

- S7-CPs
- S7-CPUs
- SIMATIC PC stations
- SIMATIC network components (PROFINET devices)
- All PROFINET devices (in general, all devices that support the DCP protocol)
- IE/PB link with its subordinate DP slaves
- FMs
  In contrast to FMs of the S7-300, FMs of the S7-400 (such as the FM 456) are not shown in the list of "accessible nodes". If, for example, you want to run a memory reset on an FM 456, please do this in the online view of the project where all configured FMs are listed (both those of the S7-300 as well as those of the S7-400). As an alternative, you can, of course, reset the FM memory using the keyswitch.

Depending on the type of component, there may be more less information displayed in the detailed view of the "Accessible Nodes" window.



Object name	Rack/Slot	Status	Module type
AS317_PN_MPI13	0/2	STOP	CPU 317-2PN/DP
CP-443-1-Advanced	-/-	Error	S7-400 CP
MAC = 08-00-06-6C...	-/-		S7-300
MAC = 08-00-06-6D...	-/-		S7-300 CP
Prüfling	0/2	Error	CPU-PBC
IE-PB-Link	-/-	Error	IE/PB Link
Motor1	-/-		ET200S
Motor2	-/-		ET200S
Ventil2	-/-		ET200S

The following table shows what information is displayed in the "Object Name" column:

Accessible Nodes is...	Displayed in the "Object Name" column	Explanation
PROFINET IO device with assigned device name	Device name	IP address and MAC address can be determined in a properties dialog. The "Details" view also shows the device type as well as status information (e.g. IP addresses inadvertently assigned twice ("doubled-up"), errors, etc.).
PROFINET IO device without assigned device name	IP address	As a substitute (for an unassigned IP address) the MAC address is displayed.
S7 CP	Name (of the station)	Only if an IP address was assigned to the CP; otherwise, MAC address. The "Details" view also shows the operating mode and the module type.
S7 CPU, PC stations	Name (of the station)	The "Details" view also shows the operating mode and the module type, and, if available, information from the associated STEP 7 project (station name, CPU name, plant designation).
IE/PB link	See PROFINET device	Special issue: if you select the IE/PB link in the "Accessible Nodes" list, the detailed list then displays the connected DP slaves along with their PROFIBUS addresses as well as other information.
Other nodes (no S7 nodes or devices with DCP protocol)	----	Cannot be displayed

Calling diagnostic functions

You can perform system diagnostics from the "Accessible Nodes" window. To do so, proceed as follows:

1. Select the component to be diagnosed.
2. Select the appropriate menu command (menu: **PLC > Diagnostics/Settings > ...**).

As of STEP 7 V5.3 Service Pack 2, the "Module Information" diagnostic function can also be used for DP slaves "behind" the IE/PB link; that is, for those devices that are not directly connected to the Ethernet.

4.4.3 Downloading to PROFINET IO Devices

Downloading to the CPU serving as IO controller for the first time in HW Config

After you have configured the whole IO system, you now have to download this configuration to the CPU. In this way, the IP address is assigned to the CPU (or more exactly: the IO controller).

Before starting the download, you should perform a consistency check in order to have the configuration checked for addresses that have been used twice and names that are not unique.

Find out if the CPU to receive the download can be downloaded to for the first time via the PROFINET interface. If not, you will then have to first download the hardware configuration via the MPI interface.

The following description of the procedure presumes that there is a connection existing between the programming device (PG) and the CPU via Ethernet.

1. Select the **PLC > Download** menu command.
2. Select the modules to be downloaded to.
3. If necessary, in the "Select Station Address" dialog box, click the "Display" button to display the modules that are actually accessible (including the CPU to receive the download with its current IP address or its MAC address if an IP address is still not available).
4. In the accessible modules, select the CPU to receive the download.
   This module is then displayed in the "Enter connection to target station" field.
5. Start the download by clicking the "OK" button.
   During this process, the CPU (i.e. the IO controller) also gets its configured IP address assigned to it.

Downloading to the CPU serving as IO controller for the first time in the SIMATIC Manager or NetPro

As an alternative method, you can download to the station (with the CPU of the IO controller) in the SIMATIC Manager or NetPro (menu command: **PLC > Download**).

If you trigger the download in the SIMATIC Manager, the configuration must have already been compiled (in HW Config or NetPro).

If you download in the SIMATIC Manager or in NetPro, the "Select Station Address" dialog box is also displayed since the configured IP address no longer matches the actual IP address. In this case, proceed as described in the previous section in order to reach the modules to be downloaded.

Prerequisite

The PG/PC interface must be set on a TCP/IP interface card; in other words, the Ethernet interface of the PG/PC can be reached from STEP 7.

Assigning a device name to an IO device (online)

In order for the configured CPU to be able to address the PROFINET IO devices, you must assign the configured device name to each individual IO device.

In order to use the following procedure, the IO devices must be accessible online for the programming device (PG)/PC on the Ethernet.

Procedure

1. In HW Config or in NetPro, select the **PLC > Ethernet > Assign Device Name** menu command
2. In the "Assign Device Name" dialog box, go to the "Device Name" field and select a device. Then, in the "Available Devices" field, select the device to receive the selected device name.
3. To make the device easier to identify, click the "Flashing" button to cause a display LED in the selected device to flash.
4. Click the "Assign Name" button.
5. Repeat steps 2 through 4 until you have assigned a name to all the devices.

Starting the CPU

After the names are all assigned, you can put the CPU into RUN mode.

At startup, the CPU distributes the configuration information to the IO devices and then goes into cyclical operation.

4.4.4 Assigning Device Names via a Memory Card (MMC)

As of STEP 7 V5.3 SP2, you can also configure the device names of PROFINET IO devices offline.

To do this, you save a configured device name to an MMC and then insert the MMC into the appropriate IO device.

If a device becomes defective and an IO device has to be completely exchanged, the IO controller automatically assigns the parameters to the new device and configures it. With an MMC card, a device can be exchanged without using a programming device (PG).

Requirements

- The programming device (PG)/PC has a prommer for MMCs.
- The IO device must support the "Assign the device name via MMC" function.
- The station with its PROFINET IO system is configured.

Procedure

1. Insert the MMC in the prommer.
2. In HW Config or in NetPro, select the IO device to be assigned the device name via MMC.
3. Select the PLC > Save Device Name to Memory Card menu command.
4. If the MMC has other data on it, a message will inform you of this condition so that you can delete the MMC before saving the device name.

4.4.5 Device Replacement without Exchangeable Medium

IO devices need to be replaced in automation systems more often. Normally, the IO devices are assigned a device name by inserting an exchangeable medium or from the PG. The CPU identifies the IO device using this device name.

If certain conditions are met, it is possible to replace an IO device without inserting an exchangeable medium (for example) and without requiring a PG. In this case, the neighbor relations between the individual IO devices and the IO controller are analyzed using Ethernet mechanisms. Based on these neighbor relations stored on the IO controller, the IO controller recognizes which IO device has been replaced and redefines the device name.

Configuration:

If you want to replace an IO device without inserting an exchangeable medium and without setting parameters from a PG, make sure that the "Device replacement without exchangeable medium" check box is selected for the IO controller interface in the "General" tab.

Note

- Device replacement without exchangeable medium / PG is possible only if the topology of the system has already been configured.
- If individual IO device of the automation system do not support device replacement without exchangeable medium, a message will be displayed for the IO device during the consistency check.
- Use only new IO devices as the replacement devices, or if you use devices that were already in use, make sure that you reset them to their original status as shipped.

4.4.6 Changing the IP-Address or the Device Name During Operation

Assigning an IP address or device name for a single node

Before a new IP address or a device name can be assigned, the following conditions must be met:

There is no data being exchanged with the IO controller.

If the SIMATIC Manager or the list of accessible nodes is open, you can assign each PROFINET-device an/another IP address another device name. To do this, select the **PLC > Edit Ethernet Node** menu command.

Changing IP addresses from a central location

If you have to check or change the IP addressees for several IO devices, proceed as follows:

1. In HW Config, select the PROFINET IO system (the "railroad track") and then select **Edit > PROFINET IO > PROFINET IO System > IP Addresses** menu command. Alternative method: you can also find this function by selecting an IO system and then right-clicking to display the context menu.
2. The dialog box that is then displayed shows all the IO devices with their current device names and IP addresses. You can change all the IP addresses from this central location.
3. Download the changed IP addresses (menu command: **PLC > Download to Module**).

When an IO controller or an IO device is selected, you can also call this function in NetPro by selecting the **Edit > PROFINET IO > PROFINET IO System > IP Addresses** menu command.

4.4.7 Use Different Method to Obtain IP Address / Device Name

Special applications such as mass produced machines or print feeders require new methods for obtaining the device names and IP addresses. The background here is the desire for a machine that can be commissioned locally at the customer's facility without needing STEP 7 or other tools.

Options for assigning IP addresses and device names

Apart from the known address and device name assignment in the "General" and "Parameters" tabs of the PN IO interface, there are three other ways in which the IP address and name can be obtained:

- Assignment by the user program.
- Assignment using the menu command: PLC > Ethernet > Edit Ethernet Node.
- Assignment when downloading the configuration to the target system in the "Select Node Address" dialog box.

Note

If neither the device name nor the IP address exist in the project, the user can select the target module with the IP address and/or the MAC address using **PLC > Display Accessible Nodes**.

Procedure

- Device name: Select the "Use different method to obtain device name" check box on the interface of the PROFINET device.
- IP address: Select the "Use different method to obtain IP address" check box on the "Parameters" tab of the Properties - Ethernet Interface dialog.

4.4.8 Diagnosing PROFINET IO Devices

The same diagnostic paths available to you in STEP 7 for PROFIBUS DP components are also available for PROFINET IO. The procedures used are identical.

When you click the **Station > Open Online** menu command in HW Config, you can then diagnose PROFINET IO devices in addition to S7 stations. This view also shows the slaves "behind" an IE/PB link.

However, the procedure for evaluating diagnostic information by means of SFBs/SFCs in the user program is somewhat different.

For PROFINET IO, there a consistent structure for data records with diagnostic information for all device manufacturers. Diagnostic information is only collected for faulty channels

The system status lists (SSLs), SFB 54 and SFB 52 have been expanded in order to also make the status of PROFINET IO systems and the diagnostic information available for an S7 user program.

- For example, to get a complete overview of the status of the PROFINET IO system, you can read SSL 0x0X91 (SFC 51).
- To read out diagnostic records directly from a faulty module, use SFB 52 (read data record) to get detailed error information on its current status.
- To read event-related (e.g. triggered by an error OB) diagnostic records, use SFB 54 in the corresponding error OB (read additional information on interrupt).

SFB 52 and SFB 54 can also continue to be used for PROFIBUS DP.

For information on which SSLs and which diagnostic records are defined for PROFINET IO and how diagnostic data records are structured, please refer to the programming manual *From PROFIBUS DP to PROFINET IO*.

4.4.9 Configuring the Watchdog Time

You can configure a watchdog time for PROFINET IO devices.

If the IO device does not receive IO data from the IO controller within the configured watchdog time, the device goes into safe mode.

You do not enter the watchdog time directly, but as a "Number of accepted update cycles with missing IO data". This makes it easier to make the setting because the update time can be shorter or longer, depending on the performance of the IO device or the setting.

STEP 7 uses the "Number of accepted update cycles without IO data" to calculate the resulting watchdog time.

Procedure

Set the update time for an individual IO device as follows:

1. Double-click an IO device.
2. Select the "IO Cycle" tab.
3. Check the set watchdog time and change it, if necessary. To do this, change the factor for "Number of accepted update cycles without IO data". Acknowledge the settings with "OK".

You should only change the default setting in special cases, such as during commissioning.

4.5 Media Redundancy

4.5.1 Options for Media Redundancy

To increase the availability of an Industrial Ethernet network with optical or electrical bus topologies, there are several options available:

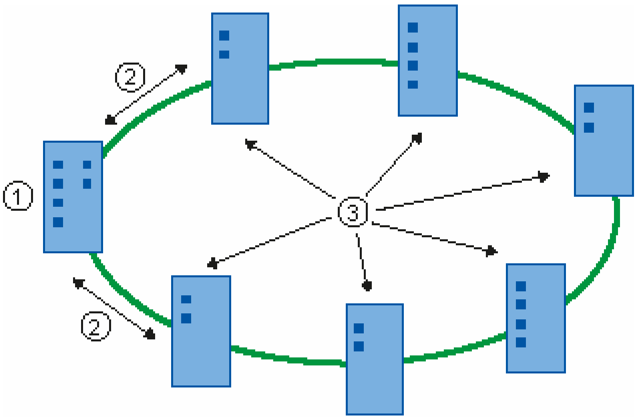
- Mesh networks
- Parallel connection of transmission paths
- Closing a linear bus topology to form a ring topology

4.5.1.1 Media Redundancy in Ring Topologies

Structure of a ring topology

Nodes in a ring topology can be IO devices, IO controllers, external switches and/or the integrated switched of communications modules.

To set up a ring topology with media redundancy, the two free ends of a linear network topology must be brought together in one device. The closing of the linear bus topology to form a ring is achieved with two ports (ring ports) of a device in the ring. This device is the redundancy manager. All other devices in the ring are redundancy clients.



Legend			
1	Redundancy manager		
2	Test frames		
3	Redundancy clients		

The two ring ports of a device are the ports that establish the connection to its two neighboring devices in a ring topology. The ring ports are selected and set during the configuration of the device.

Before physically closing the ring, download the configuration of your STEP 7 project to the individual devices.

How media redundancy works in a ring topology

The data paths between the individual devices are automatically reconfigured if the ring is interrupted at any point. After reconfiguration, the devices are accessible again in the recreated topology.

If there is no interruption in the network, the 2 ring ports in the redundancy manager are disconnected from each other so that no data frames can circulate. In terms of data transmission, the ring topology appears to be a linear bus. The redundancy manager monitors the ring topology. It does this by sending test frames both from ring port 1 and ring port 2. The test frames run round the ring in both directions until they arrive at the other ring port of the redundancy manager.

An interruption in the ring can be caused by failure of the connection between two devices or by failure of a device in the ring.

When the test frames of the redundancy manager no longer reach the other ring port due to an interruption in the ring, the redundancy manager connects its two ring ports. With this substitute path, a functioning connection is established between all remaining devices in the form of a linear bus network topology.

The time between the ring being interrupted and the re-establishment of a functioning linear bus topology is known as the reconfiguration time.

As soon as the interruption is eliminated, the original transmission paths are re-established, the ring ports in the redundancy manager are disconnected from each other and the redundancy clients informed of the change. The redundancy clients then use the new paths to the other devices.

Methods of media redundancy

The standard method of media redundancy in SIMATIC is MRP (Media Redundancy Protocol) with a best case reconfiguration time of 20 ms. Up to 50 devices can participate per ring.

4.5.1.2 Media Redundancy Protocol (MRP)

The "MRP" method conforms to the Media Redundancy Protocol (MRP) specified in the standard IEC 61158 Type 10 "PROFINET".

The best case reconfiguration time after an interruption of the ring is 20 ms.

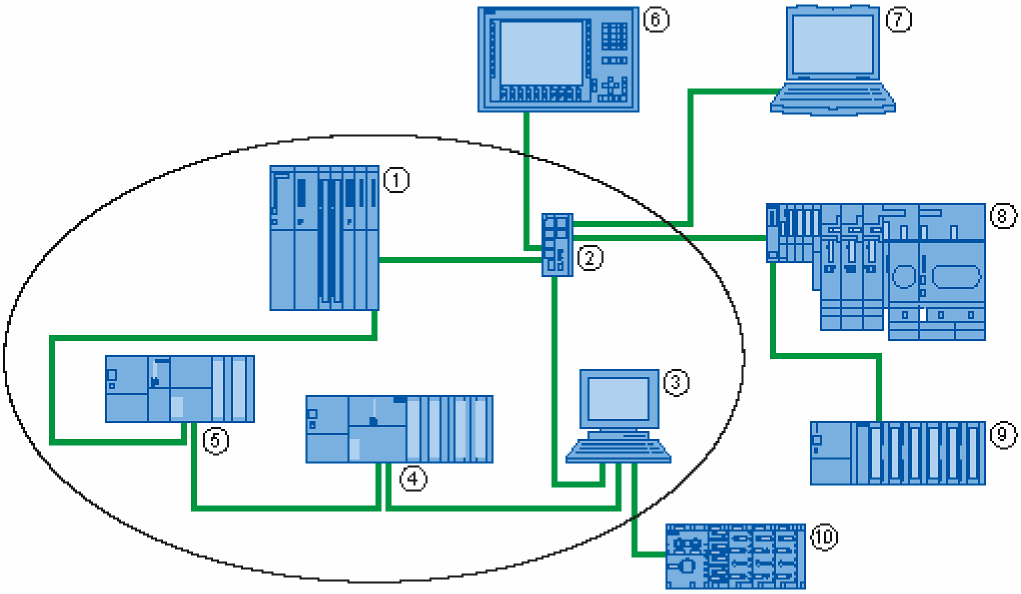
Prerequisites

The following requirements must be met for problem-free operation with the media redundancy protocol MRP:

- MRP is supported in ring topologies with up to 50 devices. Exceeding this number of devices can lead to a loss of data traffic.
- All devices must be interconnected via their ring ports.
- "MRP" must be enabled for all devices in the ring (see "Configuring Media Redundancy in PROFINET IO").
- The connection settings (transmission medium / duplex) must be full duplex and at least 100 Mbps for all ring ports. Otherwise, data traffic may fail.
  To do this, set all the ports involved in the ring to "Automatic settings" in the "Options" tab of the properties dialog during STEP 7 configuration.
- The ring in which you want to use MRP may only consist of devices that support this function.

Topology

The following schematic shows a possible topology for devices in a ring with MRP. The devices inside the white oval are in the redundancy domain.



Example of a ring topology with the media redundancy protocol MRP

Legend			
1	S7-400 with CP 443-1 Advanced	6	HMI station
2	SCALANCE X206-1 switch	7	PG/PC
3	PC with CP 1616	8	ET 200S
4	S7-300 with CP 343-1 Advanced	9	ET200M
5	S7-300 with CP 343-1 Advanced	10	ET 200pro

The following rules apply to a ring topology with media redundancy using MRP:

- All the devices connected within the ring topology are members of the same redundancy domain.
- One device in the ring is acting as redundancy manager.
- All other devices in the ring are redundancy clients.

Non MRP-compliant devices can, for example, be connected to the ring via a SCALANCE X switch of via a PC with a CP 1616.

Prioritized startup

If you configure MRP in a ring, you cannot use the "prioritized startup" function in PROFINET applications on the devices involved.

If you want to use the "prioritized startup" function, then disable MRP in the configuration.

In the STEP 7 configuration, properties dialog of the PROFINET interface > "Media Redundancy" tab > "MRP configuration" box, set the role to "Not node in the ring" in the "mrpdomain-1" domain.

4.5.1.3 Configuring Media Redundancy in PROFINET IO

Requirements

- The components involved must support the Media Redundancy Protocol (MRP).
- No IRT communication is configured.

Procedure

There are two ways of configuring media redundancy:

- Using domain management; here, MRP domains can also be managed.
- Via the PNIO interfaces of the relevant PROFINET devices

Media redundancy configuration using domain management:

1. Right-click on the PROFINET IO system and select "PROFINET IO Domain Management..." from the shortcut menu.
2. Go to the "MRP Domain" tab.
3. In the Nodes section, all the nodes in the project can be displayed sorted by stations and the use of media redundancy, selected and configured. If you click on them while holding down the Ctrl key, you can select several stations / devices and configure them together with the "Edit..." button.

You will find the options available in "Possible Settings" below.

Configuring media redundancy via the interfaces of the PROFINET devices involved:

Media redundancy must be configured for **all** PROFINET devices that operate with media redundancy:

1. Double-click on the PROFINET IO interface of the device you are configuring.
2. Go to the "Media Redundancy" tab and make the configuration settings.

You will find the options available in "Possible Settings" below.

Possible settings

Domain

- All devices configured in a ring with MRP must belong to the same redundancy domain. A device cannot belong to more than one redundancy domain. Select the same domain for all devices of your ring from the drop-down list (usually "mrpdomain-1").
- The MRP settings remain in effect following a restart of the device or following a power down and hot restart.

Role

- Depending on the device you are using, the roles "Manager", "Manager (Auto)", "Client" and "Not a node in the ring" are available.

Rules:

- A ring must have at least one device with the role of "Manager (Auto)".
- A ring may have exactly one device with the role of "Manager". No other devices with the "Manager" or "Manager (Auto)" role are permitted. All other devices can only have the role of "Client" or "Not a node in the ring".
- You want to disable MRP:
  Select the option "Not node in the ring" if you do not want to operate the device within a ring topology with MRP.

Ring port 1 / ring port 2

Here, select the port you want to configure as ring port 1 and ring port 2. The drop-down list shows the selection of possible ports for each device type. If the ports are specified in the factory, the boxes are grayed out.

Diagnostic interrupts

Select the "Diagnostic interrupts" option, if you want diagnostic interrupts relating to the MRP status to be output on the local CPU. The following diagnostic interrupts can be generated:

- Wiring or port error
  Diagnostic interrupts are generated if the following errors occur at the ring ports:
  - A neighbor of the ring port does not support MRP
  - A ring port is connected to a non-ring port
  - A ring port is connected to the ring port of another MRP domain.
- Interruption / return (redundancy manager only)
  If the ring is interrupted and when the original configuration returns, diagnostic interrupts are generated. The occurrence of both interrupts within 0.2 seconds indicates an interruption in the ring.

Notice

To ensure problem-free operation if you use a third-party device as the redundancy manager, all other devices in the ring must have the permanent role setting of "Client" before you close the ring. Otherwise you may encounter circulating frames that will bring down the network.

4.5.2 Shared Devices

4.5.2.1 What You Should Know about Configuring Shared Devices

Shared device functionality

In larger or widely distributed plants, numerous IO controllers are often used. In this situation, it is possible that sensors in close geographical proximity to each other need to send data to different IO controllers. Previously, this had to be handled by several IO devices assigned to different IO controllers. The shared device functionality allows the submodules of an IO device to be divided up among different IO controllers saving one or more head modules.

Prerequisites

To be able to use the shared device function, the IO controllers and the shared device must be in the same Ethernet subnet.

Access to the submodules of the shared device is then divided up among the individual IO controllers. Each submodule of the shared device can be assigned exclusively to an IO controller. The assignment of the individual submodules is made in HW Config.

Representation in HW Config

A shared device is shown more than once in HW Config. If, for example, a shared device is used by two IO controllers, the same device appears in two different stations in HW Config.

Instead of modules each with a submodule, only submodules are shown in HW Config to simplify the user view.

Basics of configuration

- Each IO controller assigns separate geographical addresses for the submodules of the shared device that are assigned to it. Logical addresses can be assigned to the submodules belonging to the controller as usual.
- A shared device must have the same IP parameters and the same device name in each station. During configuration, two situations must be distinguished:
  - Shared device in the same project: STEP 7 handles important consistency checks for the user. The checking of the correct assignment of the IP parameters and the monitoring of the correct access by the IO controller to the individual submodules is performed by STEP 7.
  - Shared device in different projects: The stations with the IO controllers that use the shared device are created in different projects. In each station, care must be taken that the shared device is configured identically in each station. Only one controller may ever have full access to a submodule. The IP parameters and device name must be identical. Inconsistencies in the configuration will cause failure of the shared device.

Types of access

Submodules can be assigned to one IO controller and one only.
Types of access and their significance:

- Full access. The IO controller has full access to the submodule.
  Rights of the IO controller:
  - Read and write access to input data, output data and data records
  - Assignment of logical addresses for submodules
  - Assignment of parameters for the submodule
  - Reception of interrupts from the submodule
- No access: The IO controller has no access to the submodule.
  For the IO controller, this means in particular that:
  - There is no data exchange with the submodule.
  - No interrupts can be received from the submodule.
  - Submodules cannot be assigned parameters.

4.5.2.2 Steps in Configuring a Shared Device in the same STEP 7 Project

Introduction

The following example describes the simplest configuration of a shared device: Two IO controllers share the submodules of an IO device. The two IO controllers are in the same STEP 7 project with the advantage that the consistency check is made automatically.

Procedure

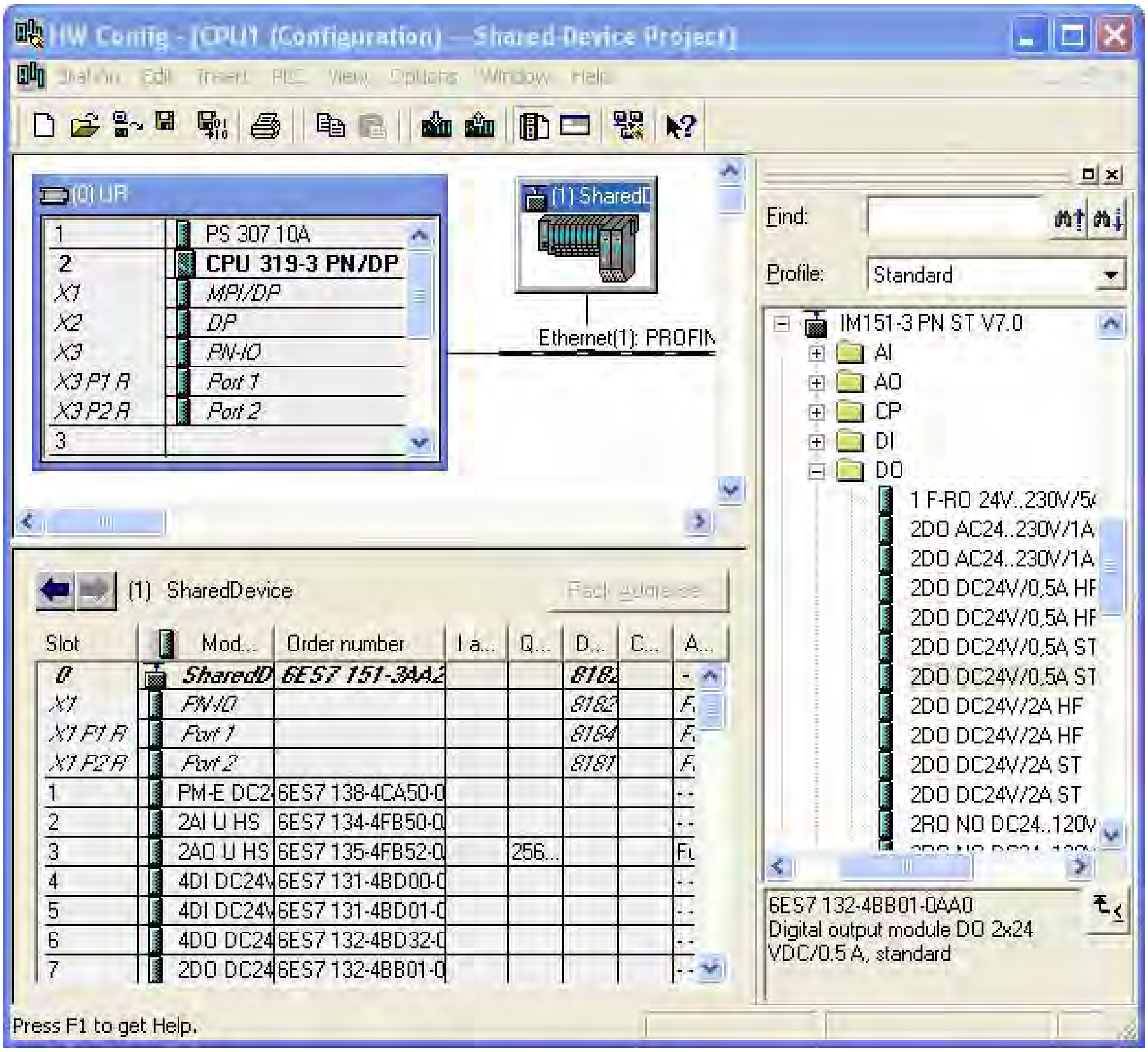
To be able to use the shared device function, you need to take certain configuration steps both in the SIMATIC Manager and in HW Config.

Preparations

1. Create a project called "Shared device project" in the SIMATIC Manager.
2. Insert two stations (SIMATIC 300).
3. Open the stations in HW Config and configure a CPU with a PROFINET interface (in the case described here CPU 319-3 PN/DP).
4. Set the parameters for the PROFINET interfaces of the stations you have just created.
5. "Save and compile" the individual stations.

Creating the shared device

1. Open one of the stations you created in HW Config.
2. Configure a distributed I/O system ET 200s (IM151-3PN) with several submodules as shown in the screenshot.



1	PS 307 10A
2	CPU 319-3 PN/DP
X1	MPI/DP
X2	DP
X3	PN-IO
X3 P1 R	Port 1
X3 P2 R	Port 2
3	

(1) SharedDevice

Slot	Mod...	Order number	I a...	Q...	D...	C...	A...
0	SharedD	6ES7 151-3AA2			818		-
X1	PN-IO				8182		F
X1 P1 R	Port 1				8184		F
X1 P2 R	Port 2				8181		F
1	PM-E DC24	6ES7 138-4CA50-0					--
2	2AI U HS	6ES7 134-4FB50-0					--
3	2AO U HS	6ES7 135-4FB52-0		256...			Fu
4	4DI DC24V	6ES7 131-4BD00-0					--
5	4DI DC24V	6ES7 131-4BD01-0					--
6	4DO DC24	6ES7 132-4BD32-0					--
7	2DO DC24	6ES7 132-4BB01-0					--

6ES7 132-4BB01-0AA0
Digital output module DO 2x24 VDC/0.5 A, standard

3. Copy the distributed I/O system you have just created using the shortcut menu (right mouse button).
4. Save the hardware configuration and close the configured station.
5. Open the other station you created earlier in HW Config.
6. To insert the distributed I/O system as a shared device, right-click on the PROFINET line. Select the "Paste Shared" command in the shortcut menu.
7. Save the hardware configuration and close the configured station.

You have successfully created the shared device; now set the assignments of the submodules to the configured stations.

Assigning submodules

The submodules must be assigned separately for each station. Remember that changes to a station will also impact the other station(s)! A submodule can only ever be assigned to one station!

1. Open the Properties dialog of the distributed I/O system in the first station.
2. Go to the "Access" tab.
3. Configure the access to the individual submodules. Select the type of access from the drop-down list in the "Value" column. You can choose between:
   - No access to the submodule: "- - -"
   - Full access to the submodule: "Full"
4. Note that the setting "Full" automatically leads to the setting "- - -" in the other station(s); The setting "- - -", on the other hand, does not automatically lead to the setting "Full" in the other station(s).
5. Save and close the station.
6. Repeat steps 1. to 4. for the second station.
7. Then download the configuration to the stations.

Shared device in the user program

The shared device has no special role in the user program. The submodules assigned in the station are addressed as usual, the other submodules do not receive addresses.

4.5.2.3 Steps in Configuring a Shared Device in Different STEP 7 Projects

Introduction

The following example describes the configuration of a shared device in different STEP 7 projects. In the example, two IO controllers share the submodules of an IO device.

Procedure

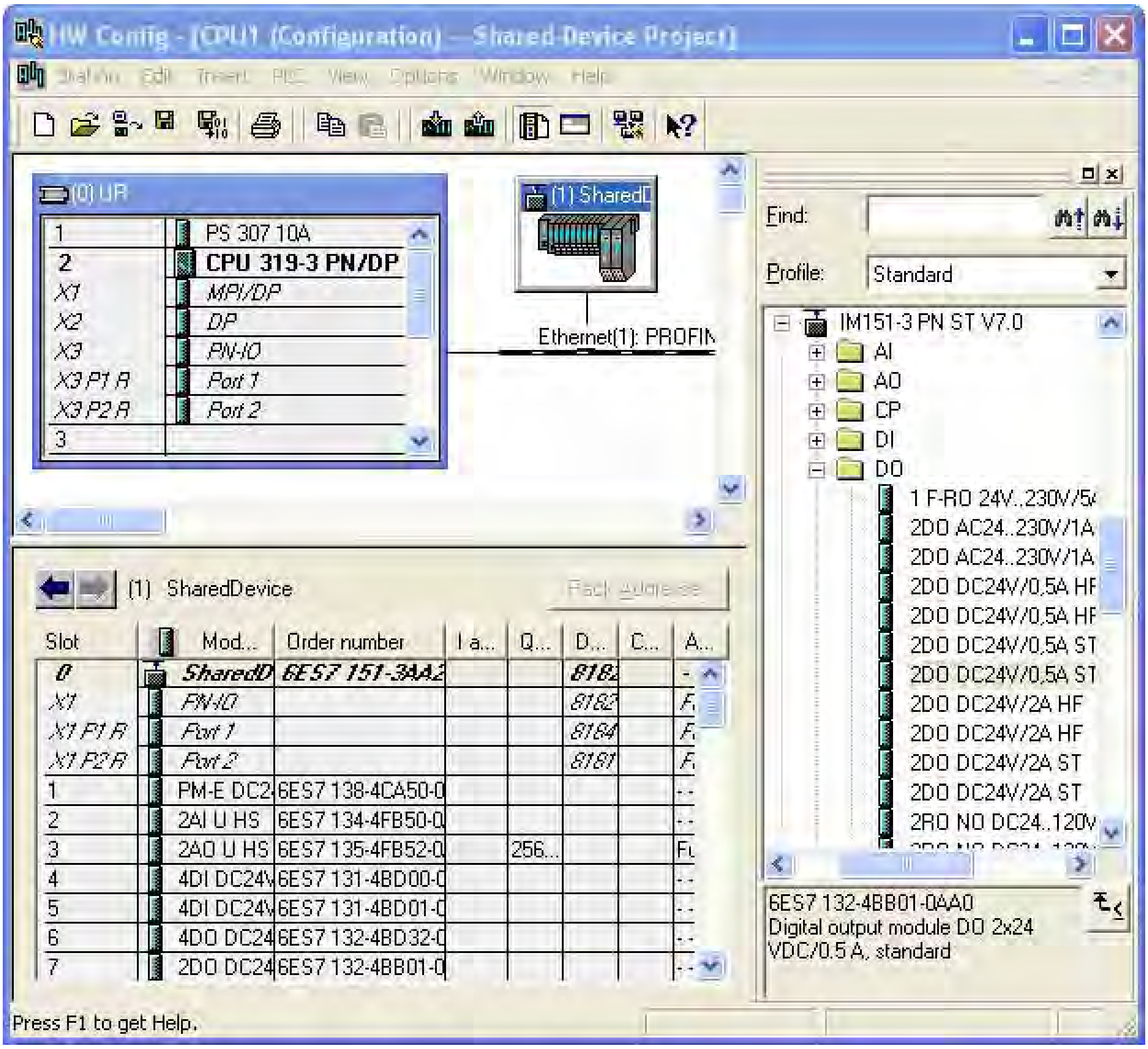
To be able to use the shared device function, you need to take certain configuration steps both in the SIMATIC Manager and in HW Config.

Preparations

1. Create a project called "Shared device 1" in the SIMATIC Manager.
2. Insert a station (SIMATIC 300) with the name "CPU1".
3. Open the station in HW Config and configure a CPU with a PROFINET interface (in the case described here CPU 319-3 PN/DP).
4. Set the parameters for the PROFINET interface of the station you have just created.
5. "Save and compile" the station and close the project.
6. Create a second project called "Shared device 2" in the SIMATIC Manager.
7. Insert a station (SIMATIC 300) with the name "CPU2".
8. Repeat steps 3-5.

Creating the shared device

1. Open the project "Shared device 1".
2. Open the station "CPU1" in HW Config.
3. Configure a distributed I/O system ET 200s (IM151-3PN) with several submodules as shown in the screenshot.



1	PS 307 10A
2	CPU 319-3 PN/DP
X1	MPI/DP
X2	DP
X3	PN-IO
X3 P1 R	Port 1
X3 P2 R	Port 2
3	

(1) SharedDevice

Slot	Mod...	Order number	I a...	Q...	D...	C...	A...
0	SharedD	6ES7 151-3AA2			818		-
X1	PN-IO				8182		F.
X1 P1 R	Port 1				8184		F.
X1 P2 R	Port 2				8181		F.
1	PM-E DC2	6ES7 138-4CA50-0					--
2	2AI U HS	6ES7 134-4FB50-0					--
3	2AO U HS	6ES7 135-4FB52-0		256...			Fu
4	4DI DC24V	6ES7 131-4BD00-0					--
5	4DI DC24V	6ES7 131-4BD01-0					--
6	4DO DC24	6ES7 132-4BD32-0					--
7	2DO DC24	6ES7 132-4BB01-0					--

6ES7 132-4BB01-0AA0
Digital output module DO 2x24 VDC/0.5 A, standard

4. Note down the precise configuration of the ET 200s you have just created.
5. Save and compile the hardware configuration and close HW Config and the project.
6. Open the project you created earlier "Shared device 2".

7. Open the station "CPU2" in HW Config.
8. Configure a distributed I/O system ET 200s with exactly the same configuration as the ET 200s in the "Shared device 1" project in station "CPU1" (notes from step 4.)
9. Save and compile the hardware configuration and close HW Config and the project.

You have successfully created the shared device; now set the assignments of the submodules to the configured stations.

Assigning submodules

The submodules must be assigned separately for each station in both projects. Note that a submodule can only ever be assigned to one station!

1. Open the "CPU1" station of the "Shared device 1" project in HW Config.
2. Open the Properties dialog of the distributed I/O system.
3. Go to the "Access" tab.
4. Configure the access to the individual submodules. Select the type of access from the drop-down list in the "Value" column. You can choose between:
   - No access to the submodule: "- - -"
   - Full access to the submodule: "Full"
5. Save and compile the hardware configuration and close HW Config and the project.
6. Open the "CPU2" station of the "Shared device 2" project in HW Config.
7. Repeat steps 2. to 5.

Caution

Access rule: a submodule can only ever be assigned to one IO controller. This means, for example, that the submodule in slot 4 can only be assigned to "CPU1" with the setting "Full access"; in "CPU2", the setting "- - -" ("No access") must therefore by selected for the submodule in slot 4.

Bandwidth reservation

When configuring shared devices in different projects, STEP 7 requires information on the other locations of use of the shared device to be able to calculate the bandwidth. Make the following settings in **both** projects:

1. Open the "Shared device 1" / "Shared device 2" project.
2. Open the station "CPU1" / "CPU2" in HW Config.
3. Open the Properties dialog of the PROFINET IO interface and go to the "Shared Device" tab
4. Make the following settings:
5. If the IO controller has full access to the interface of the IO device: The number of **external** controllers that access the IO device.
6. If the IO controller has no access to the interface of the IO device: The send clock of the IO controller with full access.
7. Save and compile the configuration.
8. Then download the configuration to the stations.

Shared device in the user program

The shared device has no special role in the user program. The submodules assigned in the station are addressed as usual, the other submodules do not receive addresses.

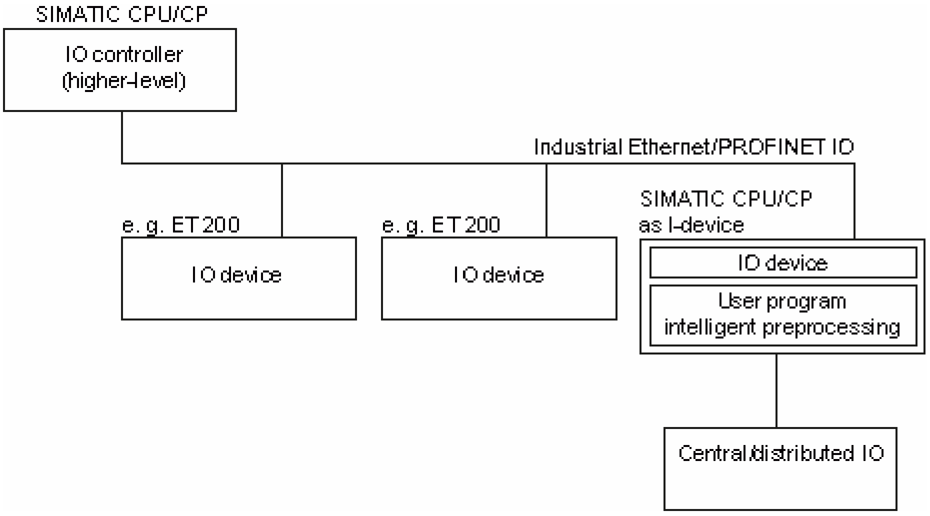
4.6 I-Devices

4.6.1 Overview

4.6.1.1 I-Device Functionality

The "I-device" functionality (intelligent IO device) of a CPU allows data to be exchanged with an IO controller and therefore to use the CPU, for example, as an intelligent preprocessing unit of subprocesses. The I-device is linked as an IO device to a "higher-level" IO controller.

The preprocessing is handled by the user program on the CPU. The process values acquired in the central or distributed (PROFINET IO or PROFIBUS DP) I/O are preprocessed by the user program and made available via a PROFINET IO device interface to the CPU or the CP of a higher-level station.


SIMATIC CPU/CP
IO controller (higher-level)
Industrial Ethernet/PROFINET IO
e. g. ET200
IO device
e. g. ET200
IO device
SIMATIC CPU/CP as I-device
IO device
User program intelligent preprocessing
Central/distributed IO

"I-device" naming conventions

In the remainder of this description, a CPU or a CP with the I-device functionality is simply called an "I-device".

4.6.1.2 Properties and Advantages of the I-Device

Areas of application

Areas of application of the I-device:

- Distributed processing
  A complex automation task can be divided into smaller units/subprocesses. This makes the process easier to handle because the subtasks are simpler.
- Separating subprocesses
  Complicated, widely distributed and extensive processes can be subdivided into several subprocesses with manageable interfaces by using I-devices. These subprocesses in turn are stored in the individual STEP 7 projects that are merged to form an overall project.

Properties

Properties of the I-device:

- Separation of STEP 7 projects
  Creator and user of an I-device can have completely separate STEP 7 projects. The interface between the STEP 7 projects is the GSD file. This allows a link to standard IO controllers via a standardized interface.
- Real-time communication
  The I-device is made available for a deterministic PROFINET IO system via a PROFINET IO interface and therefore supports real-time communication and isochronous real time.

Advantages

The I-device has the following advantages:

- Simple linking of IO controllers without additional software tools
- Real-time communication between SIMATIC CPUs and with standard IO controllers
- By distributing the computing power on several I-devices, the required computing power of the individual CPUs and, of course, of the IO controller can be reduced
- Lower communications load by processing process data locally
- Manageable processing of subtasks in separate STEP 7 projects

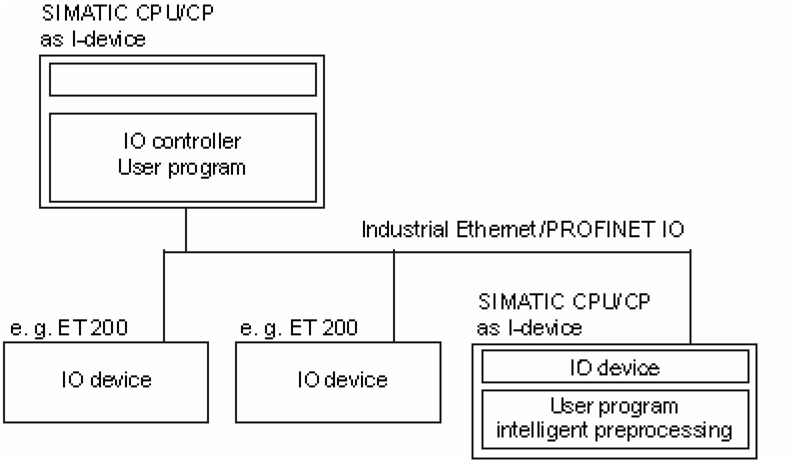
4.6.1.3 Characteristics of an I-Device

Basics

An I-device is included in an IO system like a standard IO device.

I-device

The I-device does not have its own distributed I/O (no IO controller). The configuration and parameter assignment of the I-devices in the role of an IO device is the same as for a distributed I/O system (for example ET 200).

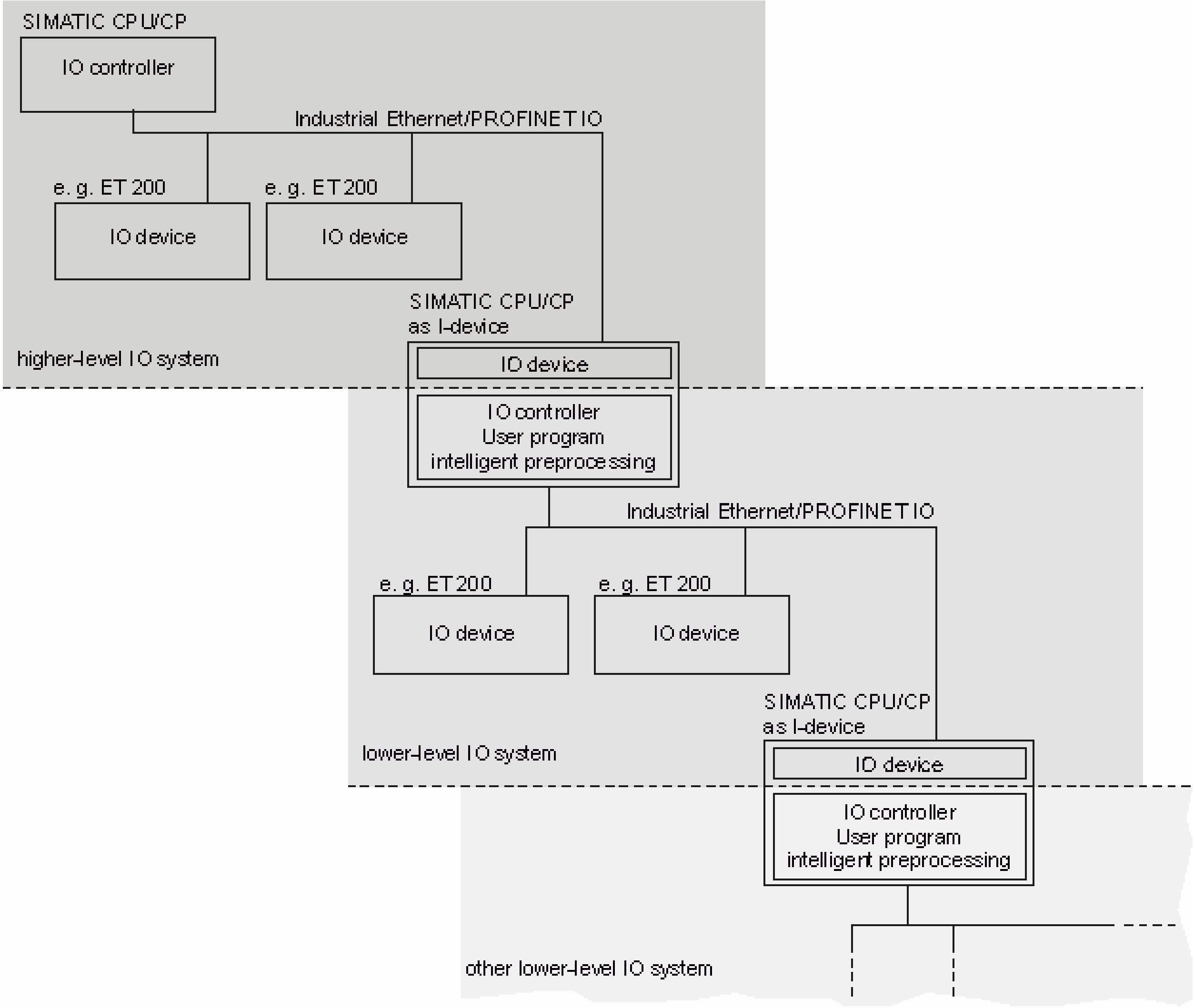


I-device with underlying PNIO system

Depending on the configuration, an I-device can also be an IO controller on a PROFINET interface in addition to having the role of IO device.

This means that the I-device can be part of a higher-level IO system via its PROFINET interface and as an IO controller can support its own lower-level IO system with its central or distributed I/O.

The lower-level IO system can, in turn, contain I-devices (see figure below). This makes hierarchically structured IO systems possible.

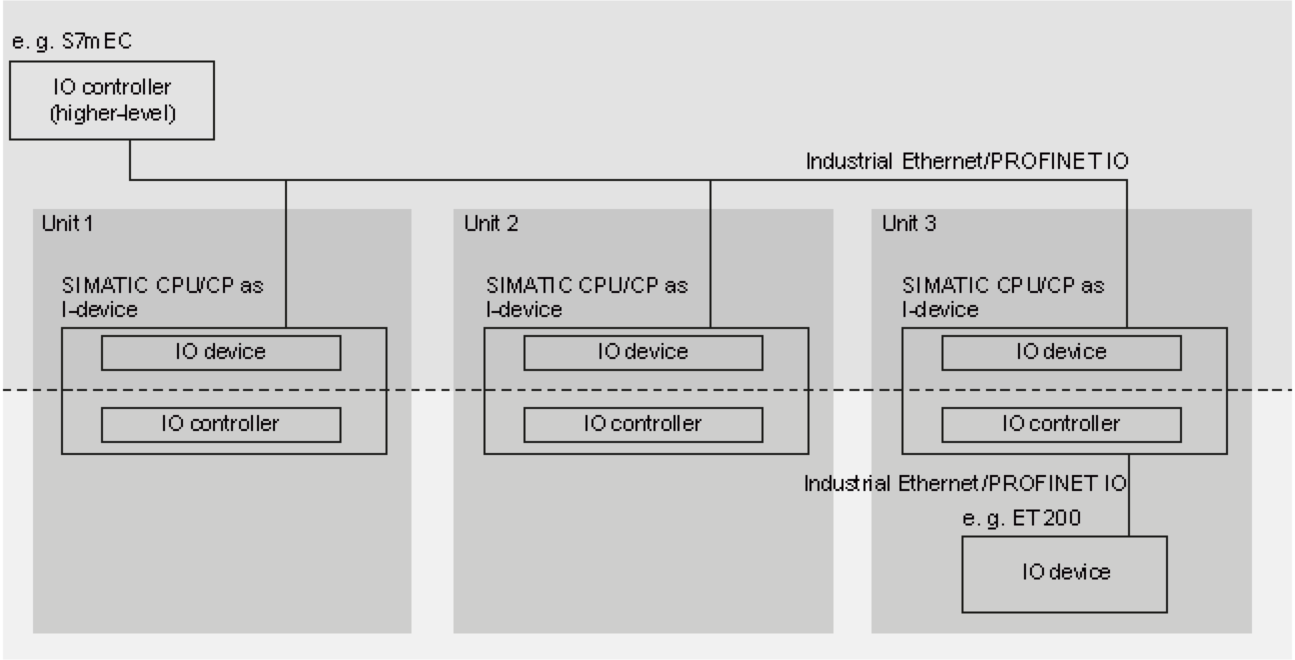


I-device as shared device

An I-device can also be used by more than one IO controller at the same time as a shared device.

Example - the I-device as IO device and IO controller

The I-device as IO device and IO controller is explained based on the example of a print process. The I-device controls a unit (a subprocess). One unit is used, for example, to insert additional sheets such as flyers or brochures in a package of printed material.


e. g. S7mEC
IO controller (higher-level)
Industrial Ethernet/PROFINET IO
Unit 1
SIMATIC CPU/CP as I-device
IO device
IO controller
Unit 2
SIMATIC CPU/CP as I-device
IO device
IO controller
Unit 3
SIMATIC CPU/CP as I-device
IO device
IO controller
Industrial Ethernet/PROFINET IO
e. g. ET 200
IO device

Unit 1 and unit 2 each consist of an I-device with central I/O. The I-device along with the distributed I/O system (for example ET 200) forms unit 3.

The user program on the I-device is responsible for preprocessing the process data. For this task, the user program of the I-device requires default settings (for example control data) from the higher-level IO controller. The I-device provides the higher-level IO controller with the results (for example status of its subtask).

With suitable configuration, the control data and the results can also be called up directly from the lower-level I/O. This allows the higher-level IO controller direct access to the lower-level I/O.

4.6.1.4 Data Exchange between Higher- and Lower-level IO System

Introduction

The next section shows the data exchange between the higher- and lower-level IO system.

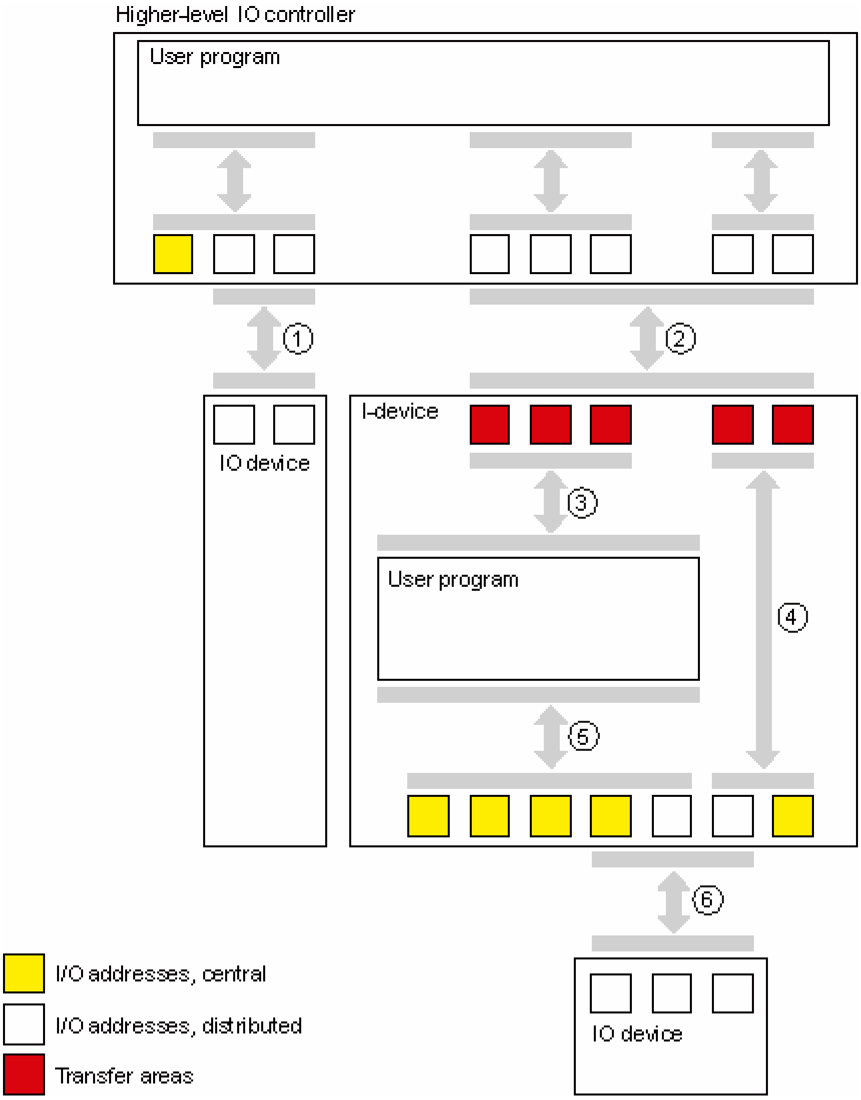
Transfer areas

The data for communication between IO controller and I-device is made available in the transfer areas. A transfer area contains an information unit that is exchanged consistently between IO controller and I-device. You will find more information on configuration and use of transfer areas in Configuration*)*

There are two types of transfer area:

- Application transfer areas are an interface to the user program of the I-device CPU. Inputs are processed in the user program and outputs are the result of the processing in the user program.
- IO transfer areas forward data from the higher-level IO controller to the I/O or vice versa. There is no processing of the values on the I-device.

The next figure shows the data exchange between the higher- and lower-level IO system. The individual communication relations are explained below based on the numbers.



(1) Data exchange between the higher-level IO controller and normal IO device

In this way, the IO controller and IO device exchange data. The data transfer is via PROFINET.

(2) Data exchange between the higher-level IO controller and I-device

In this way, the IO controller and the I-device exchange data. The data transfer is via PROFINET.

The data exchange between a higher-level IO controller and an I-device is based on the conventional IO controller / IO device relationship.

For the higher-level IO controller, the transfer areas of the I-devices represent submodules of a preconfigured station.

The output data of the IO controller is the input data of the I-device. Analogously, the input data of the IO controller is the output data of the I-device.

(3) Transfer relation between the user program and transfer area

In this way, the user program and the application transfer area exchange input and output data.

(4) Transfer relation between the I/O address area and I/O of the I-device

In this way, the I-device passes on the data of its central and distributed I/O to the IO transfer area. The data is not processed by the CPU.

(5) Transfer relation between the user program and the I/O

In this way, the user program and the central / distributed I/O exchange input and output data.

(6) Data exchange between the I-device and a lower-level IO device

In this way, the I-device and its IO devices exchange data. The data transfer is via PROFINET.

4.6.2 Configuring an I-Device in STEP 7

Introduction

Two different perspectives must be distinguished during configuration:

- Creating the I-device
- Using the I-device

Based on an example, the section on Creating an I-Device shows how an IO system is configured with an I-device. How to import an existing project and use this for your applications is described in Using an I-Device.

Basic procedure for configuring and programming

Creating the I-device

1. Configuration of the I-device with central and/or distributed I/O modules
2. Assignment of parameters for the PROFINET interface of the I-device
3. Configuration of the transfer areas of the I-device
4. Generation of a GSD file

Using the I-device

1. Installing the GSD file
2. Configuration of the higher-level IO controller
3. Assignment of parameters for the PROFINET interface of the higher-level IO controller with central and distributed I/O modules
4. Configuration of the I-device on the IO system of the higher-level IO controller
5. Programming the user programs

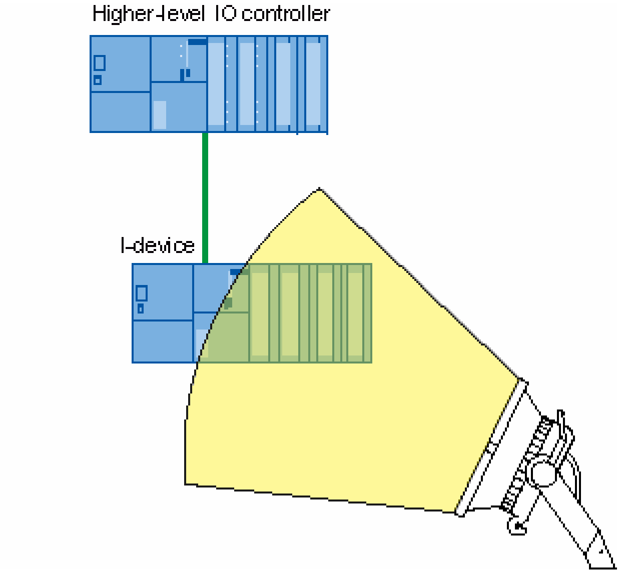
4.6.2.1 Creating an I-Device

Configuration example

How you configure a small automation system with an I-device is shown based on an example.

Control tasks and preprocessing are handled by a CPU 317-2 PN/DP operating as an I-device.

The following graphic shows the configuration of the application. it consists of a higher-level IO system and the I-device. The I-device is first illuminated separately from the higher-level IO system (cone of light from the search light).



Based on this example, you will see how to do the following:

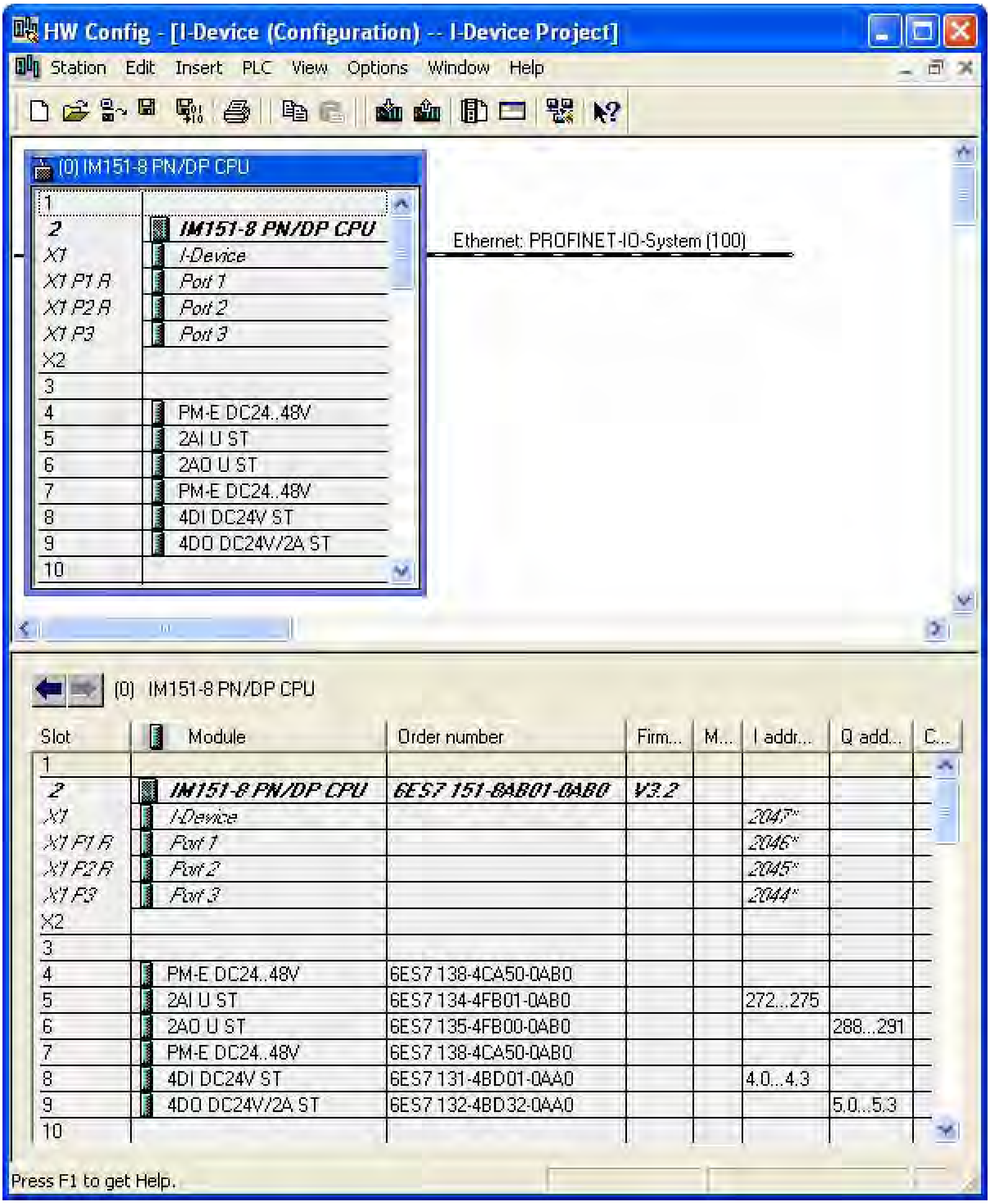
- Configure an I-Device
- Configure Transfer Areas
- Configure I/O Transfer Areas
- Generate a GSD File for an I-Device

4.6.2.2 Configuring the I-Device

Preparations

1. Create a project with the name "I-Device Project" in the SIMATIC Manager.
2. Add a new "SIMATIC 300 station" with the name "I-Device".
3. Open the station in HW Config and configure an ET 200S-CPU (IM 151- 8 PN/DP CPU).
4. Configure the IP address parameters. Refer to the section IP address parameters in the section Constraints on the Use of I-Devices
5. Add the central I/O.

The following figure shows the configuration once you have taken all the steps.


HW Config - [I-Device (Configuration) -- I-Device Project]
Station
Edit
Insert
PLC
View
Options
Window
Help
(0) IM151-8 PN/DP CPU
Ethernet: PROFINET-IO-System (100)
Press F1 to get Help.

(0) IM151-8 PN/DP CPU

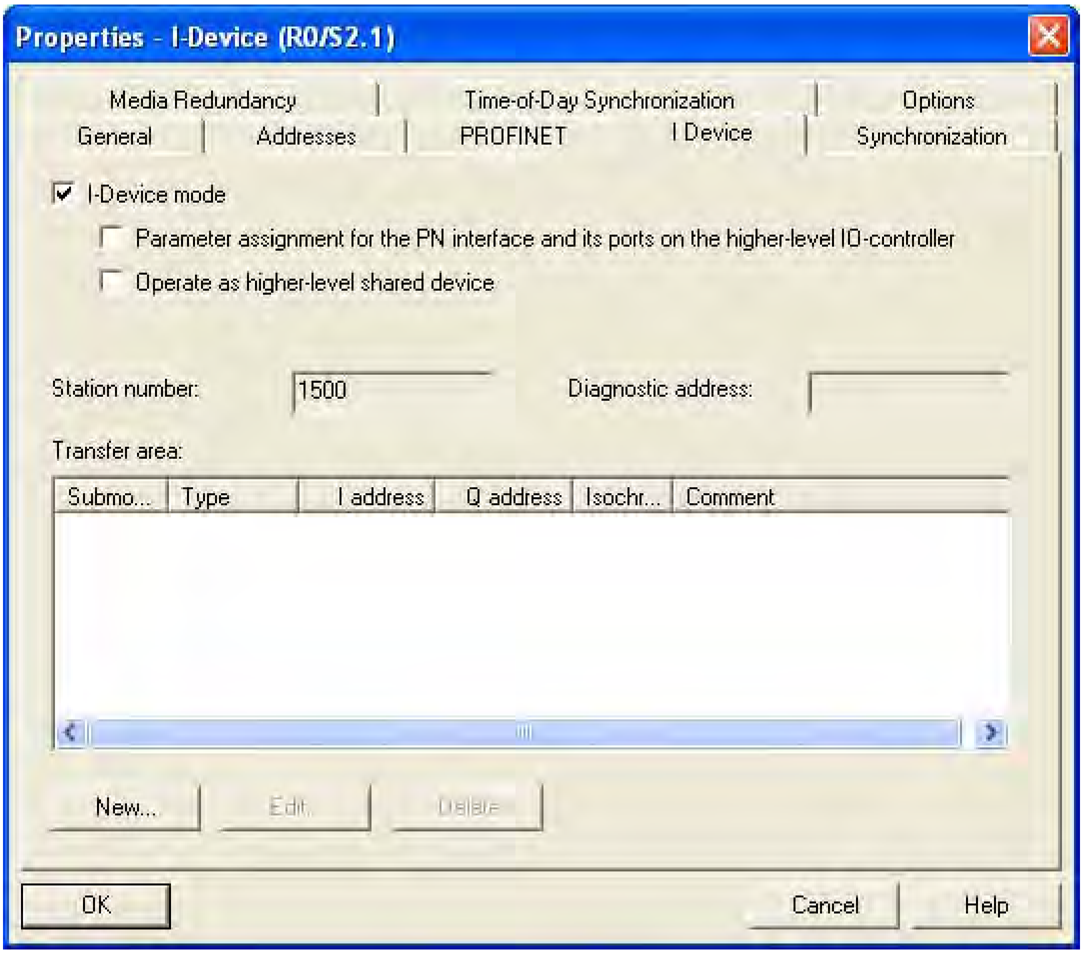
1	
2	***IM151-8 PN/DP CPU***
X1	*I-Device*
X1 P1 R	*Port 1*
X1 P2 R	*Port 2*
X1 P3	*Port 3*
X2	
3	
4	PM-E DC24..48V
5	2AI U ST
6	2AO U ST
7	PM-E DC24..48V
8	4DI DC24V ST
9	4DO DC24V/2A ST
10	

(0) IM151-8 PN/DP CPU

Slot	Module	Order number	Firm...	M...	I addr...	Q add...	C...
1							
2	***IM151-8 PN/DP CPU***	***6ES7 151-8AB01-0AB0***	***V3.2***				
X1	*I-Device*				*2047**		
X1 P1 R	*Port 1*				*2046**		
X1 P2 R	*Port 2*				*2045**		
X1 P3	*Port 3*				*2044**		
X2							
3							
4	PM-E DC24..48V	6ES7 138-4CA50-0AB0					
5	2AI U ST	6ES7 134-4FB01-0AB0			272...275		
6	2AO U ST	6ES7 135-4FB00-0AB0				288...291	
7	PM-E DC24..48V	6ES7 138-4CA50-0AB0					
8	4DI DC24V ST	6ES7 131-4BD01-0AA0			4.0...4.3		
9	4DO DC24V/2A ST	6ES7 132-4BD32-0AA0				5.0...5.3	
10							

Configuration

To be able to use the ET 200S CPU you have just configured as an I-device, you will first need to make several settings in the "I-Device" tab of the interface properties:



☑ I-Device mode

□ Parameter assignment for the PN interface and its ports on the higher-level IO-controller

□ Operate as higher-level shared device

Station number: 1500

Diagnostic address:

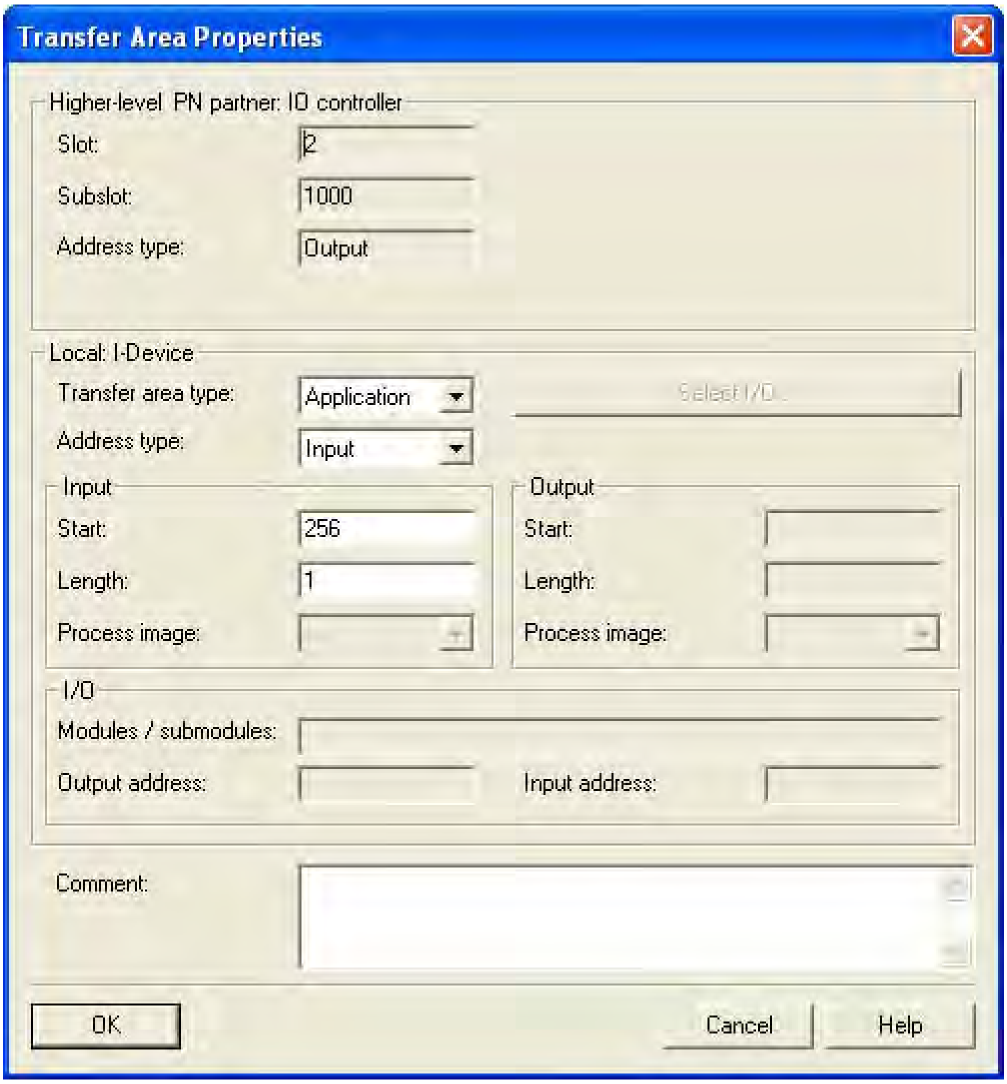
Submo...	Type	I address	Q address	Isochr...	Comment

1. Select the "I-device mode" check box.
2. If the "The "Parameter assignment for the PN interface and its ports on the higher-level IO controller" check box is selected, the interface and port parameters are assigned by the higher-level IO controller. If the check box is not selected, the parameters are assigned in this station.
3. The I-device automatically obtains the specific station number 1500 from STEP 7. This cannot be edited and is therefore unavailable. The station number is part of the geographic address for the transfer areas of the I-device.
4. The settings for the transfer areas are explained in the next section.

4.6.2.3 Configuring Transfer Areas: Application

Creating

To create an application transfer area, click the "New..." button in the "Transfer area" part of the "I-Device" tab. The "Transfer Area Properties" dialog box opens.



Slot: 2

Subslot: 1000

Address type: Output

Transfer area type: Application

Address type: Input

Start: 256

Length: 1

Here, you set the properties of the transfer area; follow the steps below:

1. The "Transfer area type" drop-down list allows you to make the following settings depending on the CPU you are using:
   - "Application"
   - "I/O"

   Select Application for an application transfer area. The values of the transfer area on the higher-level IO controller (slot and subslot) are assigned automatically by STEP 7, the boxes cannot be edited.
2. Specify whether or not the transfer area is to be an input or output transfer area locally. You do this by selecting the relevant address type from the "Address type" drop-down list.
   STEP 7 assigns the address type of the higher-level IO controller automatically. If the transfer area is to appear on the higher-level IO controller as an output, it must be an input on the I-device and vice versa.
3. Just like every other submodule, a transfer area also requires an address space to be accessed by the user program. You therefore set the start address, length and the process image of the input/output.
4. Enter any additional information as the comment and exit the dialog with "OK".

The transfer area has now been created and is displayed in the "I-Device" tab with its data.

4.6.2.4 Configuring Transfer Areas: I/O

Creating

To create an I/O transfer area, click the "New..." button in the "Transfer area" part of the "I-Device" tab. The "Transfer Area Properties" dialog box opens.

Here, you set the properties of the transfer area; follow the steps below:

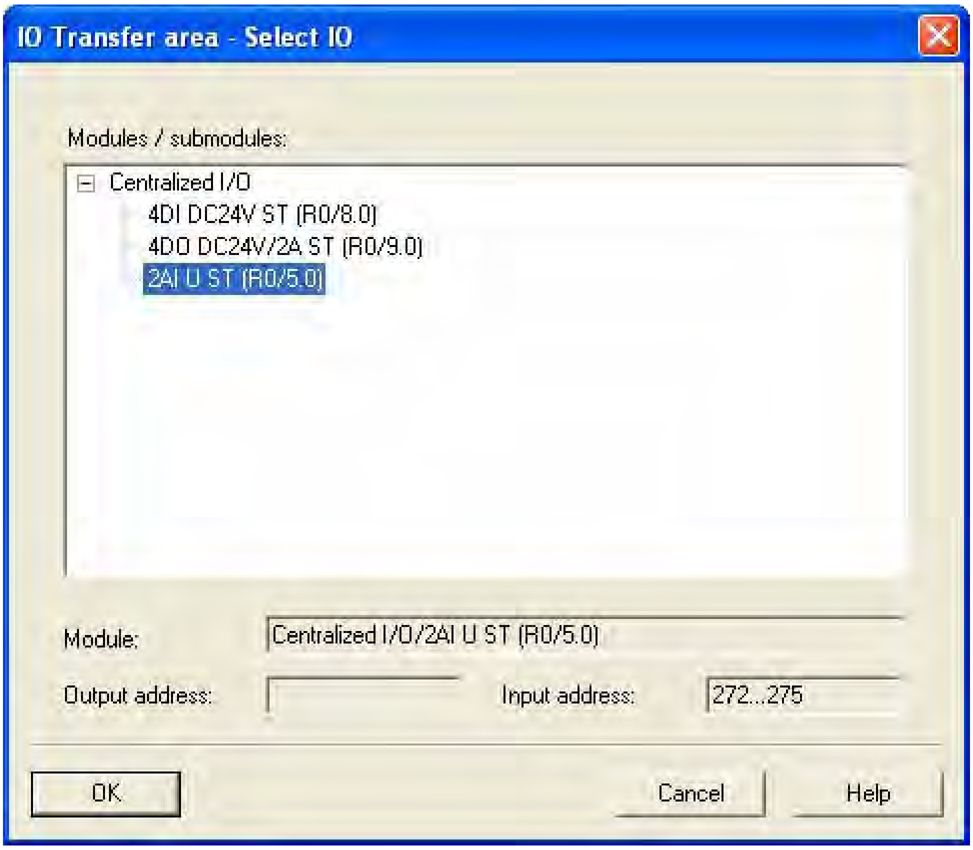
1. The "Transfer area type" drop-down list allows you to make the following settings depending on the CPU you are using:
   - "Application"
   - "I/O"

   Select I/O for an I/O transfer area. The values of the transfer area on the higher-level IO controller (slot and subslot) are assigned automatically by STEP 7, the boxes cannot be edited.

   Note

   If the "I/O" setting is not available, the CPU you are using does not support I/O address ranges.
2. Now specify which modules / submodules of the I-device will be available to the higher-level IO controller as IO transfer areas. Click the "Select I/O" button: the "IO Transfer Area - Select IO" dialog box opens.

3. Select a module / submodule and exit the dialog with "OK".



IO Transfer area - Select IO

Modules / submodules:

Centralized I/O
4DI DC24V ST (R0/8.0)
4DO DC24V/2A ST (R0/9.0)
2AI U ST (R0/5.0)

Module: Centralized I/O/2AI U ST (R0/5.0)

Output address:

Input address: 272...275

4. Just like every other submodule, a transfer area also requires an address space to be accessed by the user program. You create this by setting the start address of the input / output. The length is calculated automatically based on the selected module / submodule.
5. Enter any additional information as the comment and exit the dialog with "OK".

The transfer area has now been created and is displayed in the "I-Device" tab with its data.

4.6.2.5 Generating a GSD File

Generating and importing a GSD file

The next step is to generate a GSD file for the I-device CPU and to store this in the hardware catalog of HW Config or in the file system for other uses.

Procedure

1. Click the menu command "Options" -> "Create GSD File for I-Device" in HW Config. The "Create GSD File for I-Device" dialog box opens.
2. The I-device CPU is already set in the "I-device" drop-down list.
   The identifier assigned in the "Identifier for generic I-device" box is the later name of the generic I-device that will be shown on the higher-level IO controller. The box has the device name as default. You can assign the name yourself according to the rules of the DNS conventions.

Note

- If more than one I-device CPU is configured in a rack, you will need to select the I-device CPU from the "I-device" drop-down list.
- If the device name is obtained by a "different method" the physical address of the I-device CPU will be assigned as the identifier for the generic I-device, for example "R0S2.5" (corresponds to rack 0 slot 2.5).

---

3. Now create the GSD file by clicking the "Create" button. If the file was created successfully, the "Install" and "Export" buttons will be visible and the name of the GSD file is displayed.
4. The GSD file you have just created can now be installed and/or exported using the relevant buttons:
   - "Install" button: The GSD file is installed on your computer and included in the hardware catalog in "PROFINET IO -> Preconfigured Stations -> CPU Name" under the assigned identifier.
   - "Export" button: The GSD file can be exported to another computer for later use or for archiving purposes in the file system.

Note

The GSD file can be installed in HW Config using the menu command **Options > Install GSD Files...**.

---

5. Close the "Create GSD File for I-Device" dialog box and save and compile the hardware configuration and then close HW Config and the project.

4.6.2.6 Using an I-Device

Introduction

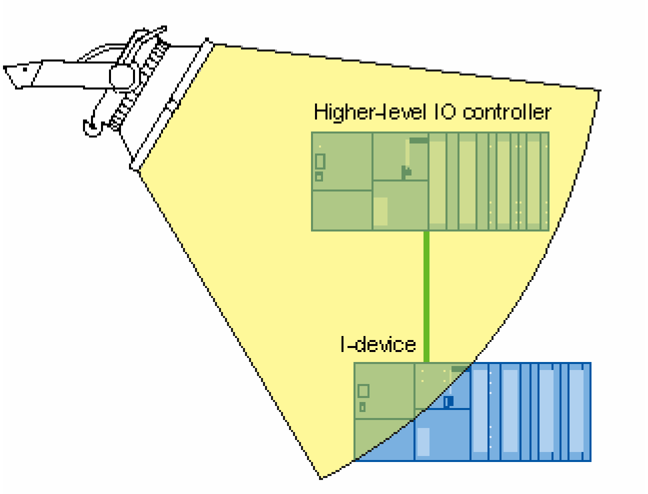
The created I-device will now be used in a higher-level IO system.

Configuration example

Once the I-device has been configured and its parameters set, we move on to the higher-level IO system.

Note

The higher-level IO system does not necessarily need to be in the same STEP 7 project as the I-device. If the higher-level IO system is configured on another computer, you will need to make sure that the GSD file of the I-device was installed.



The following steps are explained based on this example:

- Configuring the I-device in the higher-level IO system
- Access to the transfer areas

4.6.2.7 Configuring the Higher-level IO System

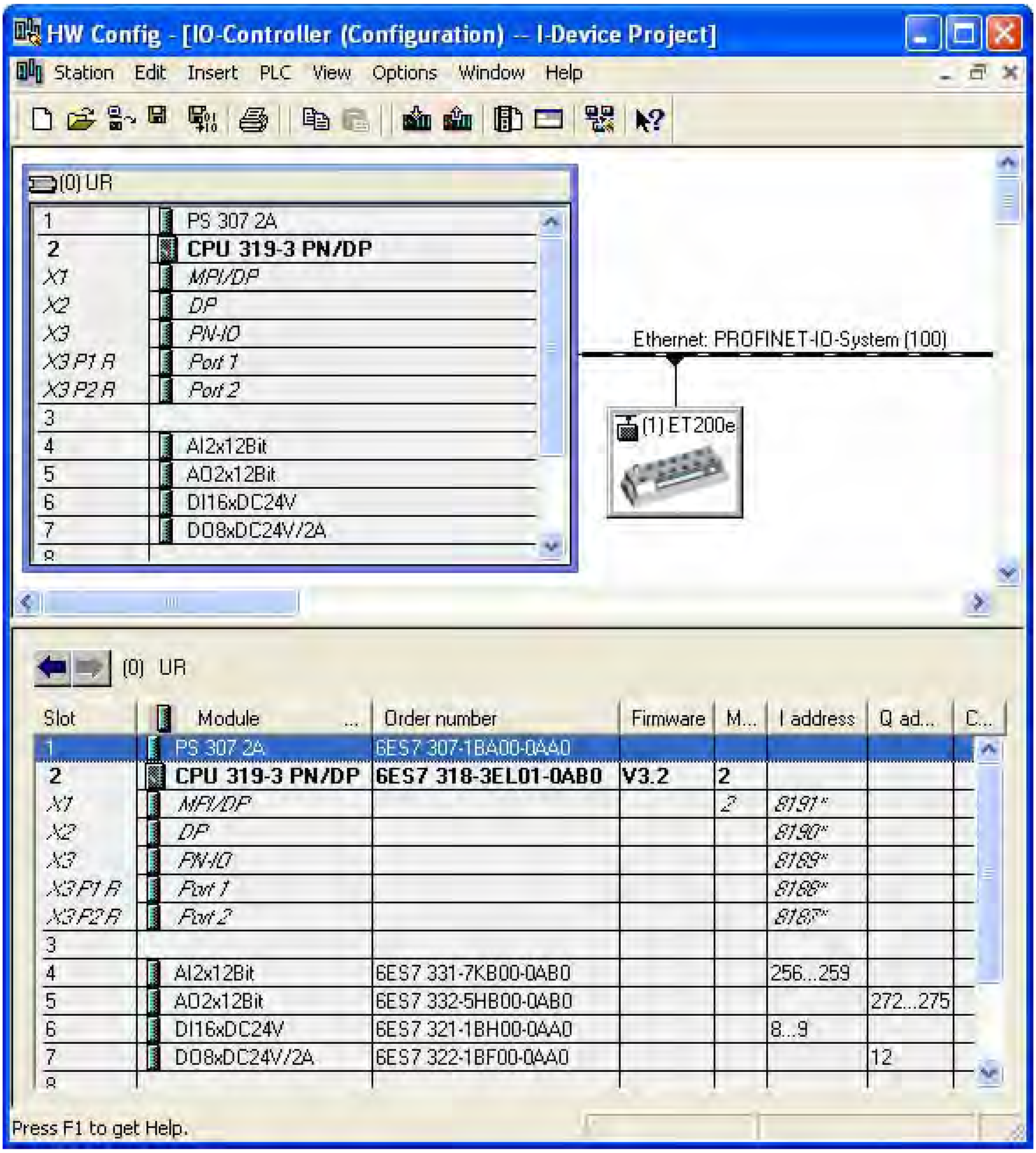
Prerequisites

As described in the earlier sections, you have configured an I-device and generated and installed the GSD file.

Basic steps

1. Create a 300 station as a higher-level IO controller with the name "IO controller".
2. Open the station in HW Config and configure a CPU 319 3 PN/DP with a PROFINET IO system.
3. Configure the central and distributed I/O.

4. The following figure shows the configuration once you have taken all the steps.


HW Config - [IO-Controller (Configuration) -- I-Device Project]
Station Edit Insert PLC View Options Window Help
(0) UR
Ethernet: PROFINET-IO-System (100)
(1) ET200e
(0) UR
Press F1 to get Help.

1	PS 307 2A
2	**CPU 319-3 PN/DP**
X1	*MPI/DP*
X2	*DP*
X3	*PN-IO*
X3 P1 R	*Port 1*
X3 P2 R	*Port 2*
3	
4	AI2x12Bit
5	AO2x12Bit
6	DI16xDC24V
7	DO8xDC24V/2A
8	

Slot	Module ...	Order number	Firmware	M...	I address	Q ad...	C...
1	PS 307 2A	6ES7 307-1BA00-0AA0					
2	**CPU 319-3 PN/DP**	**6ES7 318-3EL01-0AB0**	**V3.2**	**2**			
X1	*MPI/DP*			*2*	*8191**		
X2	*DP*				*8190**		
X3	*PN-IO*				*8189**		
X3 P1 R	*Port 1*				*8188**		
X3 P2 R	*Port 2*				*8187**		
3							
4	AI2x12Bit	6ES7 331-7KB00-0AB0			256...259		
5	AO2x12Bit	6ES7 332-5HB00-0AB0				272...275	
6	DI16xDC24V	6ES7 321-1BH00-0AA0			8...9		
7	DO8xDC24V/2A	6ES7 322-1BF00-0AA0				12	
8							

Configuration

To be able to use the I- device on the higher-level IO system, the I-device GSD file must have been installed earlier (see Generating a GSD File*)*

Follow the steps below to be able to operate the I-device with the higher-level controller:

1. Go to the hardware catalog of HW Config to the "PROFINET IO -> Preconfigured Stations" folder and select the I-device you configured there
2. Drag your I-device to the previously created PROFINET IO system.

Result: The I-device in included in the higher-level IO system. The transfer areas created on the I-device can now be addressed by the user program of the higher-level IO controller.

4.6.2.8 Example of a User Program

Introduction

This simple program example explains how the preprocessing can be implemented with an I-device. The second part of the example explains access to an IO transfer area of the I-device from the user program of the higher-level controller.

Preprocessing on the I-device

Task

The result of a simple "AND" operation on the I-device will be made available on the higher-level IO controller for further processing.

Requirements

You have configured an application transfer area on the I-device with the following properties:

- Local I-device address type: output
- Start address 568, length 1

Solution

STL code: I-device

```
STL
A        I 1.0

A        I 1.1       // ANDing of I1.0 (sensor 1) and I1.1 (sensor 2)

=        Q 568.0     // Write result of operation to Q568.0 (application transfer area of the I-device)
```

STL code: higher-level IO controller

```
STL
A        I 68.0      // Corresponds to Q568.0 of the I-device

=        Q 0.0       // Sets the initial state of Q0.0
```

Note

The addresses of the transfer areas are preset by HW Config. The addresses can, as usual, be changed by the user. In this case, the byte address I 68 was specified for the transfer area.

Access to IO transfer areas

Task

An input byte of an I/O module of the I-device will be made available on the higher-level controller:

Requirements

You have configured an IO transfer area on the I-device with the following properties:

- There is a basis of an input module configured as central I/O on the I-device CPU. In this case, the module "2AI U ST" in slot 5 with logical address 272..275
- Output address in the IO transfer area of the I-device: 223..226

Solution

STL code: I-device

```
STL
// No change necessary in user program
```

No programming is necessary in the user program of the I-device to make the IO transfer areas available. The IO transfer areas are provided by the operating system of the I-device CPU.

STL code: higher-level IO controller

```
STL
L        PIW 223     // Load input byte 223 (content of the IO transfer area and therefore process
                     //data of the underlying input module on the I-device)
```

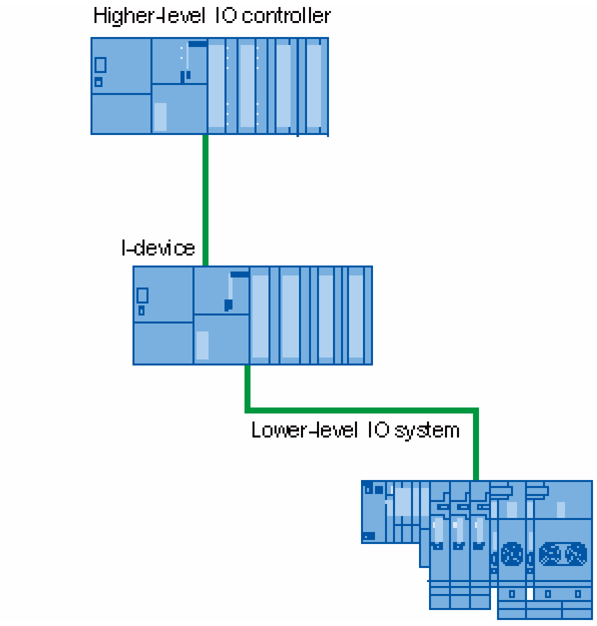
Note

The addresses of the transfer areas are preset by HW Config. The addresses can, as usual, be changed by the user. In this case, IB 27 was specified for the transfer area.

4.6.2.9 Configuring an I-Device with a Lower-level IO System

Distributed I/O on the I-device

The I-device mode also supports operation of distributed PROFIBUS DP IO or PROFINET IO.



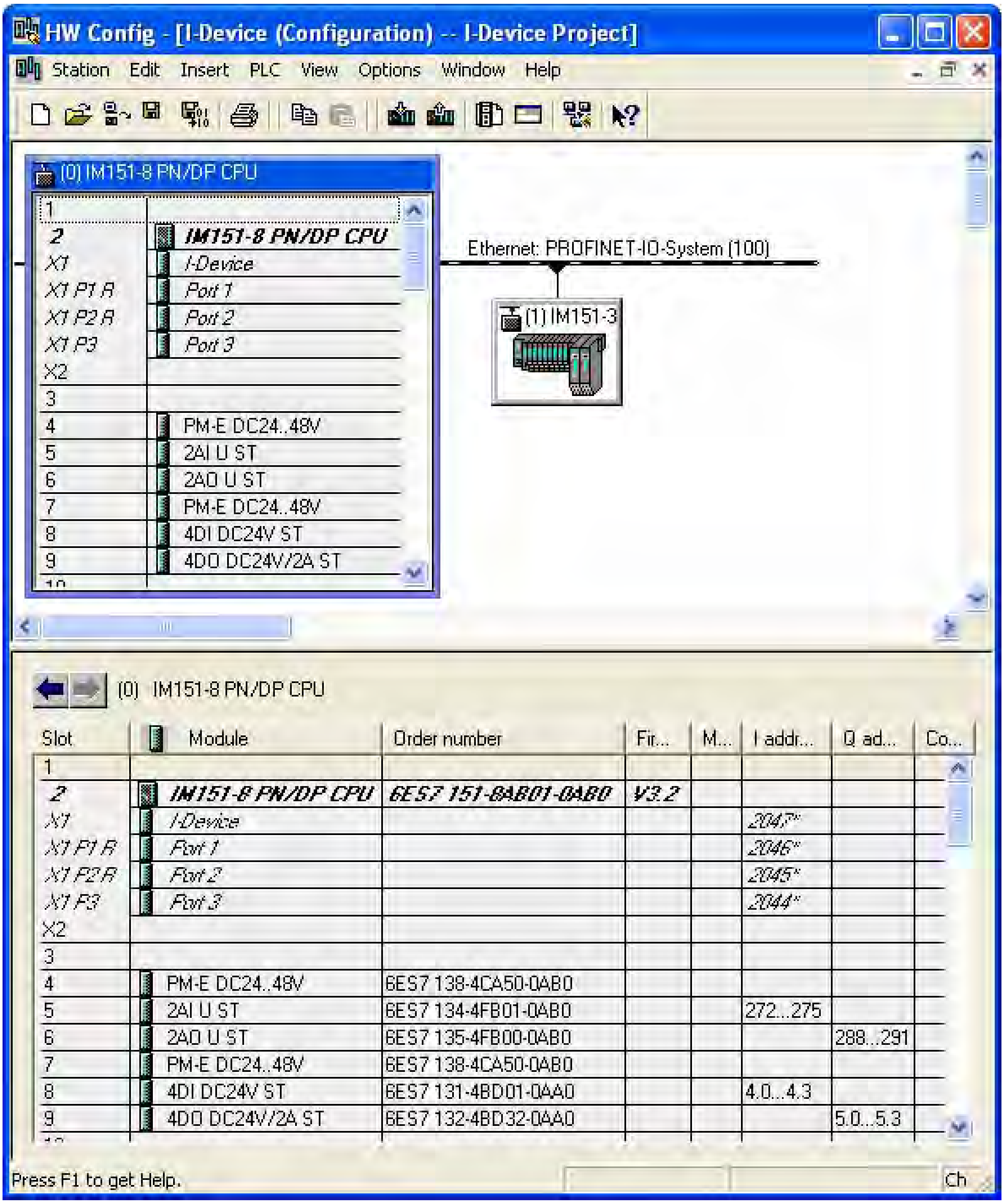
How to configure distributed I/O

The procedure for configuring distributed I/O below an I-devices is exactly the same as the familiar procedure when configuring distributed I/O.

Preparations

1. Create a project with the name "I-Device Project" in the SIMATIC Manager.
2. Add a new "SIMATIC 300 station" with the name "I-Device".
3. Open the station in HW Config and configure an ET 200S CPU with a PROFINET IO system.
4. Add a distributed I/O system ET 200S (for example IM151-3 PN ST) with inputs and outputs.

The following figure shows the configuration once you have taken all the steps.


HW Config - [I-Device (Configuration) -- I-Device Project]
Station Edit Insert PLC View Options Window Help
Ethernet: PROFINET-IO-System (100)
(1) IM151-3
Press F1 to get Help.
Ch

(0) IM151-8 PN/DP CPU

1	
2	***IM151-8 PN/DP CPU***
X1	*I-Device*
X1 P1 R	*Port 1*
X1 P2 R	*Port 2*
X1 P3	*Port 3*
X2	
3	
4	PM-E DC24..48V
5	2AI U ST
6	2AO U ST
7	PM-E DC24..48V
8	4DI DC24V ST
9	4DO DC24V/2A ST

(0) IM151-8 PN/DP CPU

Slot	Module	Order number	Fir...	M...	I addr...	Q ad...	Co...
1							
2	***IM151-8 PN/DP CPU***	***6ES7 151-8AB01-0AB0***	***V3.2***				
X1	*I-Device*				*2047**		
X1 P1 R	*Port 1*				*2046**		
X1 P2 R	*Port 2*				*2045**		
X1 P3	*Port 3*				*2044**		
X2							
3							
4	PM-E DC24..48V	6ES7 138-4CA50-0AB0					
5	2AI U ST	6ES7 134-4FB01-0AB0			272...275		
6	2AO U ST	6ES7 135-4FB00-0AB0				288...291	
7	PM-E DC24..48V	6ES7 138-4CA50-0AB0					
8	4DI DC24V ST	6ES7 131-4BD01-0AA0			4.0...4.3		
9	4DO DC24V/2A ST	6ES7 132-4BD32-0AA0				5.0...5.3	

Creating an I-device

To create the I-device, follow the steps as described earlier in "Configuring the I-Device" in the "Configuration". The rest of the steps are the same as described there.

4.6.2.10 I-Device as Shared Device

Introduction

With only a few configuration steps, an I-device can also be used as a shared device.

Configuration

To configure an I-device as a shared device, follow the steps below:

1. Configure an I-device as described in section Configuring the I-Device
2. Configure the transfer areas as described in section Configuring Transfer Areas: Application*)*.
3. Reopen the "I-Device" tab of the interface properties of the CPU.
4. Select the "Operate as higher-level shared device" check box and exit the dialog with "OK".
5. Generate the GSD file as described in section Generating a GSD File
6. The generated GSD file can be configured as described in section Steps in Configuring a Shared Device in the same STEP 7 Project as a shared device.

4.6.3 Diagnostics, Rules and Constraints

4.6.3.1 Diagnostics and Interrupt Characteristics

Diagnostics and interrupt characteristics

S7 CPUs have numerous diagnostics and interrupt functions that can, for example, report errors/faults or failures of underlying IO systems. These diagnostic messages reduce down times and simplify localization and elimination of problems.

General distinction

The diagnostics and interrupt functions known from "normal" S7 CPUs are also available when using I-devices. There are, however, several special features of diagnostics of I-devices. The explanations are organized as follows:

- Diagnostics of the I-device on the higher-level IO controller
- Diagnostics on the I-device CPU

Diagnostics of the I-device on the higher-level IO controller

The IO controller has the following diagnostics mechanisms available:

- OB83 (submodule return)
- OB85 (station failure)
- OB86 (station return)
- OB122 (access error)

Special features of IO transfer areas:

When using IO transfer areas, only the transfer area itself can be diagnosed (exists, available) but not the underlying I/O module of the I-device CPU.

Note

- Failures of the I/O are only reported acyclically when there is access.
- Load voltage diagnostics can only be handled in the application.

Diagnostics options of the I-device CPU

Diagnostics within the I-device differs from the diagnostics options in "normal" IO systems:

For the I-device CPU, it is important to know whether the higher-level controller is in STOP or RUN and whether the higher-level controller has subscribed to the transfer modules of the I-device CPU. The following options are available for this:

- OB86 (partial and complete startup of the higher-level IO controller)
- OB85 (station failure)
- OB122 (access error)
- OB83 (submodule return)

Note

The diagnostic messages of the I/O can be processed in the user program of the I-device CPU and passed on from there to the higher-level IO controller via transfer areas.

Initial status	Event	I-device response	Response of higher-level controller
Both CPUs are in RUN.	The I-device CPU changes to STOP.	/	OB85 call (process image transfer error) / On access: OB122 call (access error)
I-device CPU is in STOP, higher-level controller is in RUN.	The I-device CPU is starting up.	OB100 call (startup)	OB83 call (submodule return)
Both CPUs are in RUN.	The higher-level IO controller changes to STOP	OB85 and OB122 call (when accessing the input transfer areas) Note: The output transfer areas can be still be accessed.	/
Higher-level IO controller is in STOP, I-device CPU in RUN.	The higher-level IO controller starts up.	OB83 call (submodule return of the input transfer areas)	OB100 call (startup)
Both CPUs are in RUN.	The I-device fails.	/	OB86 call (station failure)
I-device CPU is in STOP, higher-level controller is in RUN.	I-device return	OB100 call (startup)	OB86 call (station return)
Both CPUs are in RUN.	The higher-level IO controller fails.	OB86 call (station failure)	OB86 call (station return) Note: partial station return with shared use

4.6.3.2 Rules for the Topology of a PROFINET IO System with I-Device

Introduction

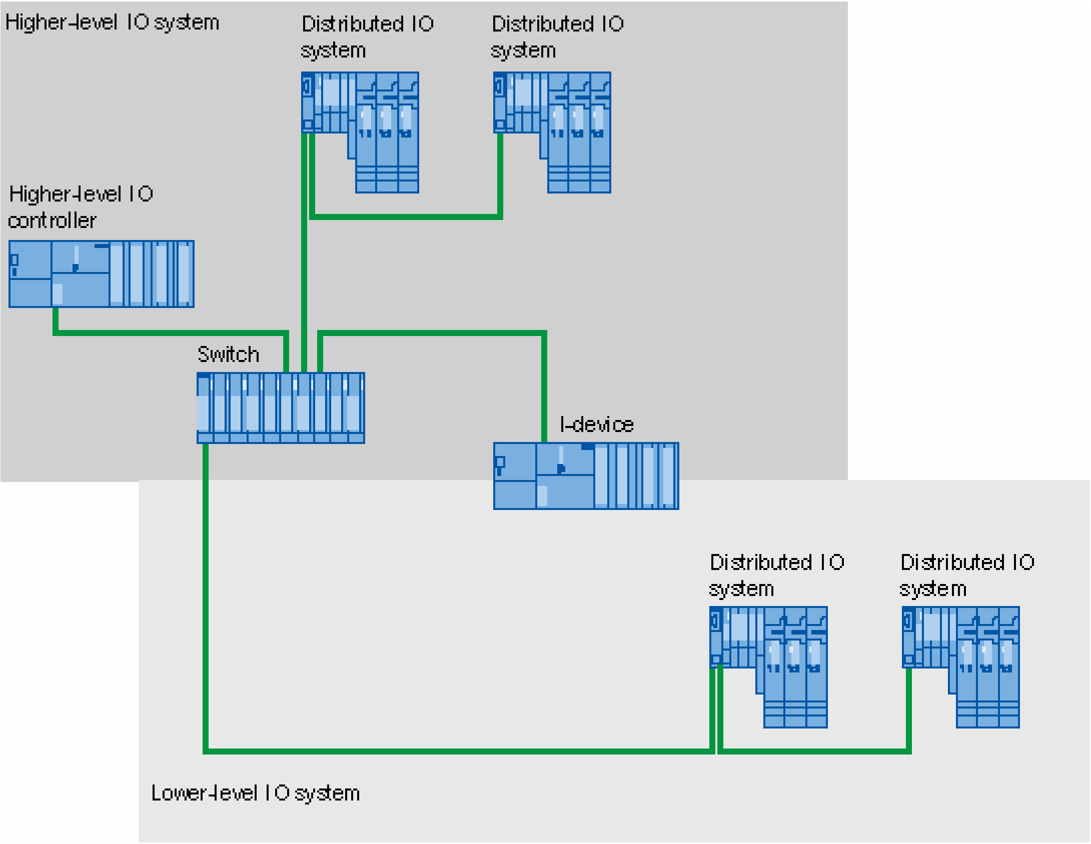
The following recommendations for the structure and configuration of an IO system when using I- devices will help you to keep the bandwidths required for communication small.

Basically, the main concern is that the following communications paths do not cross:

- Communications paths between IO controllers with their IO systems.
- Communications paths of the I-device controller and its IO system.

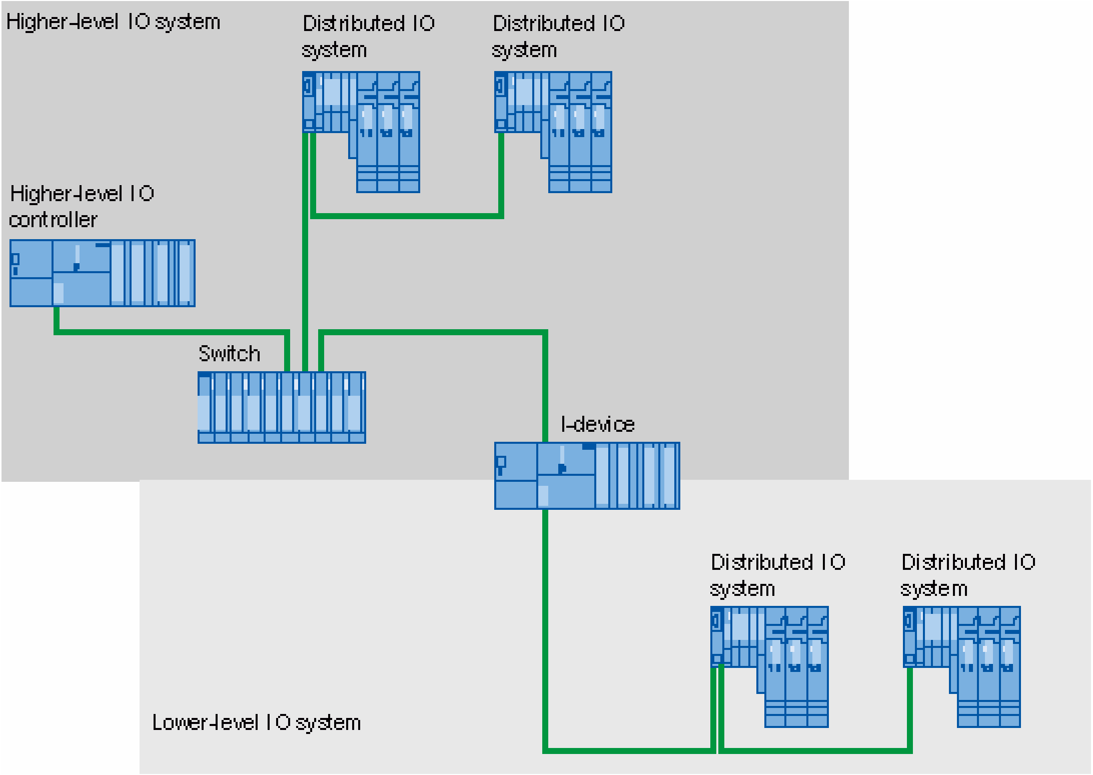
I-device with one port

Connect an I-device with only one port to a switch that is uncoupled from the higher-level IO system. Connect the lower-level IO system to another port of the switch as shown in the following figure.



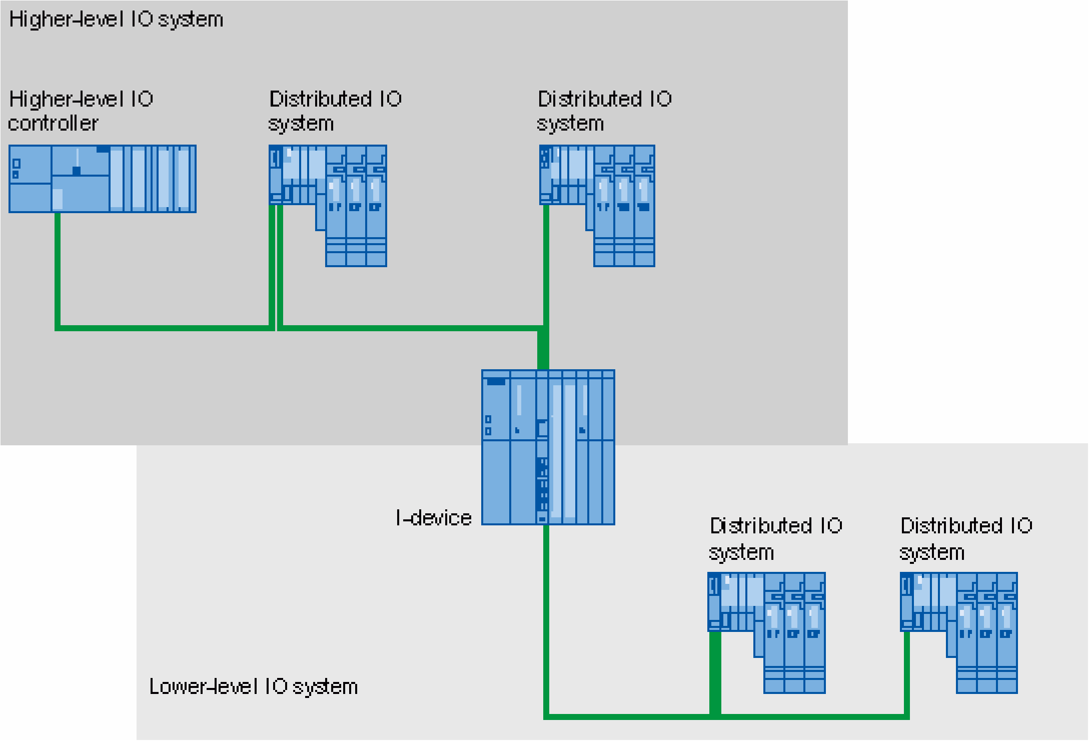
I-device with two ports

With an I-device with two ports, connect one port uncoupled from the higher-level IO system to the port of the switch. Use the second port for the lower-level IO system as shown in the following figure.



I-device with three or more ports

If you have an I-device with three or more ports, connect the I-device to one or both ports to the higher-level IO system in a linear bus topology. Connect the third port to the lower-level IO system uncoupled from the bus topology as shown in the following figure.



4.6.3.3 Constraints on the Use of I-Devices

Constraints

When using I-devices, certain constraints must be taken into account in certain situations.

Bandwidth

The number of addresses of the configured transfer areas is reflected in the usable bandwidth of the I-device:

Bandwidth of the transfer areas + bandwidth of the lower-level IO system = total bandwidth used on the I-device

If the address space of the transfer areas is too large, there is not enough bandwidth for the lower-level IO system to achieve fast update times.

Tip: Keep the address space of the transfer area as small as possible.

Rules for RT and IRT communication

IO systems with an I-device are suitable for setting up real-time applications with RT and IRT communication. Here, the following rules must be kept to:

- Both the higher-level and the lower-level IO system support RT communication. You can use RT communication for both IO systems at the same time.
- IRT communication can be combined with RT communication. IRT communication can, however, not take place in both IO systems at the same time.

PROFINET CBA and I-device

When using PROFINET CBA and I-device at the same time, make sure that no communication allocation was reserved for PROFINET CBA (communication allocation PROFINET CBA = 0.0 %) in the Properties dialog of the interface in the "PROFINET" tab.

In conjunction with an I-device, PROFINET CBA can only be used with acyclic interconnections.

Isochronous mode

You cannot operate I-devices in isochronous mode on the higher-level IO controller.

Data access

The higher-level IO controller can access different transfer areas:

- Application transfer areas if the I-device is in RUN.
- IO transfer areas of the type output if the I-device is in RUN and the I/O exists and is available.
- IO transfer areas of the input, if the I/O exists and is available, the status of the I-device CPU is then irrelevant (RUN / STOP).

The I-device CPU can only access your input application transfer area. The higher-level IO controller must be in RUN.

Characteristics of the I/O of IO transfer areas

The I/O of the I-device available to the higher-level IO controller as an IO transfer area behaves as follows:

- Output: If the I-device CPU is in RUN, the I/O exists and is available, the value written by the higher-level controller to the IO transfer area is output, otherwise 0.

  Caution

  if substitute values were configured, 0 is nevertheless output! Remember this when configuring substitute values to ensure a safe system status.
- Input: If the I/O exists and is available, the higher-level controller accepts the value of the I/O from the IO transfer area. If the I/O does not exist or is not available, it outputs an IO access error on the higher-level IO controller.

IP address parameters and device name

Just like every other IO device, an I-device also requires IP address parameters / device name to be able to communicate via PROFINET. IP address parameters consist of three parts, the IP address itself, the subnet mask and the address of the router (gateway).

The IP address parameters / device name for an I-device can be assigned in two ways:

- IP address parameters / device name via project: The IP address parameters / device name are permanently assigned during configuration (in the I-device project) in STEP 7. This is the standard method.
- Use different method to obtain IP address parameters / device name:
  - IP address parameters / device name via DCP: The IP address parameters / device name are assigned by DCP (Discovery and Configuration Protocol). This can happen in two ways:

    1. Using a setup tool such as PST or STEP 7 (via accessible nodes).

    2. By the higher-level controller.
  - IP address parameters / device name via user program: The IP address parameters are assigned in the user program of the I-device CPU.

4.7 Configuring IRT Communication

4.7.1 Introduction: Isochronous Real Time Ethernet

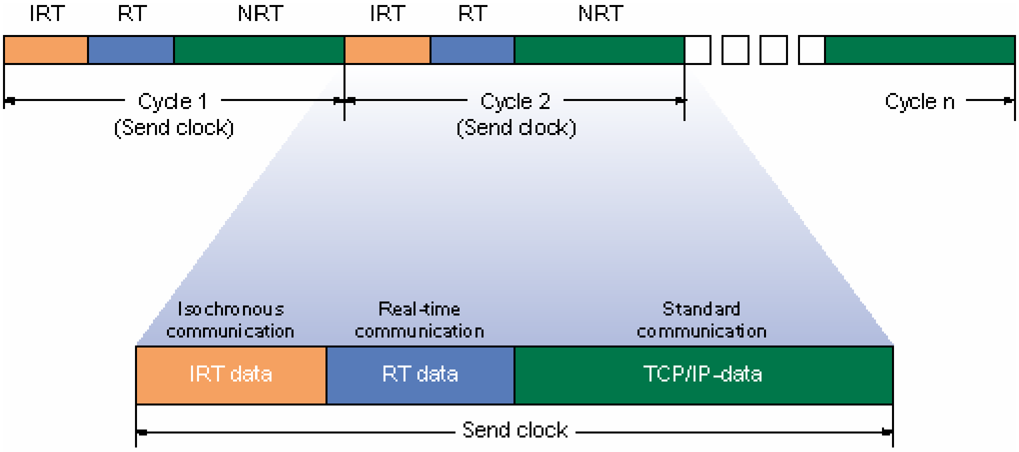
IRT is a transmission technique in which the PROFINET devices are synchronized with extreme accuracy.

A sync master provides the clock, the sync slaves synchronize with this clock. Both an IO controller and an IO device can adopt the role of sync master.

Sync master and sync slaves are always nodes in a sync domain. Within the sync domain, bandwidth is reserved for IRT communication. Real-time and non-real-time communication (TCP/IP communication) is possible outside the reserved bandwidth.

Time segments for the communications cycle

The communications cycle is divided into three time segments:

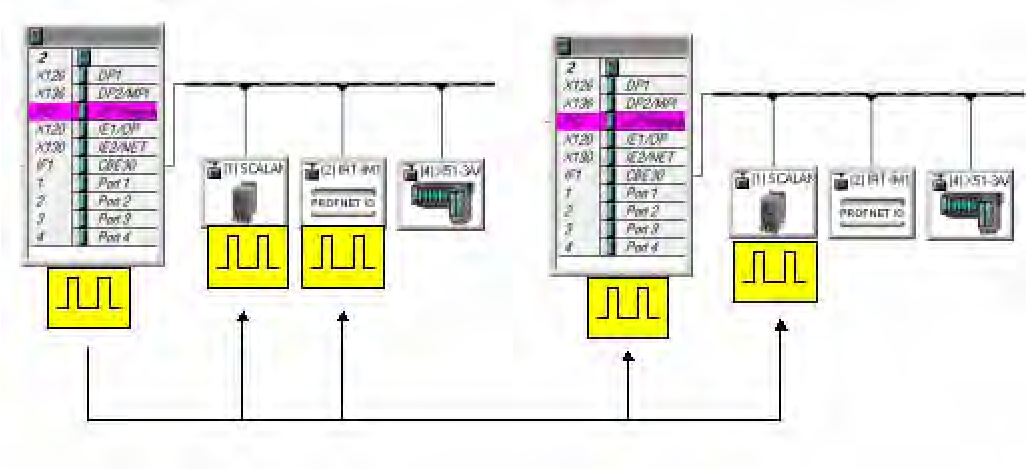


- IRT data (isochronous communication)
  You can reserve this time segment in certain steps regardless of the send clock. During this time segment, only IRT date is transferred. For this time segment, you have the choice between "high performance" for the fastest update times (this requires configuration of the topology) and "high flexibility" for simple configuration and expansion of the plant.
- RT data (real-time communication)
  The cyclic RT frames are transferred in this time segment. The Ethernet frames are prioritized compared with "normal" TCP/IP frames (priority 6 of 7).
- TCP/IP data (standard communication)
  Standard communication (TCP/IP, etc.) is handled in the remaining interval of the communications cycle.

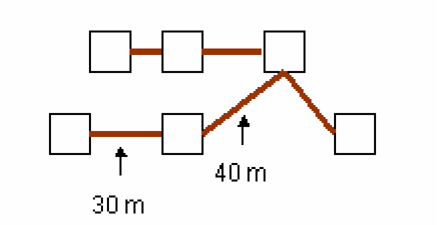
4.7.2 Overview: Basics of Configuring IRT

If you want to expand a plant with PROFINET IO by adding IRT (for example for motion control applications), this involves the following three steps:

1. Configure stations with PROFINET IO controllers and PROFINET IO devices. PROFINET devices that you need for your applications must support IRT.
2. Now specify what is synchronized with what. To do this, you must configure a sync domain with a sync master with several sync slaves.



3. Only for IRT with the IRT option "high performance": Configure the topology; in other words, specify how the ports for the individual devices are interconnected.



4. Download the configuration to the devices.

This basic description is extremely simplified. For detailed instructions, refer to the information available through the links listed below.

4.7.3 Creating a PROFINET IO Configuration for IRT

A PROFINET IO configuration is required to configure IRT; that is, one or more stations have to be configured with an IO controller and IO devices.

IRT communication is limited to components designed for it. In the hardware catalog, you will recognize such components by a reference in the info text.

Sync domain is created automatically during configuration

A sync domain is required for the synchronization of PROFINET IO devices. The sync domain ensures that all nodes within it can communicate in isochronous mode.

Simply by configuring a PROFINET IO system with components that support IRT, you have already laid the foundation for functional IRT communication.

The following sections describe what happens in the background while you are configuring an IO system.

Requirement

You have not yet configured a PROFINET IO device on Ethernet.

Sequence

1. You configure an IO controller and attach it to an Ethernet subnet.

   Reaction:

   The IO controller is assigned to the "sync domain default" of the Ethernet subnet. The "sync domain default" always exists. The IO controller operates unsynchronized.
2. You add an IO device to the IO system of the IO controller.

   Reaction:

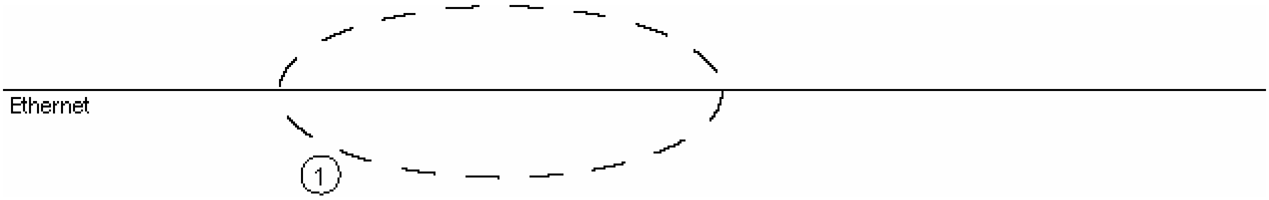
   - The IO device is automatically assigned to the sync domain of the IO controller.
   - The IO device is assigned to the RT class of the IO controller, i.e., it is assigned the RT class "RT" (not synchronized).

If you have more complex configurations, for example several sync domains on one Ethernet subnet, or if you want to change defaults (send clock, RT class, IRT proportion of the send clock, or synchronization role etc.), you must adapt these settings in the "Sync Domain" dialog.

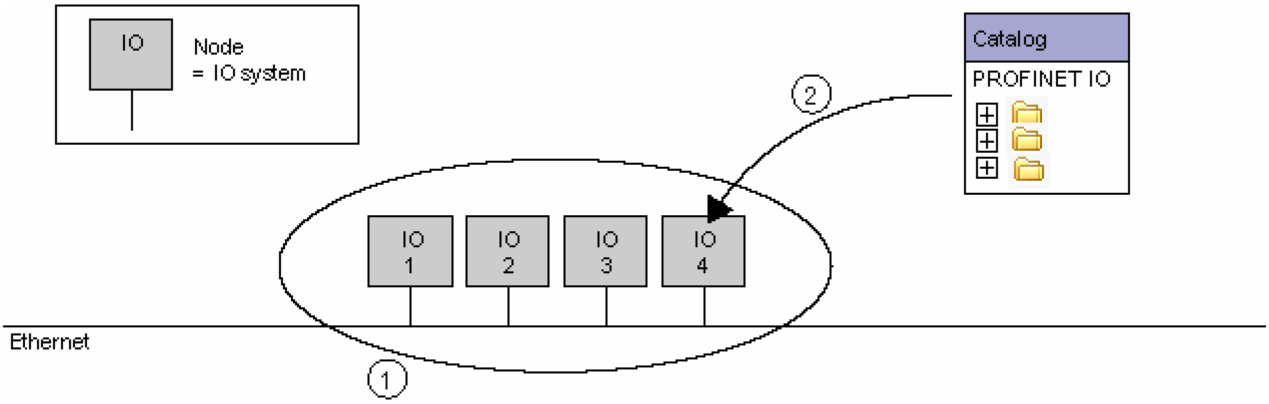
4.7.4 Working with Sync Domains

When you have created an Ethernet subnet, a special sync domain is created: All PROFINET devices that you configure for the Ethernet subnet automatically belong to this sync domain. This sync domain is called "sync domain default"; it cannot be deleted.

The following schematics illustrate the relationships.



(1) "sync domain default"; always exists



(1) "sync domain default"; always exists

(2) Newly configured PROFINET IO systems are initially in the "sync domain default".

4.7.5 Creating a New Sync Domain

All components involved in IRT communication must belong to a sync domain.

A sync domain is a group of PROFINET devices synchronized to a common clock pulse. One device (and one only) takes on the role of the sync master (initiator of the clock pulse), all other devices take on the role of sync slave.

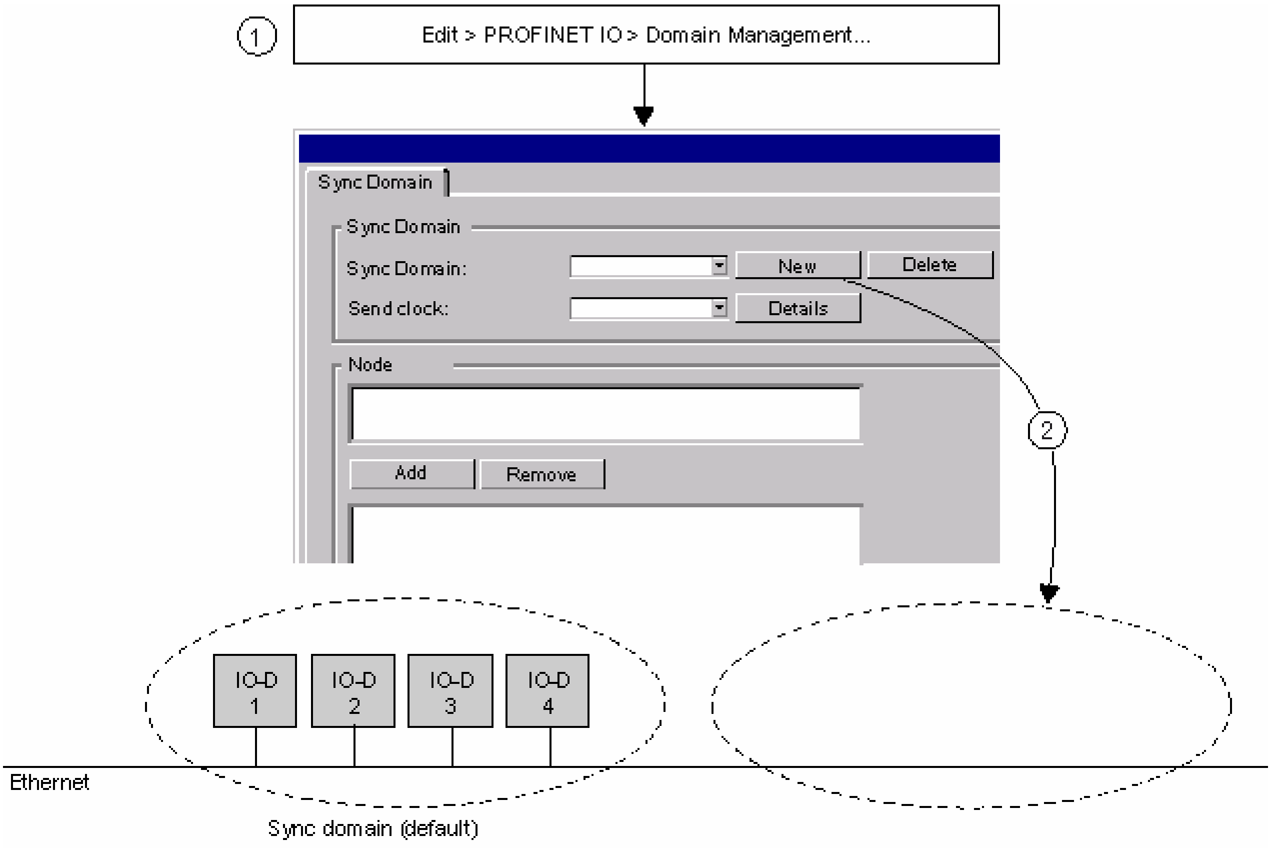
You always edit sync domains in a central dialog that you can open with the **Edit > PROFINET IO > Domain Management** menu command or by right-clicking on the PROFINET IO system and selecting **PROFINET IO Domain Management...**

Requirement

You have created a project in which stations with PROFINET IO systems are already configured. The following description assumes that there are already IO systems in the "sync domain default".

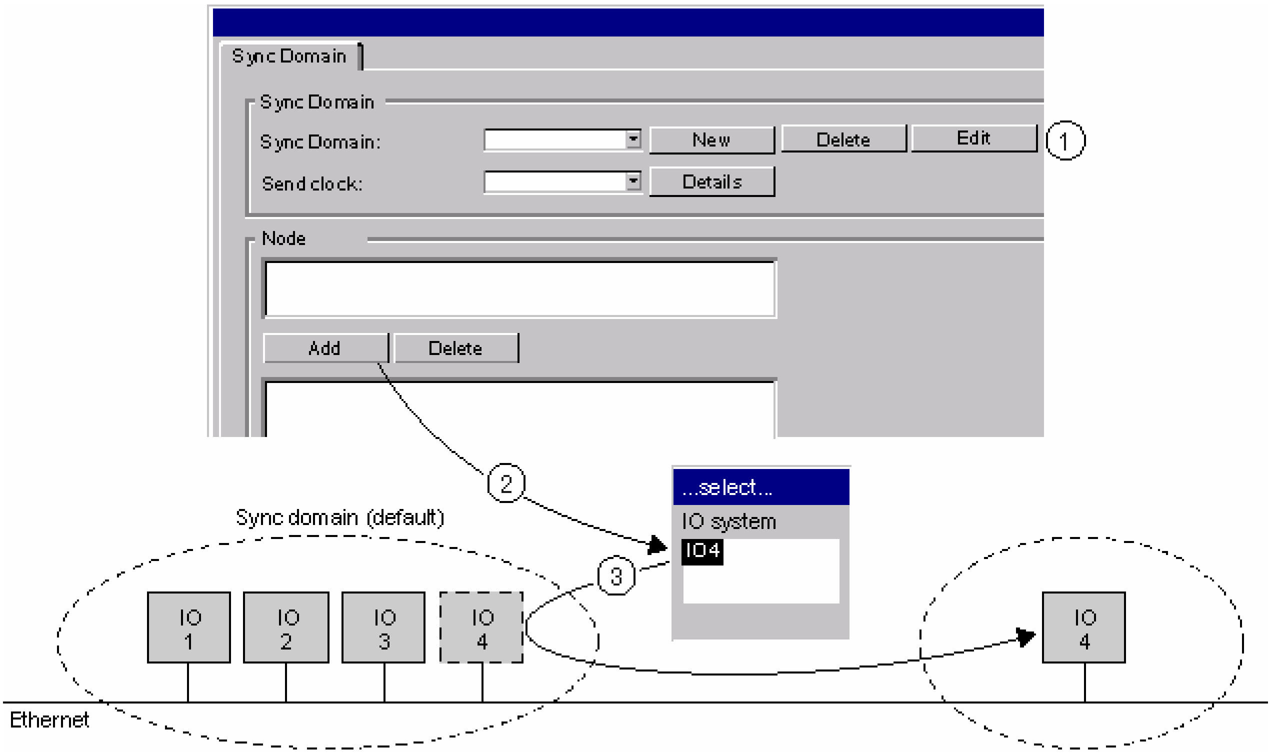
Creating a new sync domain

1. Select the **Edit > PROFINET IO > Domain Management** menu command.
   A dialog tab for creating and editing sync domains opens.
2. Click the "New" button beside the "Sync domain" box.
   A new sync domain is created.



Assigning names for a sync domain and adding an I/O system

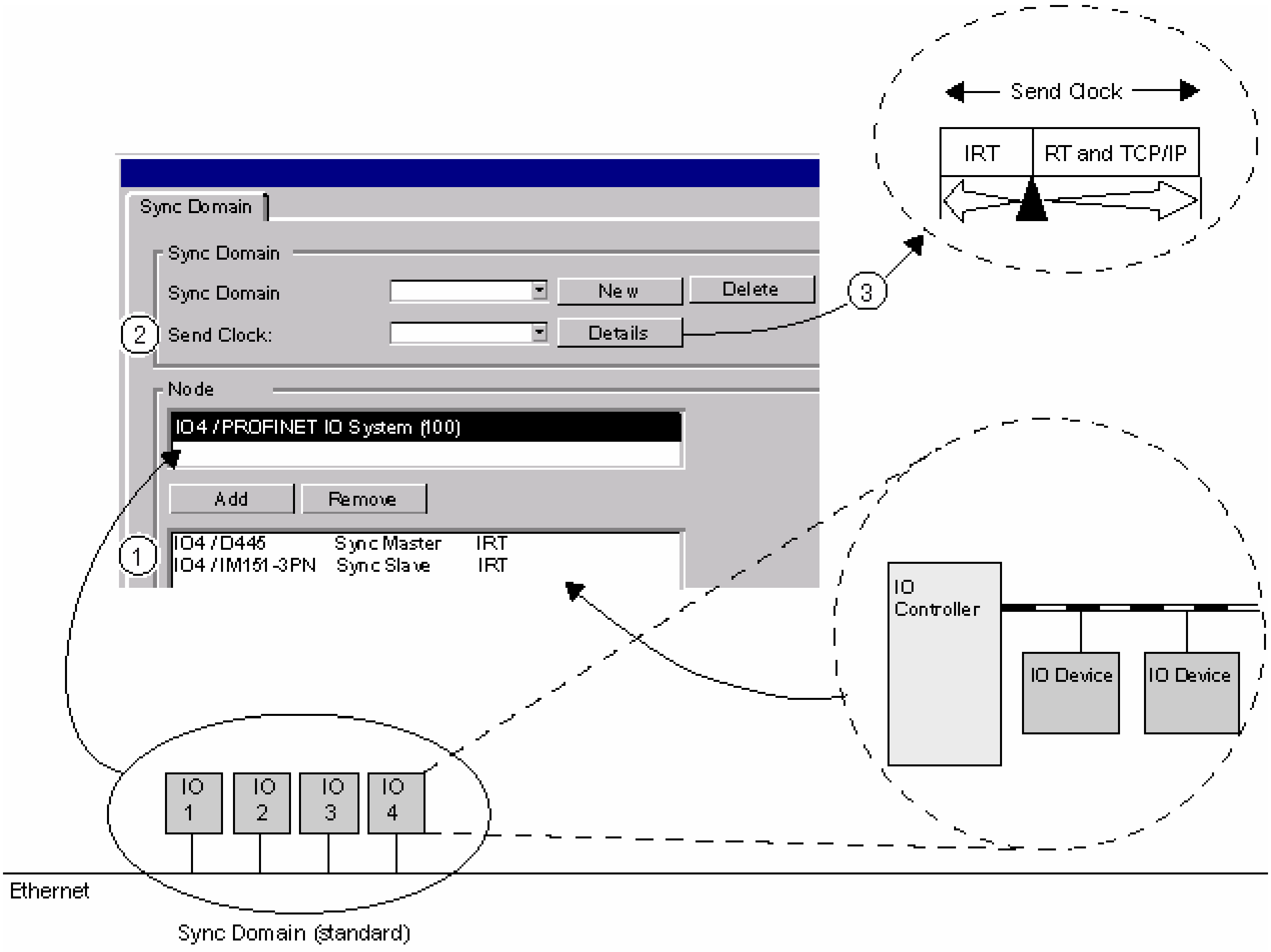
1. To change the default name of the sync domain, click on the "Edit" button.
2. Select a suitable name for the sync domain. The name of the sync domain must match the DNS conventions. These are the same rules that need to be taken into account when assigning names of PROFINET devices.
   The boxes below "Node" are initially empty.
3. Click the "Add" button
4. From the list of IO systems in the "sync domain default", select the IO systems you want to include in the new sync domain (multiple selection possible).
   Acknowledge the settings with "OK".
5. The added IO system is no longer in the "sync domain default" but in the new sync domain.



Selecting synchronization properties

The figure shows what is behind the individual lines of the two "Nodes" boxes: the IO systems in the upper boxes, the configuration of the selected IO system in the lower box.

1. Select the required synchronization role and the suitable RT class for each device for each node by double-clicking on a device in the lower "Node" box.
2. Select the required send clock in the "Send clock" box. The update times of the devices are always multiples of the send clock. The send clock, also referred to as the "synchronization clock", applies to all synchronized devices of the sync domain.
3. Using the "Details" button, select the allocations for IRT communication and RT or TCP/IP communication.



IO4 / PROFINET IO System (100)

IO4 / D445	Sync Master	IRT
IO4 / IM151-3PN	Sync Slave	IRT

Difference between the "configured" sync domain and the "runtime sync domain"

The sync domain configured here may also contain devices of the RT class, "RT". This does not mean, however, that these devices are synchronized. They are only shown here to provide a complete overview. During runtime, there are no devices of the "RT" RT class in a sync domain.

Rules

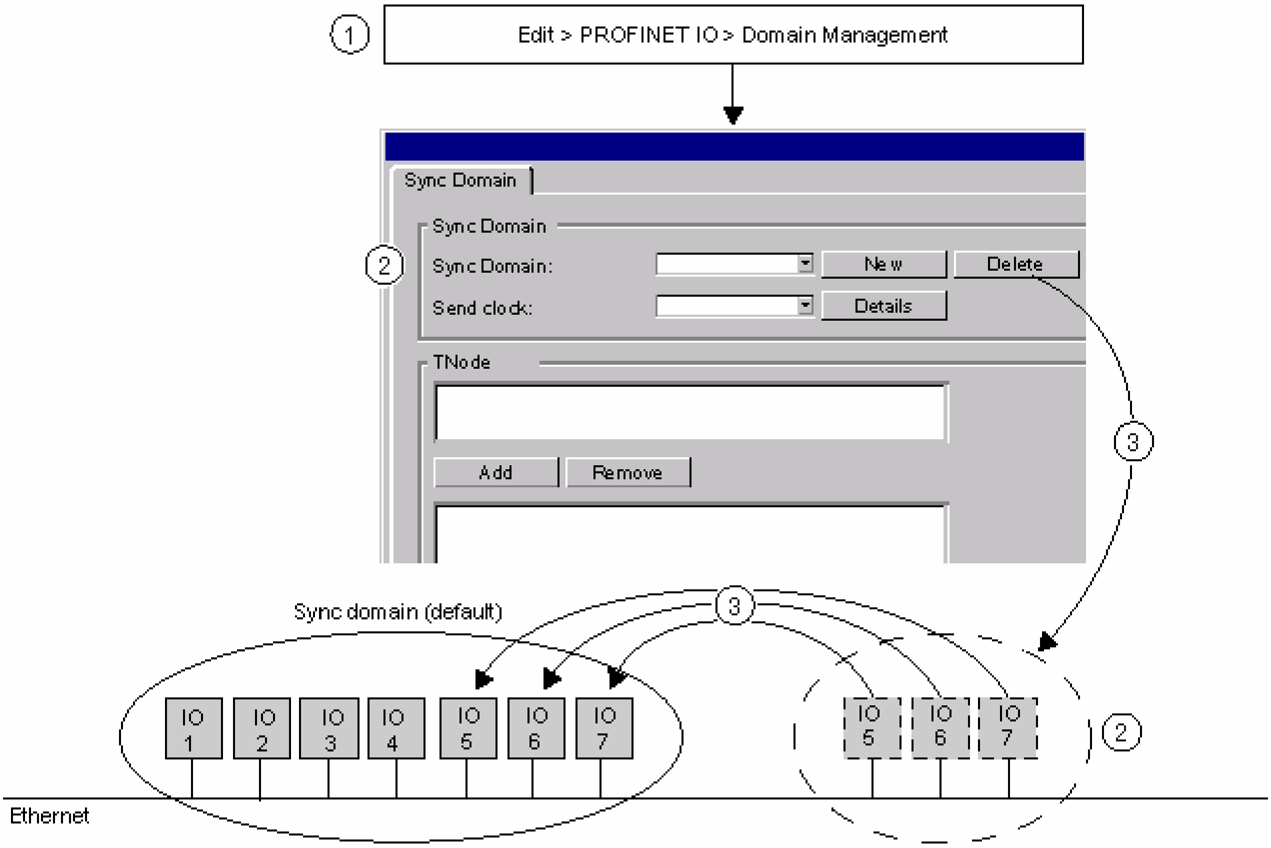
- Only devices with IRT classes may exist in a sync domain
- There can only be one sync master in a sync domain.

4.7.6 Deleting Sync Domains and Removing Nodes

Deleting a sync domain

To delete a sync domain, follow the steps outlined below:

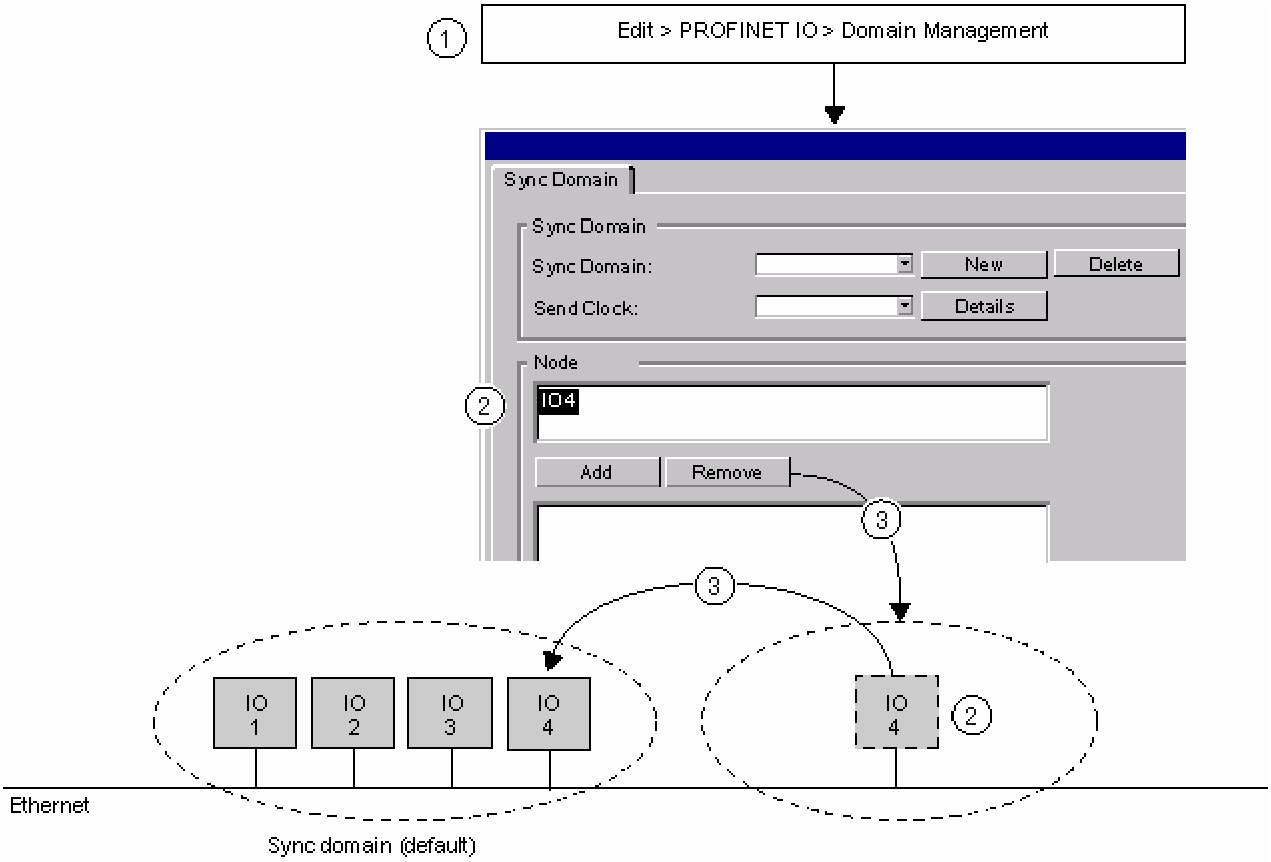
1. Select the **Edit > PROFINET IO > Domain Management** menu command.
   A tab opens in which you can create and edit sync domains.
2. Select the sync domain you want to delete.
3. Click the "Delete" button beside the "Sync Domain" box.
   The nodes of the deleted sync domain return to the "sync domain default".



Removing a node from a sync domain

To remove a node from a sync domain, follow the steps outlined below:

1. Select the **Edit > PROFINET IO > Domain Management** menu command.
   A tab opens in which you can create and edit sync domains.
2. In the "Station/IO system" box, select the IO system you want to remove from the sync domain.
3. Click the "Remove" button.
   The removed node returns to the "sync domain default".



4.7.7 Configuring a Redundant Sync Master

You can configure a redundant sync master for PROFINET IO devices in IRT mode.

A redundant sync master belongs to the same sync domain as the "primary" sync master. It prevents the failure of the communication in the sync domain if the primary sync master fails.

Concept

Both the sync master and the redundant sync master send sync frames. A sync slave synchronizes itself to the sync frame of the sync master. If the sync master fails, a sync slave then automatically synchronizes itself to the sync frames of the redundant sync master.

Requirement

There are only IRT devices in the sync domain.

Rules

You can only assign the role of the redundant sync master to one device in the sync domain.

Procedure

You can assign the role of the redundant sync master when you create or change a sync domain.

Alternative method:

Double-click on the "Interface" line of the PROFINET IO device and select "Redundant sync master" as the synchronization role.

4.7.8 Setting the Send Clock and Update Times

Send clock

The send clock is the shortest possible update time and is set in the following dialogs:

- If there is at least one synchronized PROFINET device (IRT is configured) in the IO system, you need to select the send clock in the sync domain dialog (**Edit > PROFINET IO > Domain Management** menu command).
  The send clock can then no longer be modified in the properties dialog of the PROFINET IO system or in the properties of the interfaces for the IO controller.
- If the IO system contains only unsynchronized PROFINET devices, then you can set the send clock
  - in the properties of the interfaces for the IO controller.
  - in the properties of the PROFINET IO system in the "Update Time" tab

Update times

Update times are specified during the configuration of PROFINET IO systems.

1. Select the "Update Time" tab in the "Properties PROFINET IO System" dialog.
2. Double-click on a row representing a PROFINET device.
   Here you set the update time.
3. Acknowledge the settings with "OK".

Note

The update time cannot be modified at this point (display "read-only") if the interface of the relevant device is set so that the update time is assigned automatically by STEP 7.

Ti and To Times in IRT

The mechanism of clocked reading and output of I/O signals in PROFINET with IRT works in much the same way as in clocked PROFIBUS DP. In PROFINET, the Ti and To times are also derived from the constant bus cycle time and the IO devices and the modules ensure that the signals are read in and output at constant intervals. Only the mechanism is different since there is no "global control" providing the timing in PROFINET. The Ti and To times that can be set are device-dependent. They are defined by the GSD file for the device.

The diagrams illustrating the sequence of clocked data exchange in PROFIBUS DP apply analogously to PROFINET IO.

4.7.9 Configuring the Topology

IRT communication with the IRT option "high performance" is possible only with topology configuration.

When you specify the cable properties between the ports of the switches, you have two options:

- The Topology Editor provides you with an overview of all the ports in the project and you interconnect the ports centrally. You can view the PROFINET modules and their ports in the form of a table or as a graphic, create and edit interconnections, display diagnostic information from connected devices and run an online-offline comparison.
  You start the Topology Editor by selecting an active PROFIBUS module in HW Config or NetPro and then selecting the menu command **Edit > PROFINET IO > Topology**.
- As an alternative, you can select a partner port using the properties of a port. This decides the cable between two ports and you can edit the properties of this cable.
  You open the dialog by selecting a port in HW Config and then selecting the menu command **Edit > Object Properties** or double-clicking on the port.
  Then select the "Topology" tab in the "Properties Port ..." dialog box.

4.7.10 Configuring Direct Data Exchange

With direct data exchange, data areas can be exchanged cyclically between two IO controllers that are in a sync domain and take part in IRT communication.

Principle

In much the same way as in PROFIBUS DP, data areas (I/O areas) are configured by CPUs via which the CPUs send and receive.

The CPU with a data area that is configured as the "sender" sends data via outputs.

The CPU with a data area that is configured as the "receiver" receives data via inputs.

Recommendation

We recommend that the send areas for all IO controllers are configured first, and then the receive areas.

Requirements

You have already configured the stations involved in direct data exchange.

For the sender, you can configure an update time that is a multiple of the send clock.

Configuring the sender

1. Double-click on a row in the configuration table of HW Config that represents the interface of the PROFINET interface (for example, X1)

   The properties dialog of the IO controller opens.
2. Select the "Sender" tab.
3. Click the "New" button.
4. Enter the start address and length of the address area to be used for sending in the Properties dialog of the sender.
   Add a comment for the data area so that you can later identify data sent over this area.
5. Acknowledge the settings with "OK".
6. Repeat steps 3 to 5 for other send areas.
7. If required, you can change the default diagnostic address for the send areas.
   For a communication relation containing an IO controller as sender for direct data exchange, one diagnostic address must be assigned (and one only).

Configuring the Recipient

1. Double-click on a row in the configuration table of HW Config that represents the IO controller (for example, X1)

   The properties dialog of the IO controller opens.
2. Select the "Receiver" tab.
3. Click the "New" button.
4. Click the "Assign Sender" button in the "Receiver Properties" dialog.
5. In the "Assign Sender" dialog, select the data area of the required station that will be received by the local CPU.
6. Acknowledge your selection with "OK".
7. Enter the start address and length of the address area for reception in the Properties dialog of the receiver.

   The length of the address area should not be changed because it is automatically adapted to the length of the send area. The configuration can only be compiled when the send and receive areas have the identical length!
8. Repeat steps 3 through 7 for other receive areas.
9. A diagnostic address is reserved for each assigned sender with which the receiver can recognize the failure of the sender.
   Click the "Diagnostic Addresses" button if you want to edit these addresses.

4.7.11 Downloading an IRT Configuration

The configuration data must be downloaded to all devices that are involved in IRT communication.

Before downloading, first run a consistency check and eliminate any existing errors.

Recommendation

Sync domains are not limited to one station. Download the configuration from the network configuration (NetPro).

To do this, select the Ethernet subnet and then select the menu command **PLC > Download to Current Project > Stations on the Subnet**.

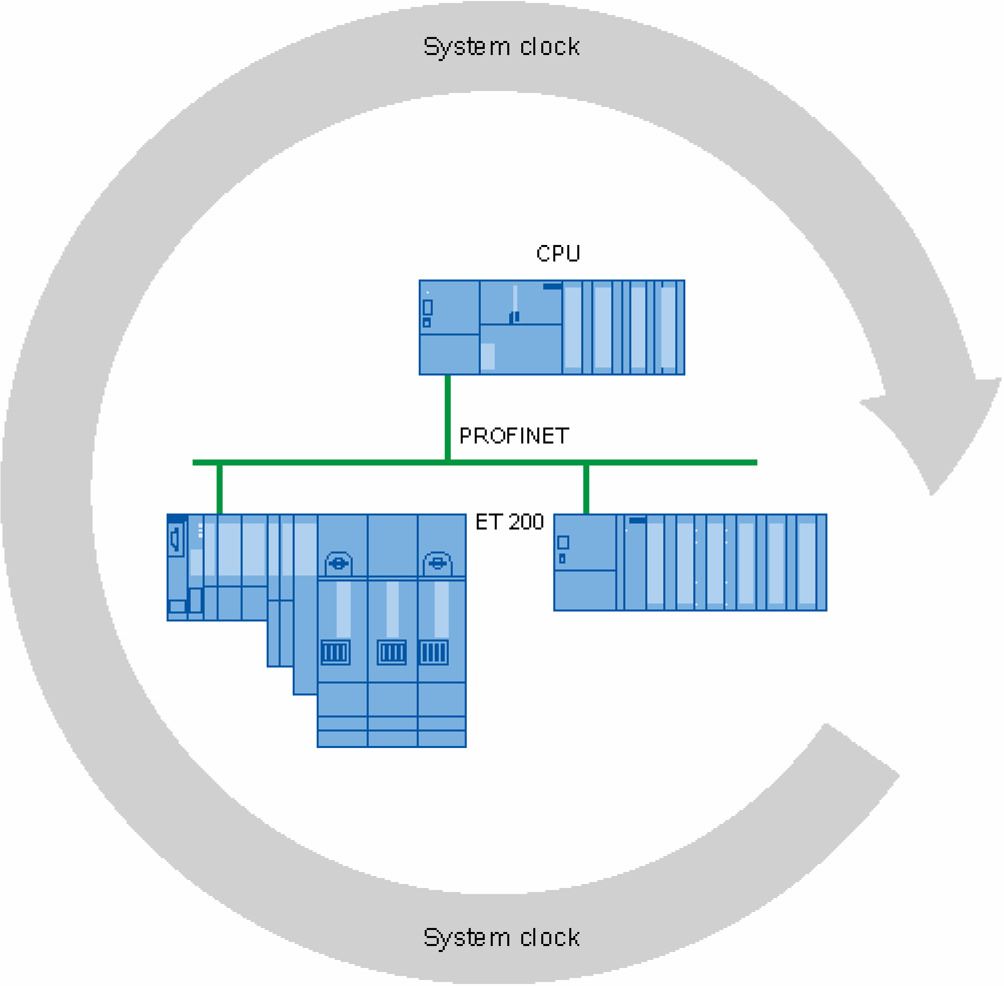
4.8 Isochronous Mode

4.8.1 What is Isochronous Mode?

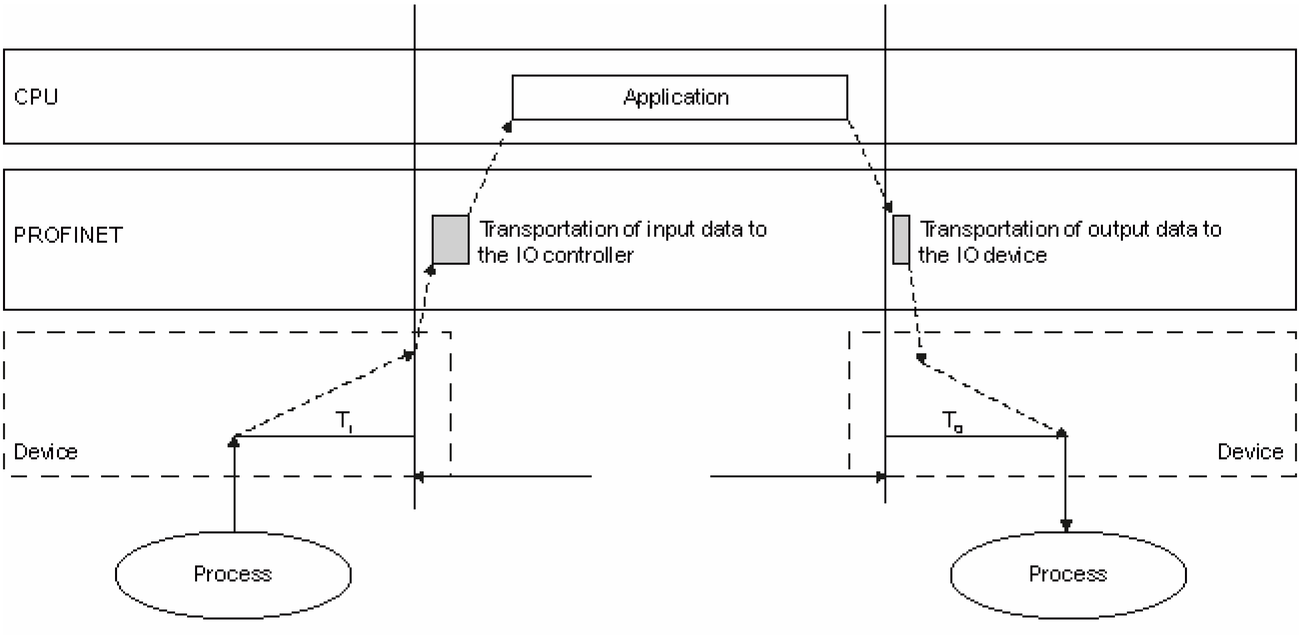
Why isochronous mode?

If public transport traveled as fast as it possibly can and reduced stopping times to an absolute minimum, passengers would very often only see the vehicles disappearing into the distance. The overall travel time is, however, decided by the train, bus or underground clock, because well adjusted timing is essential to a good service. This also applies to distributed automation engineering. Not only fast cycles but also the adaptation and synchronization of the individual cycles result in optimum throughput.

Just-in-time



The fast and reliable reaction time of clock synchronization is based on the fact that all data is made available just-in-time. The constant PROFINET IO cycle beats the time here.


CPU
Application
PROFINET
Transportation of input data to the IO controller
Transportation of output data to the IO device
T_i
T_o
Device
Device
Process
Process

T_DC data cycle

Ti time for reading in

To time for outputting the output data

To ensure that all input data is ready for transportation via the PROFINET IO line when the next PROFINET IO cycle begins, the IO read cycle has a lead time Ti so that it starts earlier. Ti is the "flash gun" for all inputs. This Ti is necessary to compensate for analog to digital conversion, backplane bus times etc. We recommend that the lead time Ti is set automatically by STEP 7. Otherwise, it can be set by the user in the configuration.

The PROFINET IO line transports the input data to the IO controller. The synchronous cycle interrupt OB (OB61, OB62, OB63 or OB64) is called. The user program in the synchronous cycle interrupt OB decides the process reaction and provides the output data in time for the start of the next data cycle. The length of the data cycle is always configured by the user.

To is the compensation from the backplane bus and the digital to analog conversion within the device. To is the "flash gun" for all outputs. We recommend that To is set automatically by STEP 7. Otherwise, it can be set by the user in the configuration.

Advantages of isochronous mode

At the times Ti and To, a snapshot of the values is made throughout the plant. This snapshot allows the values to acquired simultaneously so that they can be processed with consistency.

Advantages of using isochronous mode:

- When acquisition of measured values needs to be synchronized, movements need to be coordinated and process reactions need to be defined and take place simultaneously
- Simultaneous acquisition of signals for control processes, measurement and motion control tasks regardless of location

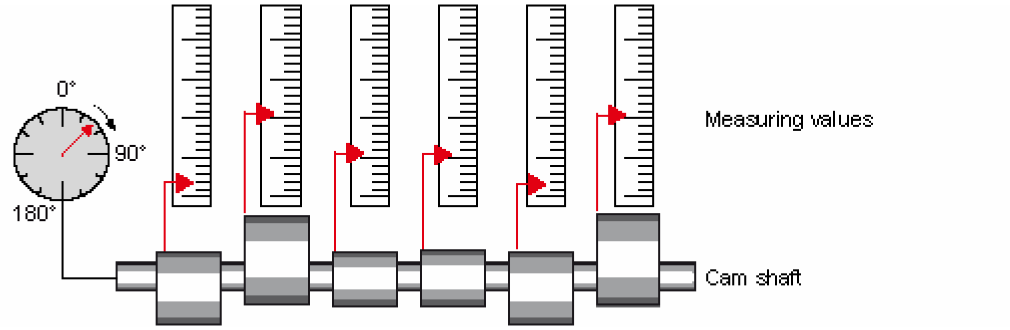
Isochronous mode and non isochronous mode distributed I/O

It is possible to combine isochronous mode distributed I/O with non isochronous mode distributed I/O on one IO controller.

4.8.2 Applications of Isochronous Mode

Example: Measuring at several points in isochronous mode

In the production of cam shafts, these must be measured precisely to ensure quality assurance.



Working sequence with isochronous mode

By using the system property "isochronous mode" and the associated simultaneous measured value acquisition, measurement can be performed continuously and the time required for measurement is reduced. This leads to the following working sequence:

- Turn the cam shaft continuously
- During the continuous turning, measure the positions and cam excursion synchronously
- Process the next cam shaft

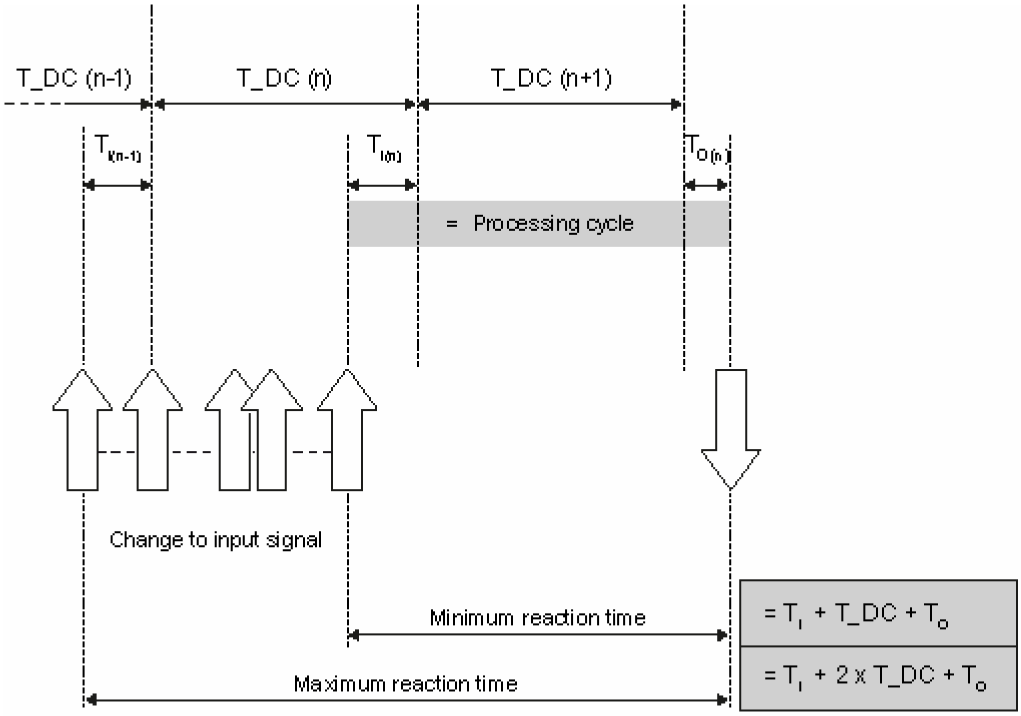
This means that with a single turn of the cam shaft, all the positions of the cam shaft and the corresponding measured values (red) are measured synchronously. The machine clock increases with same or improved measuring accuracy.

4.8.3 How Does Isochronous Mode Work?

The essential basis for synchronized processing cycles is the constant PROFINET IO bus cycle time. The system property "isochronous mode" links the SIMATIC automation solution to the constant PROFINET IO bus cycle time.

This means:

- The data read in at Ti does not change during the data cycle. The time Ti moves the reading in of the values back by a fixed time.
- The user program for processing the IO data is synchronized with the data cycle by the synchronous cycle interrupt OBs OB61 to OB64.
- The data output at To does not change during the data cycle. The time To moves the output of the values back by a fixed time.
- All input and output data is transferred consistently. This means that all data of the process image belongs together both logically and in terms of time.



T_DC data cycle

Ti Time for reading in the input data

To time for outputting the output data

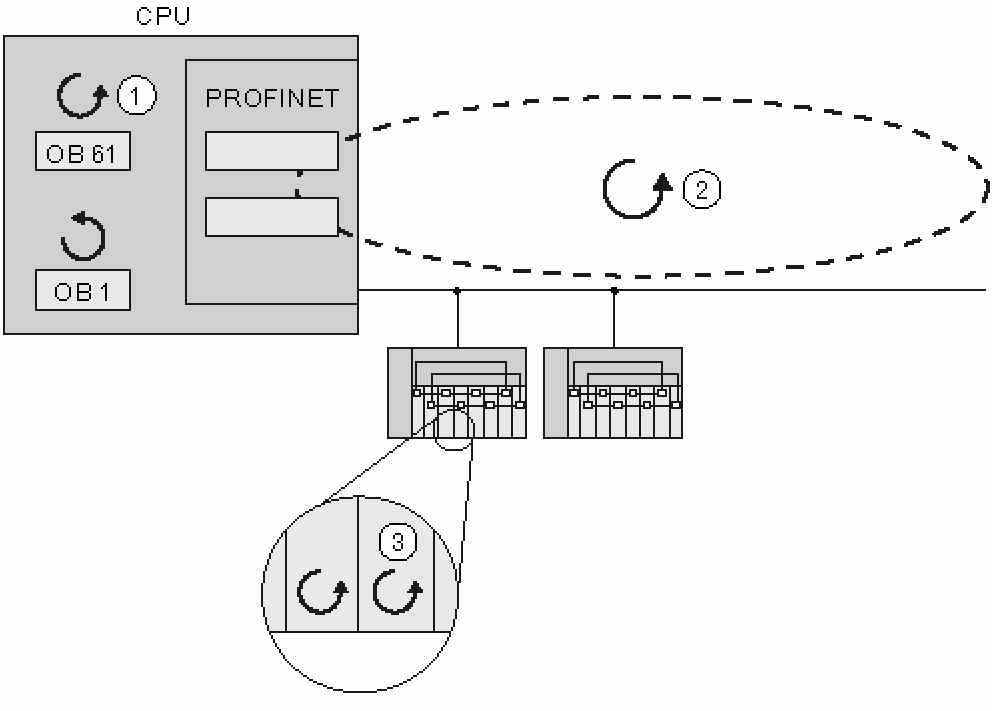
By synchronizing the individual cycles it is possible to read the input data in data cycle "n-1", to transfer and process the data in data cycle "n" and to transfer the calculated output data at the start of data cycle "n+1" and switch it to the "terminals". This results in an actual reaction time of minimum "Ti +T_DC +To" to maximum "Ti + 2xT_DC + To". 2xT_DC results from the fact that the change to the input value by sampling in the data cycle has a window. The output data is always set at specific time.

With the system property "isochronous mode", the system throughput times of the SIMATIC controller are constant, the SIMATIC controller is strictly deterministic via the PROFINET IO line.

4.8.4 Synchronized Processing Cycles

Process reaction with isochronous mode

Process reactions are represented as follows:



(1) Application cycle on the CPU

(2) Transfer cycle PROFINET IO

(3) Conversion cycles on the IO devices

In the example above, the response with isochronous mode is represented based on a sample structure with one IO controller and two IO devices. The process data, transfer cycle via PROFINET IO and user program are synchronized with each other to achieve maximum determinism. The input and output data of distributed I/O in the plant is acquired simultaneously and output simultaneously. The constant PROFINET IO cycle sets the rhythm here.

The OB1 cycle or cyclic interrupt cycles are not included in this clock cycle.

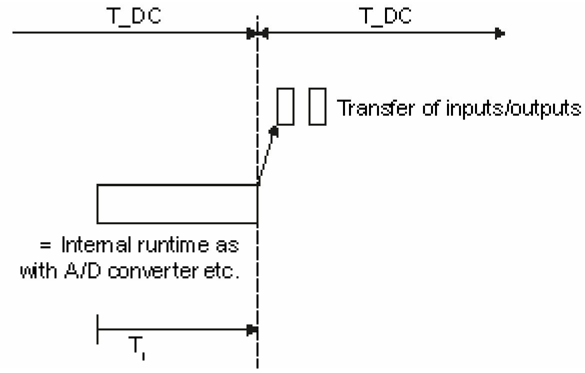
Synchronization with the user program is achieved with the synchronous cycle interrupt OB61 (or OB61 to OB64).

Generally all inputs are read in at one time ("flash gun" via the inputs), processed and output to the outputs at one time.

4.8.4.1 The Ti Value

Effect of Ti

The effect of Ti is illustrated by the following figure:


T_DC
T_DC
Transfer of inputs/outputs
= Internal runtime as with A/D converter etc.
T_I

Sequence

So that a consistent state of the inputs can be transferred to the IO controller at the start of the new system clock cycle, the reading in must be brought forward by the time Ti. The time Ti for a particular input module includes at least the signal conditioning and conversion time on the electronics modules and the time for transfer to the interface module of the inputs on the IO device backplane bus.

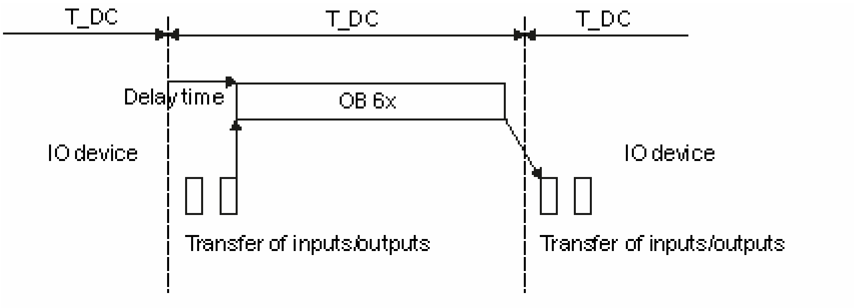
In the plant, the values are read in at the same time because the Ti of the various input modules is set to the same value and this value is greater than or equal to the highest minimum Ti of all input modules. With the default value, STEP 7 ensures that a common Ti and To are set and that the times are kept as short as possible.

If certain modules have long conversion times, they will influence these values as a whole. The only option is then for the user to set a longer time for these modules and leave the shorter time for the others. This would be the case, if they are not all set to the same value. The overall result must then be taken into account in the control task.

4.8.4.2 The User Program OB6x

Effect

The effect is illustrated by the following figure:



T_DC   system clock

Ti   Time for reading in the input data

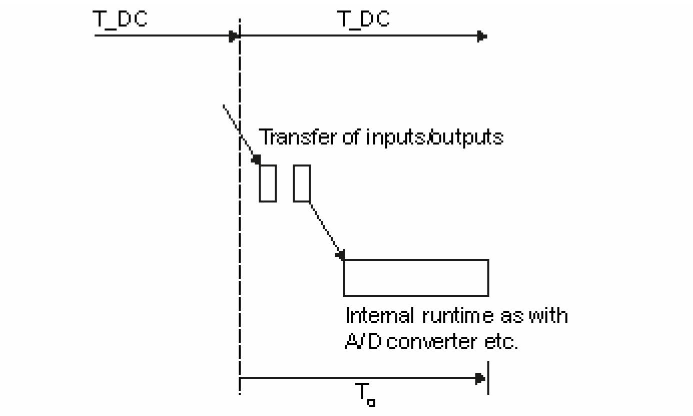
Sequence

The delay time is calculated by STEP 7 as a default value. This compensates the transfer time of the inputs from the bus. OB6x is linked to the system clock cycle. You can, however, also correct the delay time (see Configuration).

4.8.4.3 The To Value

Effect of To

The effect is illustrated by the following figure:



T_DC	system clock
To	time for outputting the output data

Sequence

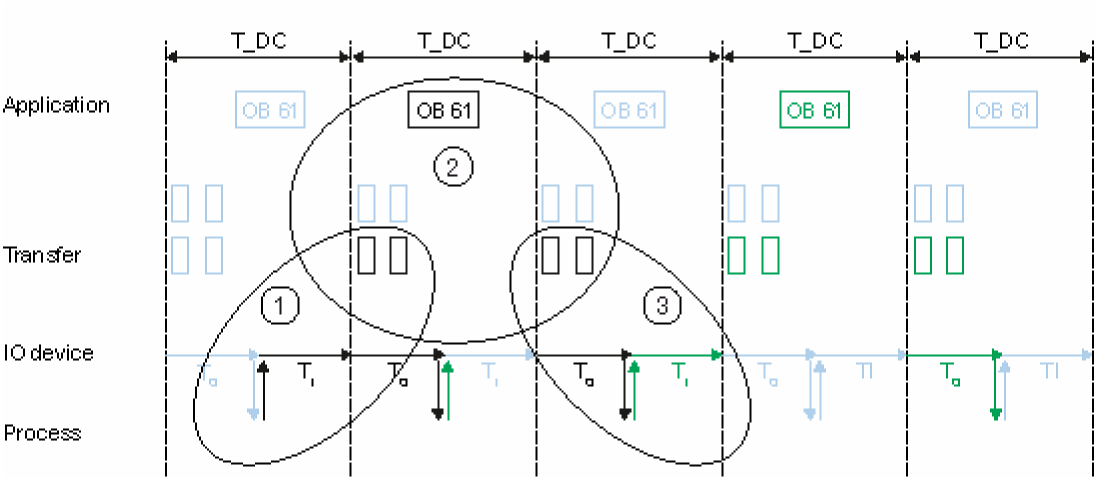
To ensure that a consistent status of the outputs can be transferred to the process at the start of the new system clock cycle, the data is output at the time To following the last clock beat. The time To includes the minimum time for a specific output module for transferring from the IO controller to the IO device and for transferring the outputs from the interface module to the electronics module on the IO device.

In the plant, the values are written at the same time because the To of the various output modules is set to the same value and this value is greater than or equal to the highest minimum To of all output modules. With the default value, STEP 7 ensures that a common Ti and To are set and that the times are kept as short as possible.

4.8.4.4 Isochronous Mode over Several System Clock Cycles

Effect

This figure combines the information from the previous figures.



T_DC system clock

(1) See The Ti Value

(2) See The User Program OB6x

(3) See The To Value

Here, you can see that Ti, OB61 and To occur one after the other. You can also see that Ti, OB61 and To occur parallel to each other in one system clock cycle.

4.8.5 Engineering

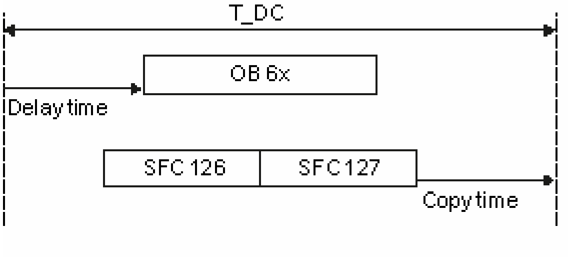
4.8.5.1 Basics of Programming

Programming in the Synchronous Cycle Interrupt OBs

The programming of the isochronous mode program section is solely in the synchronous cycle interrupt OBs OB61 to OB64. Since the synchronous cycle interrupts are executed with a high priority, only the time-critical program sections should be executed in OB6x. The synchronous cycle interrupt is called with a delay time configured by the user.

Access to the isochronous mode I/O by calling system functions

The updating of the isochronous mode I/O is achieved solely by calling the system functions SFC126 "SYNC_PI" and SFC127 "SYNC_PO" and therefore in the process image. Direct access to the I/O areas returns current values of the process that do not, however, necessarily relate to other values.



The system functions SFC126 "SYNC_PI" and SFC127 "SYNC_PO" can only update the process image partition in the permitted execution window. The execution window for the possible SFC 126/127 calls is from the end of the cyclic data exchange in PROFINET IO until shortly before the end of T_DC. The data exchange must be started within this time window. If the execution window is violated by execution of the system function SFC126 or SFC127, the system functions indicate this with an error message to this effect.

Note

To prevent inconsistent data being returned for OB6x, you should avoid using SFC14/15 (direct data access) in the isochronous OB.

Models of program execution

Depending on the order in which the system functions SFC126 "SYNC_PI" and SFC 127 "SYNC_PO are called in OB6x, program execution can be divided into two basic models:

- **IPO** model (read **I**nputs - **P**rocessing - write **O**utputs), CACF = 1
- **IPO** model (read **I**nputs - **P**rocessing - write **O**utputs), CACF > 1

CACF is the IO controller application clock cycle.

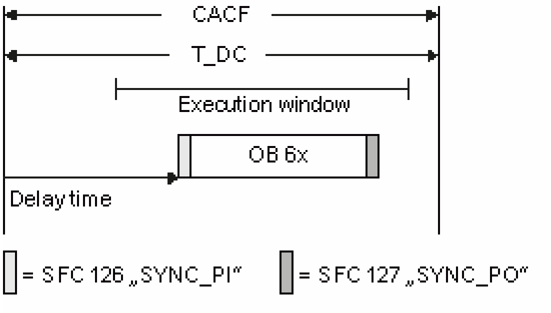
4.8.5.2 Program Execution According to the IPO Model with Short Time

Characteristics of the IPO model CACF = 1

The main feature of the IPO model is that the processing of the I/O data is completed within one system clock cycle T_DC. The shortest reaction times are achieved with this model.

IPO model data cycle = 1

With the IPO model, there is a constant processing time from the "input terminal" to the "output terminal" of Ti + T_DC + To. A process reaction time Ti + 2xT_DC + To can be guaranteed.



The figure shows the IPO model from acquisition through processing on the CPU to output of the process values. STEP 7 calculates the delay time for your or you can enter it yourself (see Configuration). In this time, the read input data is its way on PROFINET IO.

The start is the transfer with SFC126 "SYNC_PI" and the end is SFC127 "SYNC_PO". The delay time is set as default to the start of the execution window but can be moved.

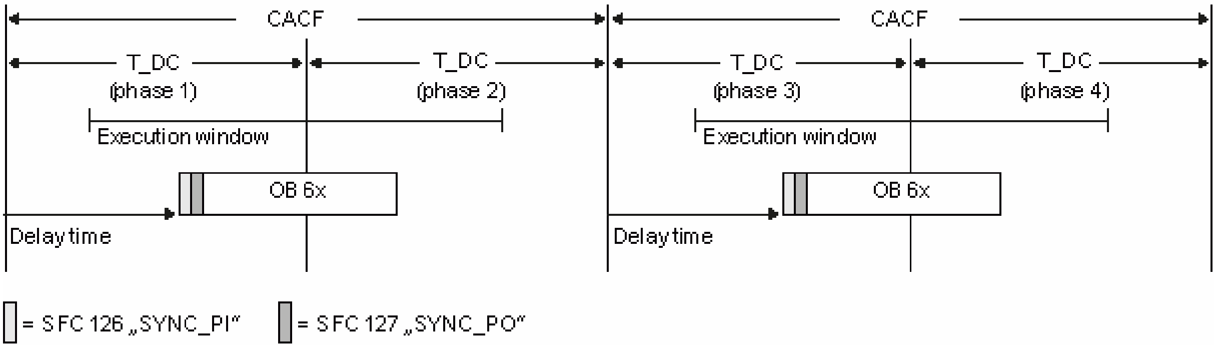
SFC126 "SYNC_PI" and SFC172 "SYNC_PO" may only be executed in the execution window. Execution must be completed within the data cycle T_DC. If CACF is set to a value >1, this time is available for other processing in the process.

4.8.5.3 Program Execution According to the IPO Model with Long Time

Characteristics of the IPO model CACF > 1

The IPO model with CACF > 1 is an IPO model extending over several IO controller application clock cycles.

The SFCs can be called in the first data cycle and execution of OB6x is spread over several data cycles.



The figure shows the signals of the IPO model over time from the acquisition through the processing on the IO controller to the output of the process values. STEP 7 calculates TM for you. In this time, the read input data is on its way on PROFINET IO.

The IPO model > 1 is particularly suitable for large I/O structures because this model allows longer calculation times when processing the input data and obtaining the corresponding output data.

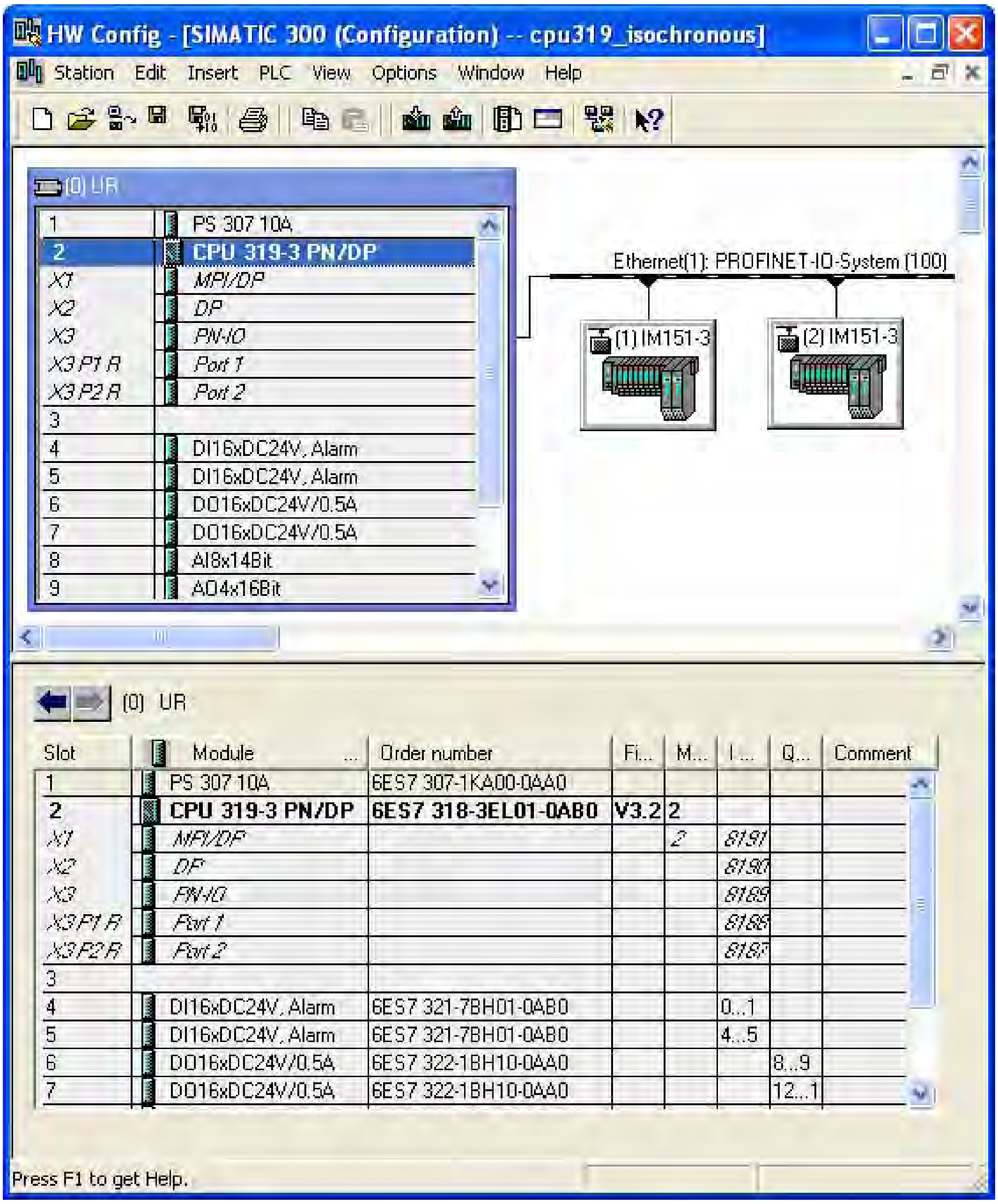
With the IPO model, there is a constant processing time from the "input terminal" to the "output terminal" of Ti + 2xT_DC + To. A process reaction time Ti + (2xCACF+1) + T_DC+ To can be guaranteed.

4.8.5.4 Configuration

Requirements for creating a configuration

First perform the steps below:

1. Create a project with the name "cpu319_isochronous" in the SIMATIC Manager.
2. Insert a new "SIMATIC 300 station".
3. Open the station in HW Config and configure a CPU319-3 PN/DP with a PROFINET IO system.
4. Add two IM151-3 PN HS to the PROFINET IO system.


HW Config - [SIMATIC 300 (Configuration) -- cpu319_isochronous]
Station Edit Insert PLC View Options Window Help
Ethernet(1): PROFINET-IO-System (100)
(1) IM151-3
(2) IM151-3
Press F1 to get Help.

(0) UR

1	PS 307 10A
2	CPU 319-3 PN/DP
X1	MPI/DP
X2	DP
X3	PN-IO
X3 P1 R	Port 1
X3 P2 R	Port 2
3	
4	DI16xDC24V, Alarm
5	DI16xDC24V, Alarm
6	DO16xDC24V/0.5A
7	DO16xDC24V/0.5A
8	AI8x14Bit
9	AO4x16Bit

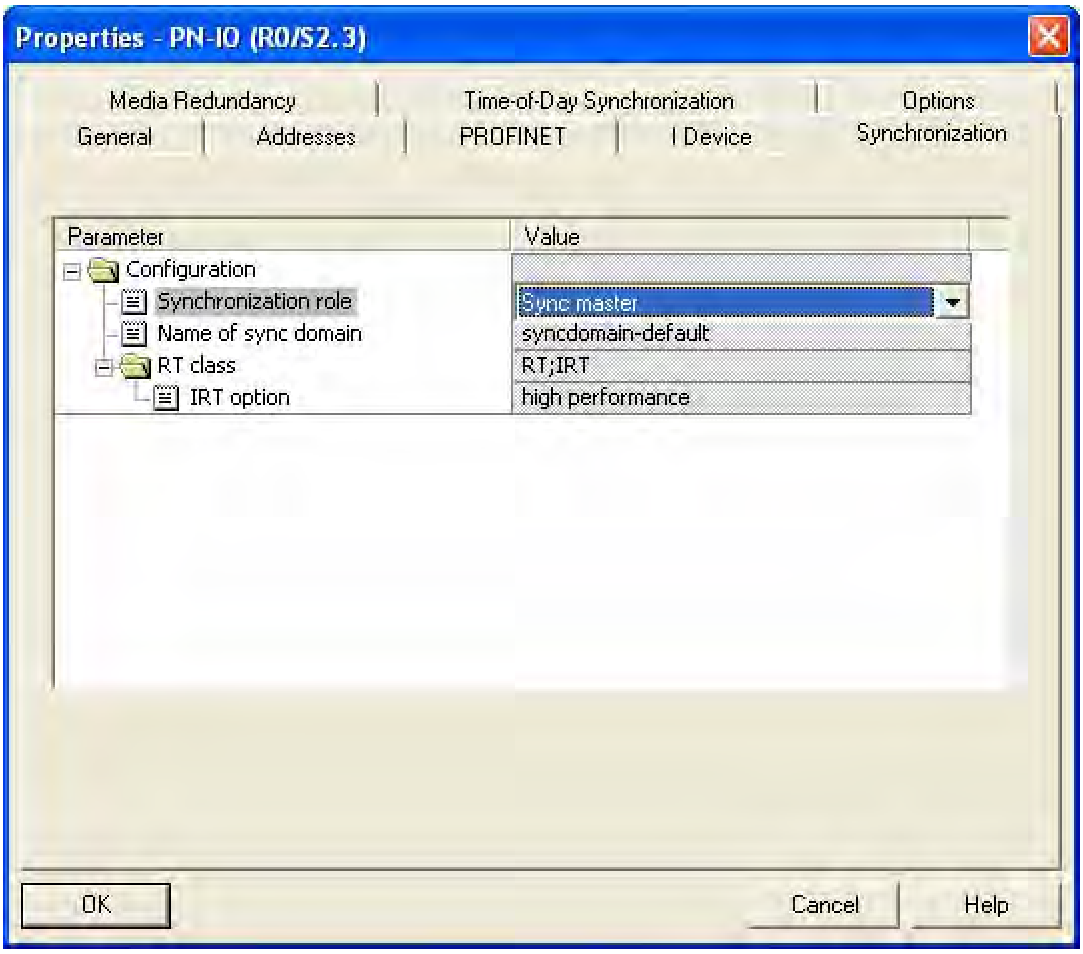
(0) UR

Slot	Module ...	Order number	Fi...	M...	I...	Q...	Comment
1	PS 307 10A	6ES7 307-1KA00-0AA0					
2	CPU 319-3 PN/DP	6ES7 318-3EL01-0AB0	V3.2	2			
X1	MPI/DP			2	8191		
X2	DP				8190		
X3	PN-IO				8189		
X3 P1 R	Port 1				8188		
X3 P2 R	Port 2				8187		
3							
4	DI16xDC24V, Alarm	6ES7 321-7BH01-0AB0			0...1		
5	DI16xDC24V, Alarm	6ES7 321-7BH01-0AB0			4...5		
6	DO16xDC24V/0.5A	6ES7 322-1BH10-0AA0				8...9	
7	DO16xDC24V/0.5A	6ES7 322-1BH10-0AA0				12...1	

Synchronization of the CPU

The CPUs must be configured as sync masters, as shown in the following screenshot.

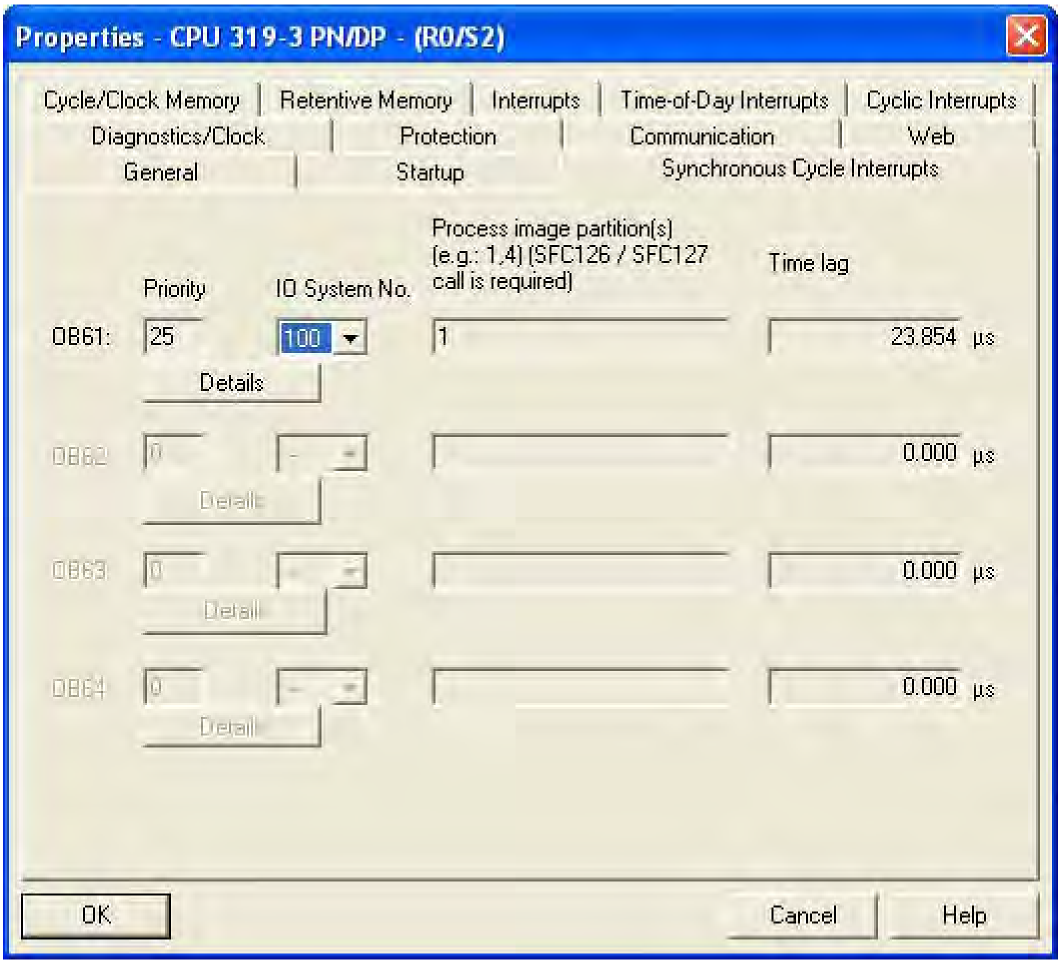
Configure the CPU with the IRT option "high performance".



Parameter	Value
Configuration	
Synchronization role	Sync master
Name of sync domain	syncdomain-default
RT class	RT;IRT
IRT option	high performance

CPU properties

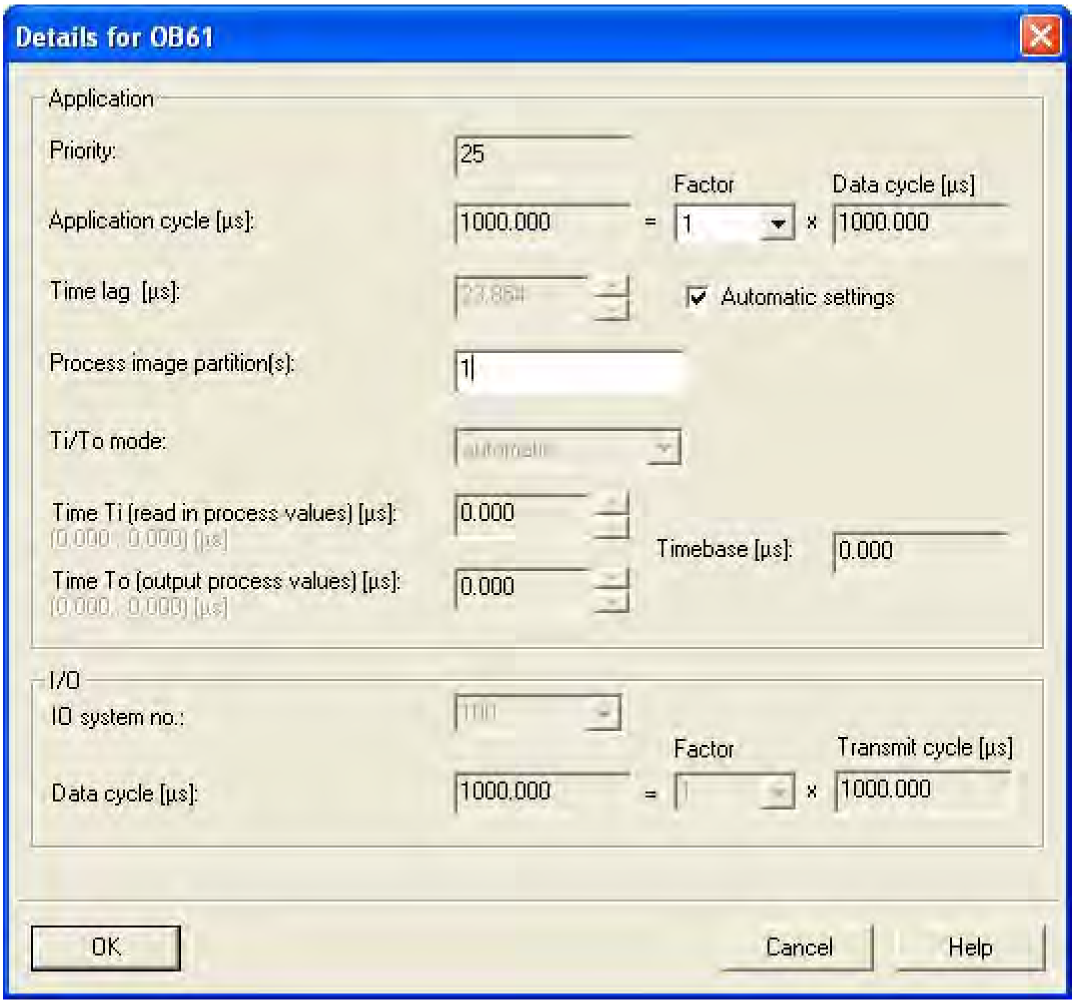
1. Open the "Synchronous Cycle Interrupts" tab in the CPU properties and assign the previously created PROFINET IO system (100) to OB61.



Properties - CPU 319-3 PN/DP - (R0/S2)

	Priority	IO System No.	Process image partition(s) (e.g.: 1,4) (SFC126 / SFC127 call is required)	Time lag
OB61:	25	100	1	23.854 µs
OB62	0	-		0.000 µs
OB63	0	-		0.000 µs
OB64	0	-		0.000 µs

2. Open the "Details for OB61" dialog and select the relevant process images there.



Details for OB61

Application

Priority: 25

Application cycle [µs]: 1000.000 = Factor 1 x Data cycle [µs] 1000.000

Time lag [µs]: 23.864

☑ Automatic settings

Process image partition(s): 1

Ti/To mode: automatic

Time Ti (read in process values) [µs]: 0.000
(0.000 .. 0.000) [µs]

Time To (output process values) [µs]: 0.000
(0.000 .. 0.000) [µs]

Timebase [µs]: 0.000

I/O

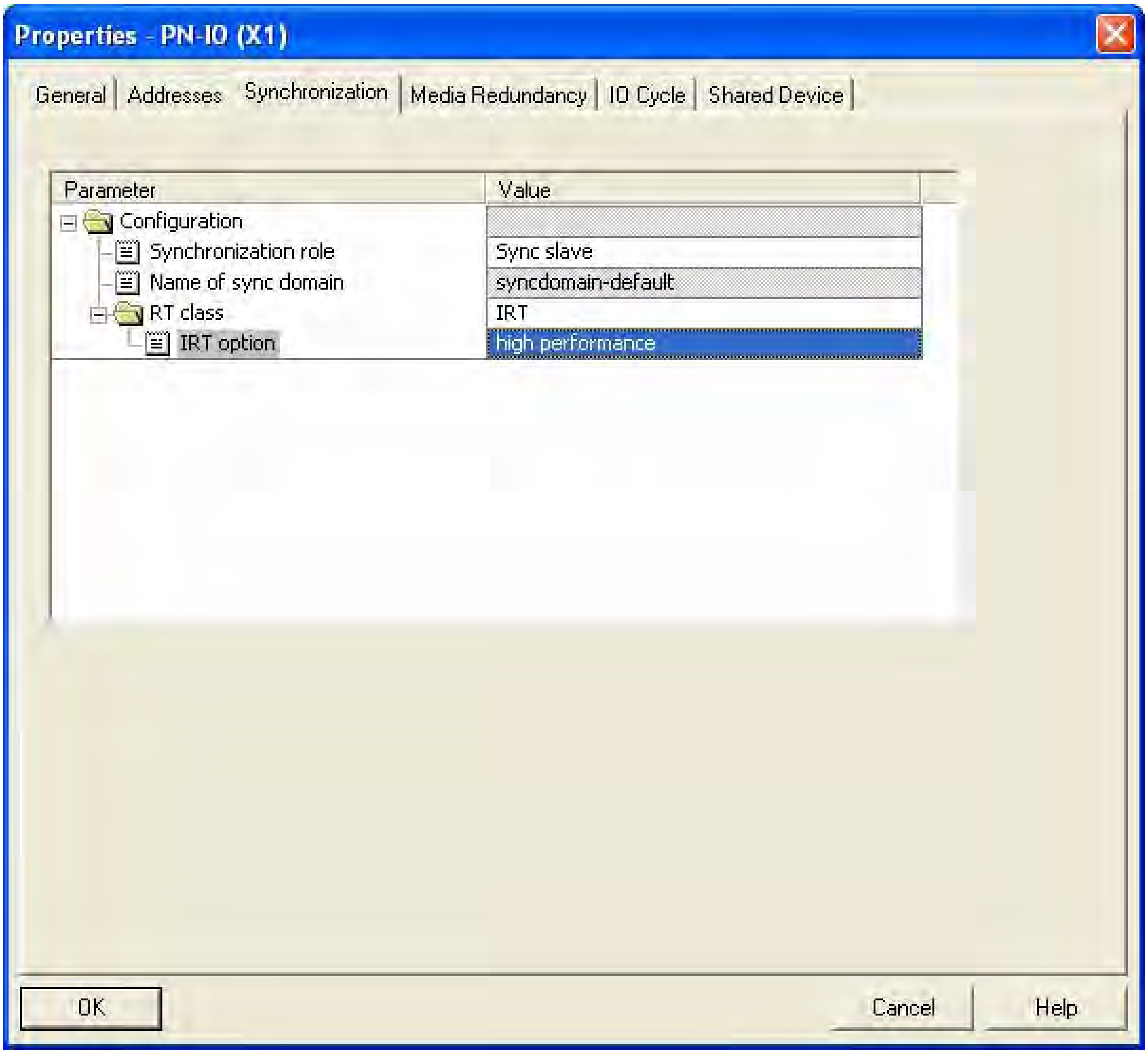
IO system no.: 100

Data cycle [µs]: 1000.000 = Factor 1 x Transmit cycle [µs] 1000.000

Synchronization of the IO devices

Set the synchronization individually for the IO devices to be operated in isochronous mode. To do this, double-click on the PNIO interface and make the following settings there:

1. Open the "Synchronization" tab.
2. Change the synchronization role of the IO device to "Sync slave".
3. Set the RT class "IRT".
4. Select the IRT option "high performance".



Parameter	Value
Configuration	
Synchronization role	Sync slave
Name of sync domain	syncdomain-default
RT class	IRT
IRT option	high performance

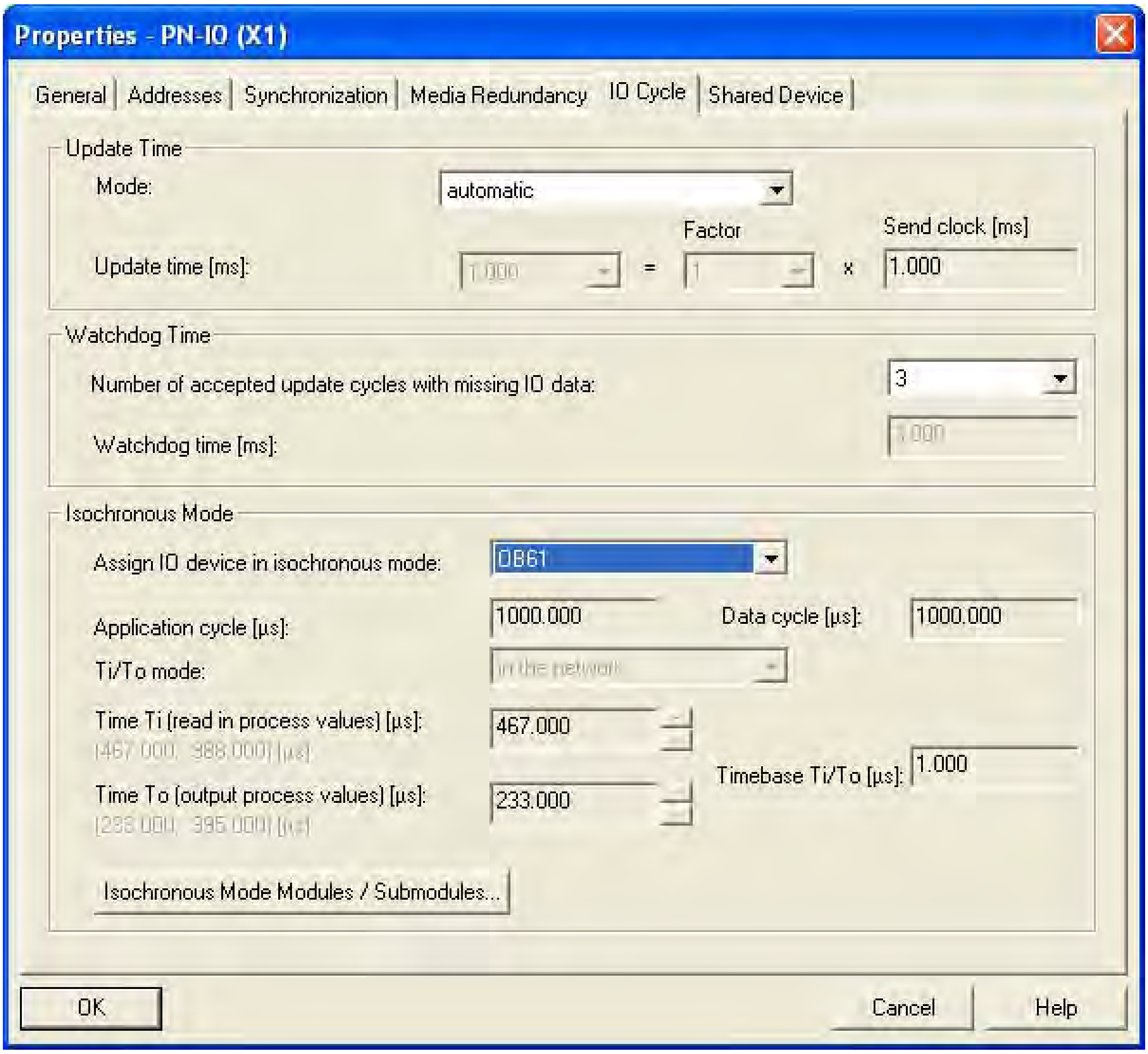
Note

Remember that the IRT option "high performance" is possible only with topology configuration.

IO device properties

Open the "IO Cycle" tab of the PNIO properties dialog and set the following:

1. "Update time" section: mode "Automatic"
2. "Isochronous mode" section: assign isochronous mode to an IO device "OB61"



Update Time

Mode: automatic

Update time [ms]: 1.000 = Factor 1 x Send clock [ms] 1.000

Watchdog Time

Number of accepted update cycles with missing IO data: 3

Watchdog time [ms]: 3.000

Isochronous Mode

Assign IO device in isochronous mode: OB61

Application cycle [µs]: 1000.000

Data cycle [µs]: 1000.000

Ti/To mode: in the network

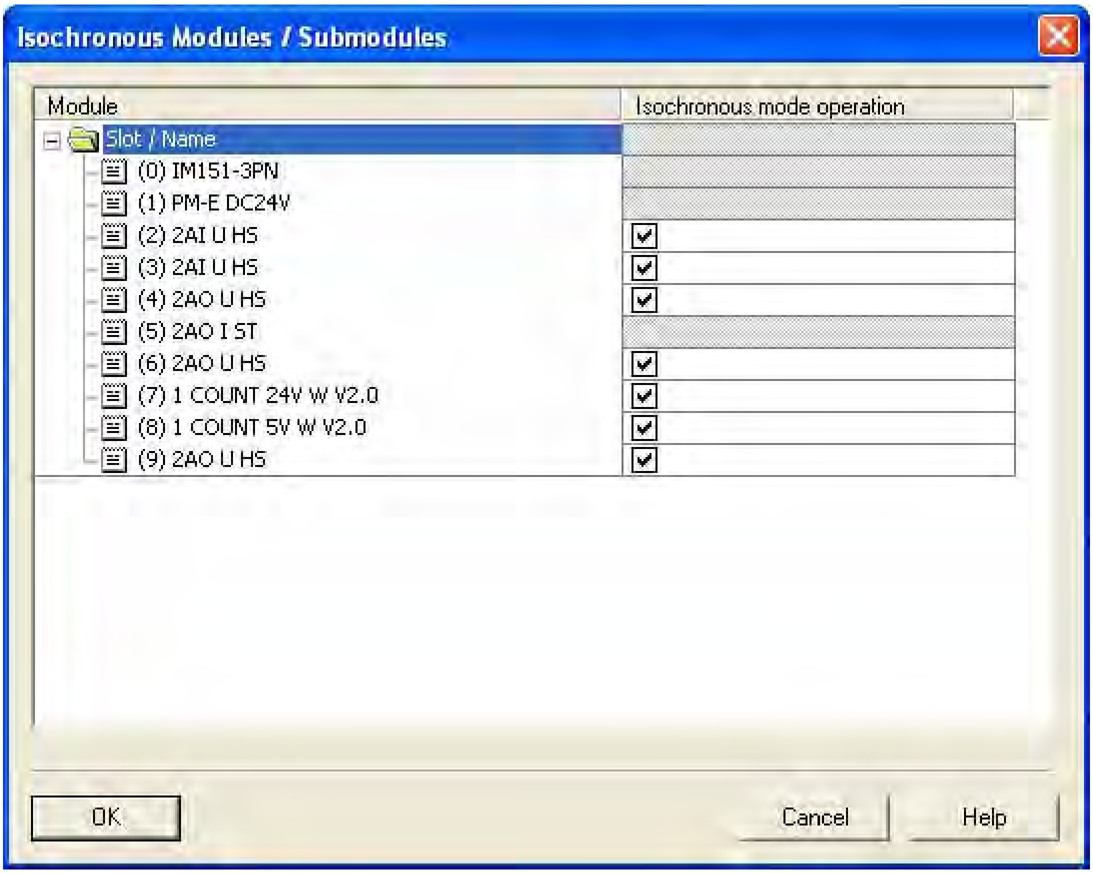
Time Ti (read in process values) [µs]: 467.000
[467.000...988.000] [µs]

Time To (output process values) [µs]: 233.000
[233.000...395.000] [µs]

Timebase Ti/To [µs]: 1.000

Isochronous Mode Modules / Submodules...

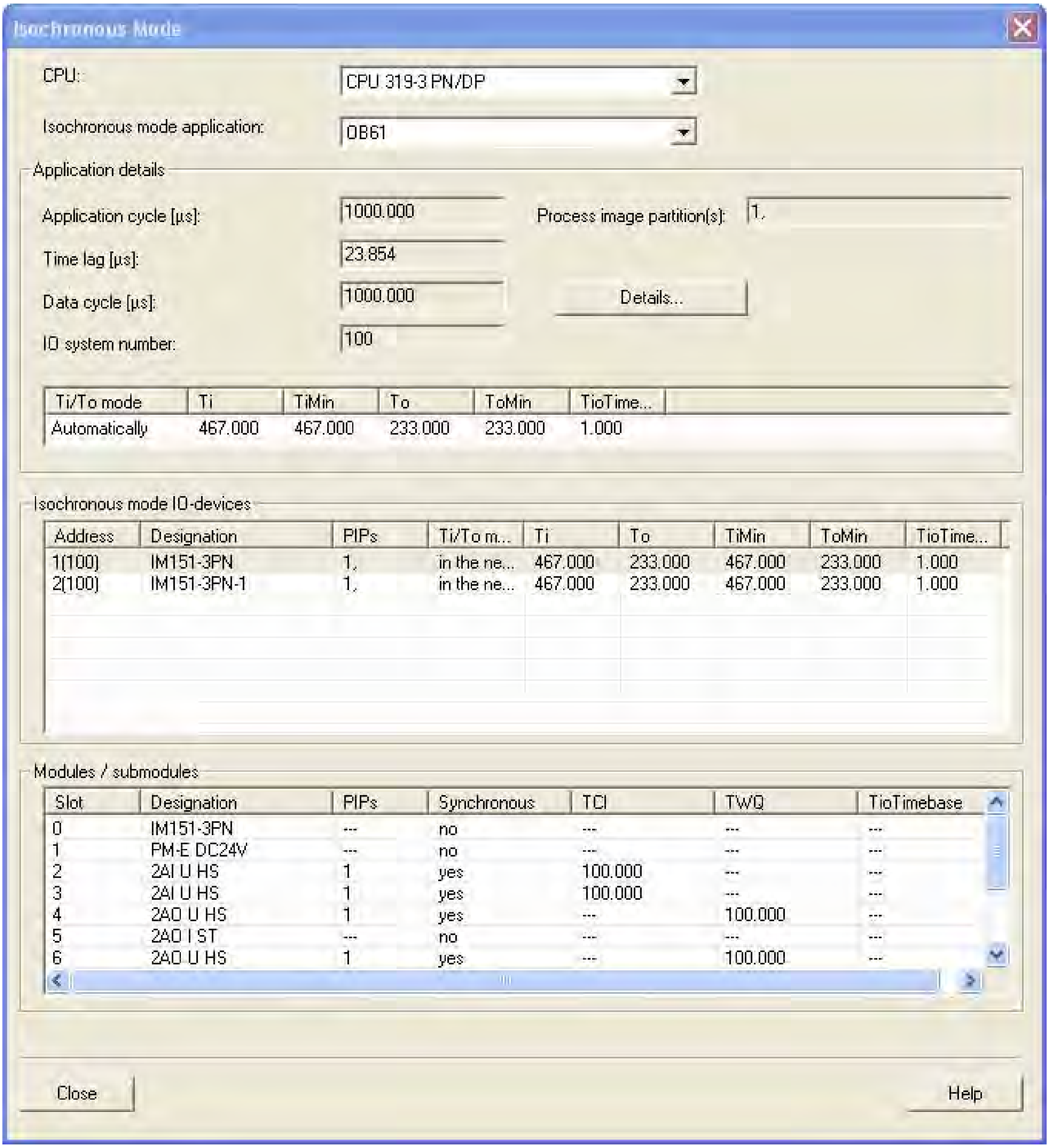
The "Isochronous Mode Modules / Submodules..." button provides you with an overview of the modules operating in isochronous mode. The dialog also allows you to enable / disable operation in isochronous mode for individual modules.



Module	Isochronous mode operation
Slot / Name	
(0) IM151-3PN	
(1) PM-E DC24V	
(2) 2AI U HS	☑
(3) 2AI U HS	☑
(4) 2AO U HS	☑
(5) 2AO I ST	
(6) 2AO U HS	☑
(7) 1 COUNT 24V W V2.0	☑
(8) 1 COUNT 5V W V2.0	☑
(9) 2AO U HS	☑

Overview

If you open **Edit > PROFINET IO > Isochronous Mode** in HW Config, you obtain an overview of the entire in isochronous mode project.



CPU: CPU 319-3 PN/DP

Isochronous mode application: OB61

Application details

Application cycle [µs]: 1000.000

Process image partition(s): 1,

Time lag [µs]: 23.854

Data cycle [µs]: 1000.000

IO system number: 100

Ti/To mode	Ti	TiMin	To	ToMin	TioTime...
Automatically	467.000	467.000	233.000	233.000	1.000

Isochronous mode IO-devices

Address	Designation	PIPs	Ti/To m...	Ti	To	TiMin	ToMin	TioTime...
1(100)	IM151-3PN	1,	in the ne...	467.000	233.000	467.000	233.000	1.000
2(100)	IM151-3PN-1	1,	in the ne...	467.000	233.000	467.000	233.000	1.000

Modules / submodules

Slot	Designation	PIPs	Synchronous	TCI	TWQ	TioTimebase
0	IM151-3PN	---	no	---	---	---
1	PM-E DC24V	---	no	---	---	---
2	2AI U HS	1	yes	100.000	---	---
3	2AI U HS	1	yes	100.000	---	---
4	2AO U HS	1	yes	---	100.000	---
5	2AO I ST	---	no	---	---	---
6	2AO U HS	1	yes	---	100.000	---

Automatic times

In this dialog, you can see the times set by STEP 7. You have the option of setting the times manually. If you want to do this, follow the steps described in the next section.

How do I stop times being assigned automatically

If the automatic calculation results in a Ti or To that does not suit your purposes, you will need to take this module out of the automatic calculation and have Ti or To calculated individually.

4.8.6 Diagnostics and Interrupt Characteristics

Diagnostics and interrupt characteristics

To synchronize clock cycles, you can use the diagnostics and interrupt functions of STEP 7. These reduce down times and simplify localization and elimination of problems.

Below, you will find the events for diagnostics and interrupt functions and remedies for the problem.

Diagnostics in detail

Event	Cause of the problem	Remedy
OB80 comes and OB6x reports how many calls it has missed.	OB6x takes too long	• Shorten OB6x • Increase the data cycle • Set a shorter delay time •
OB6x reports "Overrun", it came too late.	• OB6x was executed too quickly. • SFC 126/127 report "too early". • SFC 126/127 report "too late".	• Extend the delay time
Diagnostic interrupt	Module reports a module fault	The module changes to STOP or OB82 (if it exists) is called.
I/O access error	Input or output module does not respond.	The module changes to STOP or OB122 (if it exists) is called.
Rack failure	Station is not accessible.	The module changes to STOP or OB86 (if it exists) is called.
The IO controller is sync slave and not synchronized.	• If the GC_Viol bit is set to "1" in OB6x, ...	... you will need to turn on the SYNC master.
IO device is not synchronized with OB6x.	• For the IO device, "station failure" is displayed. • The channel error "Sync discrepancy" is mapped to Sync Violation on IO controller, there is a power failure.	The module changes to STOP or OB86 (if it exists) is called

5 Configuring a SIMATIC PC Station (SIMATIC PC based)

5.1 Creating and Assigning Parameters to SIMATIC PC Stations

Introduction

The SIMATIC PC station (referred to here simply as "PC station") represents a PC or an operator station that includes the SIMATIC components as applications (such as WinCC), slot PLCs, or software PLCs for automation tasks. These components are configured within the PC station, or they can be the end point of a connection.

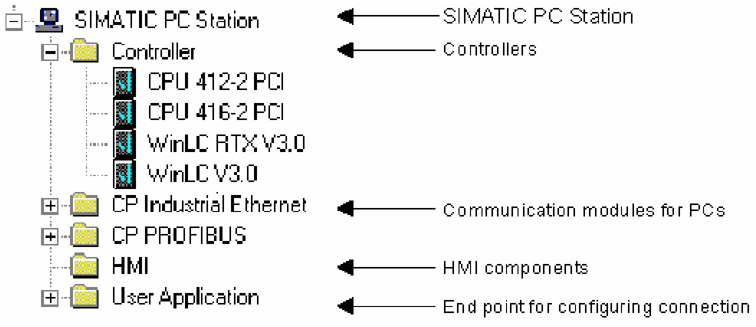
Overview: Configuring a SIMATIC PC Station and the Connections for It

The complete configuration of a PC station is handled with the "SIMATIC PC Station" station type. The components that are configured there are then available for configuring connections. You can configure the following components in a SIMATIC PC station:

- Software PLC WinLC (Version V3.0 and higher)
- Slot PLC CPU 41x-2 PCI (WinAC Slot 412 and WinAC Slot 416)
- SIMATIC communication processor for the PC
- SIMATIC HMI: WinCC and ProTool Pro
- Applications (such as user applications based on SAPI)
- OPC-Server
- S7 modular embedded controller (S7-mEC) (available as HSP Update 135)

For PC-based components such as WinLC (≤ V2.x) or CPU 416-2 DP ISA, you must continue to use the SIMATIC 300 or SIMATIC 400 station type. You can find the components in the "Hardware Catalog" window under SIMATIC PC Based Control 300/400.

The following figure shows the structure of the Hardware Catalog for SIMATIC PC stations:



SIMATIC PC Station

Controllers

Communication modules for PCs

HMI components

End point for configuring connection

Procedure

1. In the SIMATIC Manager, insert a SIMATIC PC station in your project) menu command **Insert > SIMATIC PC Station**).
2. Change the name of the SIMATIC PC station as you see fit.
   If the computer that you are now using for configuration and for loading stations is identical with the SIMATIC PC station inserted in the SIMATIC Manager, then the name of this station must be identical with the name specified in the Component Configuration Editor. Only then is the SIMATIC PC station properly "assigned" in the project; in other words, it is then recognized in the SIMATIC Manager and in the network view as an assigned SIMATIC PC station.
3. Double-click the SIMATIC PC station object and then the Configuration object.
   HW Config opens for editing the station configuration. Row 125 of the configuration table is occupied by the station manager (cannot be deleted).
4. Use a drag-and-drop operation to insert those components into the configuration table of the SIMATIC PC station that represent the configuration of the real PC. The components are located in the "Hardware Catalog" window under SIMATIC PC Station:
   - Software PLC WinLC in slot 2 (for information on expanded configuration options, see Slot Rules for PC-based SIMATIC Controllers)
   - Slot PLC CPU 41x-2 PCI (WinAC Slot 412 and WinAC Slot 416) in slot 3 (for information on expanded configuration options, see Slot Rules for PC-based SIMATIC Controllers)
   - CPs in one of the slots from 1 to 32 (if necessary, observe the restrictions in the Product Information for the current slot-CP!)
   - SW placeholder for configuring connections such as "User Application" or HMI components (if installed) also in slots 1 to 32
   - OPC Server for OPC Clients accessing the variables of remote automation systems via configured connections also in slots 1 to 32.
5. Select the rack and open the Properties dialog of the PC station (menu command **Edit > Object Properties**), select the Configuration tab, and put the path for storing the configuration file (*.XDB file) there.
   The connection data and addresses for CPs and applications are stored in this file (see step 6).

6. Select the menu command **PLC > Save and Compile**.
   When the configuration of a SIMATIC PC station is saved and compiled, system data and a configuration file (*.XDB file) are created. This system data and the configuration file must then be downloaded to the programmable logic controller or installed (see Downloading a PC Station).
   You set the location of the configuration file on the PC station by means of the "Set PG/PC Interface" program ("Configuring STEP 7" tab).
   You can find more information on installing on a programming device/PC in the description "SIMATIC NET, S7 Programming Interface" and in the online Help on the "Configuring STEP 7" tab.
7. If you want to configure connections for the SIMATIC PC station:

   Select the menu command **Options > Configure Network**. In NetPro, you can configure S7 connections for the control components and for applications. You can also configure redundant S7 connections for the OPC server. If you want to edit the SIMATIC PC configuration from NetPro, double-click the SIMATIC PC station object (as you would for SIMATIC 300/400 stations).
8. If you want to download the configuration:
   Select the menu command **PLC > Download**. Only PC-based controls will be offered for the download (see Downloading a PC Station).

Special Features of Configuring WinLC V3.0 and Applications

The PROFIBUS-DP interface for WinLC is already included in WinLC as an integrated interface of WinLC (DP-Master in slot 2.1). Accordingly, the card required for this function (communications module such as CP 5613) is not placed and configured explicitly in HW Config. The configuration is handled by means of the DP Master entry in slot 2.1.

Configuration with Applications and PROFIBUS CPs

Applications can access distributed I/O by means of PROFIBUS-DP configuration. To create the configuration, proceed as follows:

1. Place an application and a PROFIBUS-DP CP in the configuration table for the SIMATIC PC station.
2. Double-click the CP to call up the Properties dialog.
3. In the Properties dialog, assign an application to the CP.

5.2 Slot Rules for a SIMATIC PC Station

Components such as CPs, PC-based user programs for configuring connections and OPC servers can be inserted in slots (index) 1 to 32.

SoftPLC

Up to STEP 7 V 5.2 Service Pack 1, the software PLC WinLC can only be inserted in Index 2. As of STEP 7 V 5.3, the WinLC variant V 4.1 is supported. It can be inserted in all slots and can be operated with up to four SlotPLCs (CPU 41x-2 PCI as of V 3.4).

SlotPLC

Up to STEP 7 V 5.2 Service Pack 1, the SlotPLC CPU 41x-2 PCI (WinAC Slot 412 and WinAC Slot 416) can only be inserted in Index 3.

As of STEP 7 V 5.3, the CPU 41x-2 PCI V 3.4 is supported. It can be inserted in all slots. Up to 4 of these CPUs can be inserted, even together with the new SoftPLC WinLC V 4.1.

Address Areas with Multi-PLC Operation

If there are several WinAC controllers inserted in a PC station, then each of them has its own I/O address areas. The controllers (e.g. SlotPLCs) can communicate with each other by means of S7 communications mechanisms (BSEND/BRCV); otherwise, however, they function independently of each other, comparable to the operation of S7-400 CPUs in a segmented rack, such CR1. This property allows the compactness and integration density of a PLC to be increased.

Further Information

For further information, please refer to the manuals for the SoftPLCs and SlotPLCs as well as the manual "Commissioning PC Stations - Manual and Quick Start".

5.3 Comparison: S7 Station - PC Station

The configuration of S7 stations is represented in the SIMATIC Manager by the Hardware object; if you double-click Hardware, you start the application for configuring an S7 station. End points of a connection are modules that are plugged in to the real station. Configurable CPUs, CPs, or interface modules form the interfaces to subnets.

The configuration of PC stations is represented in the SIMATIC Manager by the Configuration object; if you double-click Configuration, you start the application for configuring a PC station (as of STEP 7 V5.0 Service Pack 3).

5.4 Appending SIMATIC PC Configurations of Previous Versions

PC Configurations of STEP 7 V5.1 Projects (up to SP 1)

As of STEP 7 V5.1, Service Pack 2 you can download communications to the PC station in the same way as to an S7-300 or S7-400 station (without having to take the roundabout via configuration file). Nevertheless, a configuration file is always generated during a storing or compiling operation in order to enable the transmission of the configuration to the target PC station using this method.

This bears the consequence that "older" PC stations cannot interpret some of the information included in the newly generated configuration files. STEP 7 automatically adapts itself to this circumstance:

- If you create a **new** SIMATIC PC station configuration with STEP 7 as of V5.1, Service Pack 2, STEP 7 assumes that the target PC station was configured with the help of SIMATIC NET CD as of 7/2001, that is, under the presumption that S7RTM (Runtime Manager) is installed. The configuration files are generated in such a way that they can be interpreted by a "new" PC station.
- If you append a SIMATIC PC station configuration of a previous version (for example, the PC station was configured with STEP 7 V5.1, Service Pack 1),. STEP 7 does **not** presume that the target PC station was configured with the help of SIMATIC NET CD as of 7/2001. Those configuration files are then generated in such a way that they can be interpreted by an "old" PC station.

If this default behavior does not match your requirements, you can modify it as described below:

Setting in the Context Menu "Configuring Hardware ":

1. Open the PC station hardware configuration
2. Right-click on the station window (white area)
3. Select the context-sensitive menu "Station Properties"
4. Check or clear the "Compatibility" checkbox.

Setting in the Context Menu "Configuring Networks"

1. Open the network configuration
2. Highlight the PC station
3. Select the menu command **Edit > Object properties**
4. In the dialog, select the "Configuration" tab
5. Check or clear the "Compatibility" checkbox.

PC Configurations of STEP 7 V5.0 Projects

You must convert the station if you want to edit a SIMATIC PC station configuration with STEP 7 as of V5.0, Service Pack 3 to configure new components that are only supported by Service Pack 3 or higher:

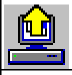
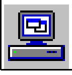
1. In the SIMATIC Manager, highlight the SIMATIC PC station and select the menu command **Edit > Object properties**.
2. In the "Functions" tab of the properties dialog, click on the "Expand" button.
   The SIMATIC PC station is then converted. Now, it can only be edited with STEP 7 V5.0, Service Pack 3 or later versions.

5.5 Highlighting the SIMATIC PC Station to be Configured in the Network View

If you want the SIMATIC PC station that you are using to create the STEP 7 project to have a highlighted display in the network view and in the SIMATIC manager for the project, the following prerequisites must be met:

- No programming device (PG)/PC can be assigned
- The PG/PC interface must be set to "PC internal (local)"
- The PC station must be configured with the Station Configuration Editor, which is available on the SIMATIC NET CD as of 7/2001.
- The name of the PC station entered must be **identical** in the Station Configuration Editor and in STEP 7 (station properties of the SIMATIC PC station).

Since this name is then identical everywhere, STEP 7 can then make the correct association and highlight the assigned PC station. Should any one of the above-mentioned prerequisites not be met, then the "standard" icon for the PC station is displayed.

Meaning	Icon
SIMATIC PC station highlighted	
SIMATIC PC station not highlighted	

6 Saving, Importing and Exporting Configurations

6.1 Saving a Configuration and Checking the Consistency

To save a configuration with all set parameters and addresses, select the menu command **Station > Save** or **Station > Save and Compile**.

If you use the menu command **Station > Save and Compile**, the configuration is saved in the active project as a "Station" object and if valid system data blocks (SDB) could be created, they are stored in the (offline) "Blocks" folder for the associated modules ("SDB carrier," for example, CPU). The system data blocks are represented by the "System Data" folder/symbol.

💼	**Symbol for system data**

To be able to save incomplete configurations, select the menu command **Station > Save**. Using this command no system data blocks are created when saving. The save procedure takes less time than if you save and compile, but you should be aware that there may be inconsistencies between the configuration saved in the "Station" object and the configuration saved in the system data.

Before downloading, you should check your station configuration for errors using the menu command **Station > Consistency Check**.

6.2 Importing and Exporting a Configuration

Introduction

From STEP 7 V5 you can handle station configurations not only together with the whole project (for example, save or open them), but also export and import them independently of the project in text form (ASCII file).

Applications

- You can distribute files by electronic means (for example, by e-mail)
- You can save files for use with future STEP 7 versions
- You can print export files using word processing systems or process them for documentation purposes.

What is exported/imported?

During hardware configuration the data that can be exported or imported are those data required for the configuration and parameter assignment of modules. As of STEP 7 V5.1, Service Pack 1, it is also possible to export and import subnet data.

The following data are **not** acquired:

- Data which are managed via other applications (for example, programs, connections, shared data)
- The CPU password which was set (for this reason, with F-CPUs the setting "CPU contains safety program" is reset since it cannot be made without a password).
- Data not specific to one station (for example, links to intelligent DP slaves or configuration for direct data exchange)

Note

If your configuration contains modules from earlier optional packages, it is possible that not all module data will be acquired when you use the "Export Station" function. In this case, you should check that the module data are complete after importing.

Export File

You can set what is stored in the exported text file and in what form it is stored when you export (**Station > Export** menu command):

- Legible or Compact Format

  Important: If you export the station configuration in order to read it in with other STEP 7 versions, select the "**Compact**" option.
- The name of the file can be (*.cfg) freely selected
- With or without symbols
- With or without subnets
- Default values for module parameters can be left out (STEP 7 "knows" the default values and adds them from the internal module data when you import)

Caution

If you export a station configuration with symbols, you can no longer import this file with STEP 7 V5, SP 1 or an older STEP 7 version.

Procedure (Exporting)

1. Open a station configuration or save the station configuration you are currently editing (**Station > Save** menu command).
2. With a station configuration open, select the menu command **Station > Export.**
3. In the dialog box that appears, enter the path and name of the export file, the format, and other options.
4. Confirm your settings with "OK."

Procedure (Importing)

Recommendation: Do not import an exported station configuration from the same project. In this case, STEP 7 cannot resolve the network assignment. For the import, select a different or new project.

1. Make sure that you have the same Windows language setting for the import as for the preceding export. Different Windows language settings for export and import may lead to errors during imports.
2. With an empty station configuration open, select the menu command **Station >Import**
   If no station configuration is open, a dialog box appears for the selection of a project. In this case, navigate to the project into which you want to import the station configuration.
3. In the dialog box that appears, navigate to the text file you want to import.
4. Confirm your settings with "OK."
   On importing, STEP 7 checks the imported file for errors and consistency and outputs messages.

Importing into an Existing Station

You can also import a station into an open station configuration. On importing, STEP 7 queries whether already configured modules/interface submodules should be overwritten. For every component you can decide whether you want to keep it or overwrite it.

When a component is overwritten, all settings (parameters) in the import file become valid. Any settings that are not included in the import file remain in the station configuration.

6.3 Exporting and Importing CAx Data

As of STEP 7 V5.4, you can export stations in a format that is "understood" by CAx systems, where they can be processed further.

Similarly, you can also export station data from a compatible CAx system and import them into a station in an STEP 7 project.

In this sense, "CAx" stands for CAD, CAE, E-CAD and E-CAE and refers to the data used in these special programs.

This feature saves you from having to enter the same data twice.

What is exported?

- SIMATIC stations (S7-300, S7-400, S7-400H, PC stations)
- GSD-based modules (PROFIBUS DP, PROFINET IO)
- Subnets

What is not exported?

The CAx interface does not supply parameters of objects. For this reason, the export is not complete, which means that you will have to expect data loss after a station has been exported and then imported. The export is intended to make data from Hardware Configuration in STEP 7 usable in CAx systems.

In particular, the following items are **not** exported:

- Blocks, parameters and connections
- CiR submodules (for configuration changes in RUN mode)
- Intelligent DP slaves (I slaves) that are configured in their own station
- HART submodules
- IO Link hardware configured with S7 PCT (Port Configuration Tool)

Export Files

For each station, an export file is created in XML format. This XML file follows the SimaticML-CAx-v1.0-xsd scheme.

If there are slaves or IO devices in the station that are integrated by means for GSD files, these files are also copied into the target folder.

Procedure: Exporting CAx Data

1. Open the project containing the stations to be exported.
2. If you want to export an individual station, select it in the SIMATIC Manager. If you want to export several stations, then select the project (no station selected).
3. In the SIMATIC Manager, select the **Options > CAx Data > Export** menu command.
4. In the "Export CAx Data" dialog box, select the destination folder, and, if applicable, the stations to be exported.
5. Click the "Export" button.
   If an export file from a station is already present in the destination folder, select the appropriate option in the dialog that is displayed (Replace, Don't Replace...).
   The status (success or failure) of the export process is displayed in the "Status" column.

Import

An import can be made into an "empty" station or into an already configured station.

If the station already contains objects (racks, modules, slaves, ...) then a comparison of the incoming and existing objects is carried out during the import. The results of this comparison are shown in the import dialog box. The following criteria apply during the comparison:

- Modules/submodules are compared in terms of their slots.
- Subnets are compared in terms of their names and types.
- Racks and slaves are compared in terms of their numbers.

Procedure: Importing CAx Data

1. Open the station to receive data imported from a CAx system.
2. In the SIMATIC Manager, select the **Options > CAx Data > Import** menu command.
3. In the dialog that is then displayed, go to the XML file that was generated by the CAx system.
4. Confirm your selection with "OK".
5. In the "Import CAx Data" dialog box, select the objects to be imported ("Selection" column).
   If the station to receive data imported from a CAx system already contains objects, then these objects may have other properties than the objects in the XML file. In this case, you will have to decide which properties are to apply (see the Help on the dialog box).
6. Click the "Import" button.
7. After the import is completed, evaluate the status of the objects. If necessary, take a look at the log of the import process. To do so, click the "Show Log" button. The log file contains status information on the complete import process. This log will be overwritten during the next import.
8. If components with GSD files were used in the CAx system, you will have to install these components for the corresponding stations.

Error Analysis after Import

Errors during imports are shown in red.

If a module or an object cannot be created, check whether the values are set for the following attributes:

- Order number
- Version
- Submodule
- Slot
- Subslot (for a submodule)

If the modules still cannot be imported, then it is probably incompatible with an existing configuration.

After the import is complete, you should run a consistency check and eliminate any inconsistencies found.

7 Synchronous Operation of Multiple CPUs (Multicomputing)

7.1 What You Should Know About Multicomputing

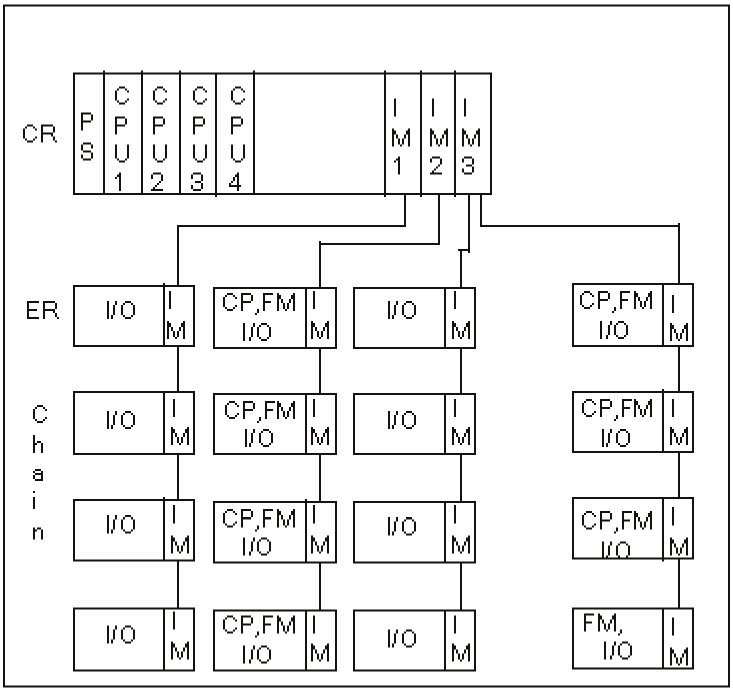
What is Multicomputing Operation?

Multicomputing means the simultaneous operation of more than one CPU with multicomputing capability (up to a maximum of four) in one central rack of the S7-400.

The participating CPUs automatically change their operating modes synchronously, meaning the CPUs start up together and go into STOP mode together. The user program for each CPU executes independently of the user programs in the other CPUs. This means control tasks can be run in parallel.

Example

The following figure shows a programmable controller that will operate in multicomputing mode. Each CPU can access the modules assigned to it (FM, CP, SM).



Difference Between Multicomputing and Operation in Segmented Racks

Simultaneous unsynchronized operation of more than one CPU in a segmented rack CR2 (physically segmented, cannot be set by user) is also possible. This is, however, not multicomputing. The CPUs in a segmented rack form their own independent subsystem and behave like single processors. There is no shared address area.

"Multicomputing mode" and "unsynchronized operation in a segmented rack" at the same time is not possible.

7.1.1 Special Features of Multicomputing

Slot Rules

In multicomputing mode up to four CPUs can be inserted simultaneously in a central rack in any order.

If you use CPUs that can manage only module start addresses that are divisible by four (generally CPUs before 10/98), you must follow this rule for **all** configured CPUs when you assign the addresses. The rule applies to a case where you are also using CPUs that permit byte-by-byte assignment of module start addresses in single-computing operation.

Bus Connection

The CPUs are interconnected via the communication bus (corresponds to a connection via MPI).

Behavior During Startup and Operation

During startup the CPUs in multicomputing operation check automatically whether they can synchronize. Synchronization is only possible:

- If all (and only) the configured CPUs are inserted and not defective
- If correct configuration data (SDBs) were created and downloaded for all inserted CPUs.

  If one of these prerequisites is not met, the event is entered in the diagnostic buffer under the ID 0x49A4. You will find explanations of the event IDs in the reference online help on standard and system functions (SFBs/SFCs).

When the CPUs exit STOP mode, the startup types are compared (COLD RESTART/WARM RESTART/HOT RESTART). This ensures that all the CPUs in the programmable controller execute the same type of startup and all CPUs have the same operating mode.

Address and Interrupt Assignment

In multicomputing mode the individual CPUs can access the modules assigned to them during configuration with STEP 7. The address area of a module is always exclusively assigned to a CPU.

All CPUs share the same address space; this means that the logical address of a module is only present once in the multi-computing station.

An interrupt input is assigned to every CPU. Interrupts received at this input cannot be received by the other CPUs. The assignment of the interrupt line is made automatically during parameter assignment of the modules.

The following rules apply to interrupt servicing:

- Hardware interrupts and diagnostic interrupts are sent to only one CPU.
- If there is a module failure, the interrupt is serviced by the CPU to which the module was assigned with STEP 7.
- When a CP is inserted or removed and the "Address setting for LOCK/UNLOCK with FETCH/WRITE, SPEED Send/Receive" check box exists in the "Addresses" tab of the Properties dialog, then the following applies:
  - If the check box is selected, a remove/insert interrupt is reported only on the assigned CPU and OB83 is started.
  - If the check box is not selected, a remove/insert interrupt is reported all CPUs and OB83 is started.
- If a rack fails, OB86 is called on every CPU.

You will find more detailed information on OB86 in the reference online help on organization blocks.

7.1.2 When to Use Multicomputing

Multicomputing has advantages in the following situations:

- When your user program is too large for one CPU and memory is used up, distribute your program among more than one CPU.
- If part of your system must be processed quickly, take these program sections out of the main program and run them on a separate fast CPU.
- If your system consists of various parts that can be clearly delineated and can be controlled relatively autonomously, run the program for system section 1 on CPU 1 and system section 2 on CPU 2 etc.

7.2 Configuring Multicomputing Operation

Setting Multicomputing Operation

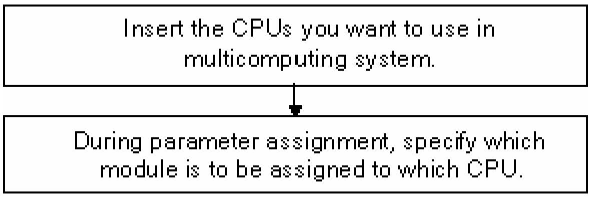
Multicomputing operation results implicitly when you insert a second (third or fourth) multicomputing CPU in a rack suitable for this kind of operation (for example, the rack UR1). When a module is selected, the info text in the "Hardware Catalog" window tells you whether the CPU has multicomputing capability.

Requirements

Before you can configure modules in your programmable controller for multicomputing mode, the following conditions must be satisfied:

- You must have set up your programmable controller as described in the "S7-400, M7-400 Programmable Controllers, Hardware and Installation" manual.
- You must have opened the configuration table in the project window by double-clicking the "Hardware" object.
- You must have arranged a rack in the station window and the rack is shown as being open (the rack slots are visible).

Basic Procedure



Insert the CPUs you want to use in multicomputing system.

During parameter assignment, specify which module is to be assigned to which CPU.

Note when Downloading and Uploading

Only the complete station configuration should be downloaded to all CPUs. This avoids inconsistent configurations.

When uploading to a programming device, the station configuration is loaded from all programmable modules, one after another (CPU by CPU). You have the option of aborting the upload process even if not all the configuration data (SDBs) were uploaded. In this case, parameter assignment information will be lost.

7.2.1 Configuring Modules for Multicomputing Operation

To configure the programmable controller in multicomputing mode, proceed as follows:

1. Use the drag & drop function to copy the CPUs you want to work in multicomputing operation from the "Hardware Catalog" window to the appropriate rows of the rack.
2. Double-click on a CPU and set the CPU number in the "Multicomputing" tab (when you insert the CPU, the CPU numbers are assigned automatically in ascending order).
3. For all modules to be assigned to CPU1, proceed as follows:
   - Arrange the modules at the intended position in the rack.
   - Double-click on the module and select the "Addresses" tab.
   - In the "CPU No." box select CPU 1.

     Note: The CPU assignment is displayed for modules that can trigger interrupts in the "Inputs" or "Outputs" tab as the "Target CPU for interrupt."
4. Repeat the steps listed under 3 for the modules that are to be assigned to the remaining CPUs.

7.2.2 Displaying the CPU Assignment

You have a choice of two methods of highlighting the modules that are assigned to a specific CPU:

- Select the menu command **View > Filter > CPU No. x - Modules** (x = CPU number).

  All module names in the configuration table that are not assigned to this CPU are grayed out (exception: distributed I/O modules, interface modules, and power supplies).
- Select the relevant CPU and select the pop-up menu command **Filter Assigned Modules**.

Note

The filter you set has no effect on the print function or the "Address Overview" dialog box.

You can modify the CPU assignment via the "Addresses" tab (exception: interface modules and power supplies).

7.2.3 Changing the CPU Number

If you have inserted several CPUs and want to change the CPU number, proceed as follows:

1. If four **CPUs** are inserted: Delete one of the inserted CPUs.
   If two or three CPUs are inserted: Continue with the next step.
2. Double-click the CPU whose number you want to change.
3. Select the "Multicomputing" tab.
4. Select the required CPU number.

7.3 Programming CPUs

Programming

Programming for the multicomputing mode is essentially the same as programming a single CPU.

Extra steps are, however, necessary if you want to synchronize the CPUs so that they react to events together.

Calling SFC35

If you want all the CPUs to react to events (for example, interrupts) together in the multicomputing mode, you program an SFC35 "MP_ALM" call. Calling SFC35 triggers a multicomputing interrupt that causes a synchronized request for OB60 on all CPUs. This OB contains local variables that specify the triggering event in greater detail.

When SFC35 is called, the information about the events is transferred to all CPUs in a job identifier. The job identifier allows 16 different events to be distinguished.

When they service the multicomputing interrupt, both the sending user program and the user programs on the other CPUs check whether or not they recognize the job and then react as programmed.

You can call SFC35 at any point in your program. Since the call is only of practical use in RUN mode, the multicomputing interrupt is suppressed if it is triggered in STARTUP mode.

The multicomputing interrupt can only be triggered again after the current multicomputing interrupt has been serviced (acknowledged).

You will find more detailed information on SFC35 in the reference online help on SFBs/SFCs.

Programming OB60

You can program a specific OB60 for each separate CPU and download it to the CPU. This means that the execution times can differ from CPU to CPU. This leads to the following behavior:

- The interrupted priority classes on the CPUs are continued at different times.
- A multicomputing interrupt is not serviced if it occurs during the execution of OB60 on any of the CPUs. A message is, however, generated and you can check this and react accordingly.

If OB60 is not loaded on one of the CPUs, this CPU returns immediately to the last priority class and continues the program there.

You will find more detailed information on OB86 in the reference online help on organization blocks.

8 Modifying the System during Operation (CiR)

8.1 Modifying the System during Operation via CiR

CiR (Configuration in RUN) represents a method you can use to modify a system that is operated with a standard S7-400 CPU or an S7-400-H-CPU in single operation. Modifications are made while your system is in operation, that is, except for a small interval of at most 2.5 seconds, your CPU will remain in RUN mode.

For more information, refer to the "Modifying the System During Operation via CiR" manual. It is also available electronically as a PDF file.

9 Configuring H-Systems

Overview

H-systems ("high-availability" systems) are fault-tolerant automation systems that are employed to increase system availability and thus reduce production stops and downtimes.

For more information, refer to the "Automation System S7-400H - Fault-tolerant Systems" manual.

10 Networking Stations

10.1 Networking Stations within a Project

Context: Network Configuration and STEP 7 Projects

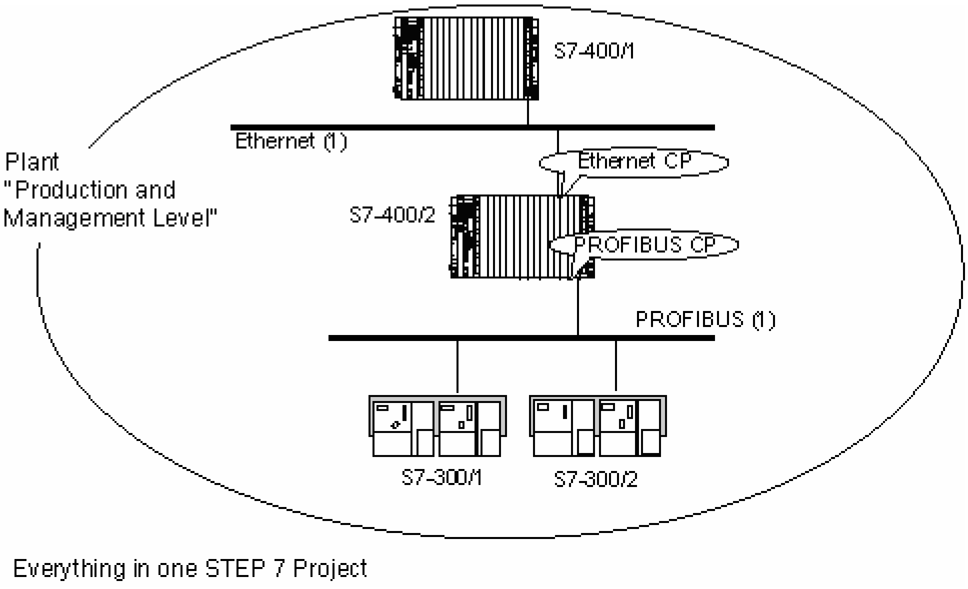
Subnets lie directly beneath a project in the object hierarchy and can therefore only be managed within a project. You can, however, connect nodes together that were configured in different projects.

If possible, you should create and configure nodes that you want to connect in networks in the same project.

Subnets and Stations

You can create the subnets and stations in a STEP 7 project and then configure the stations for communication very easily.

As a result of the different tasks of the stations or the fact that the plant has increased in size and scope, it may be necessary to run a number of subnets. These subnets can also be managed in one project. A station can be assigned to a number of subnets by assigning the communication nodes (for example, CPs) accordingly.



10.2 Properties of Subnets and Communication Nodes

Setting the Properties of Subnets and Communication Nodes in a Project

It does not matter whether you are intending to communicate in the network using global data or communication connections: the basis for communication is always a configured network.

With STEP 7:

- You create a graphic view of your network (comprising one or more subnets)
- You set the subnet properties/parameters for each subnet
- You set the node properties for each networked module
- You document your network configuration.

The following table shows how you are given support by STEP 7 when configuring your communication task.

Communication Method	Configured How?	Remarks
PROFIBUS-DP	Configuring the hardware	Also possible in NetPro
Actuator-sensor interface (AS-i)	Configuring the hardware	Linked to S7 stations via the DP/AS-i Link
Communication by means of non-configured connections (S7 Standard communication)	Configuring the hardware	Set the properties of the MPI subnet and the MPI nodes. The user program provides special system functions for non-configured connections.
Communication by means of configured connections (S7 communication)	NetPro (Configuring Networks and Connections)	S7 and point-to-point (PTP) connections can be configured with the STEP 7 Standard package. Optional packages are required for other connection types (for example, FMS for PROFIBUS).
Global data communication	Defining Global Data	Set the properties of the MPI subnet and the MPI nodes and configure address areas for data exchange in the GD table

10.3 Rules for Network Configuration

You should observe the following rules for configuring networks:

Every node in a subnet must have a different node address.

CPUs are shipped with the default node address 2. However, you can only use this address once in a subnet, so you will have to change the default address for any other CPUs.

For S7-300 stations, the following applies: When planning the MPI addresses for a number of CPUs you must leave "MPI address gaps" for FMs and CPs with their own MPI addresses to avoid double address assignments.

Only when all the modules in a subnet have unique addresses and when your actual structure matches the network configuration you have created should you load the settings via the network.

Assigning MPI Addresses

- Assign the MPI addresses in ascending order.
- Reserve the MPI address 0 for a programming device.
- You can connect up to 126 (addressable) nodes in an MPI network; up to 8 nodes for a transmission rate of 19.2 Kbps.
- All MPI addresses in an MPI subnet must be unique.

Refer to the S7-300 or S7-400 Hardware Installation Manuals for other rules concerning the installation of a network.

Assigning PROFIBUS Addresses

- Every DP master and DP slave in the PROFIBUS network must be assigned a unique PROFIBUS address in the range 0 through 125.
- Assign the PROFIBUS addresses in ascending order.
- Reserve the "0" PROFIBUS address for a programming device that you later connect briefly to the PROFIBUS network for service purposes.

10.4 Ethernet Address Assignment

To configure an Ethernet CP, you must assign a MAC address or IP address to the Ethernet interface.

You can find extensive information on Ethernet CPs in the SIMATIC NCM documentation. In this documentation you can find principal information on how to configure an Ethernet partner in STEP 7.

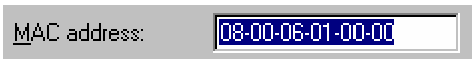
Configuring Partners and Determining the Interface Properties

1. In the opened hardware configuration, select an Ethernet CP from the hardware catalog. Drag and drop it to a suitable row of the configuration table.
2. Double-click on the CP symbol in the configuration table
3. In the "General" tab, click on "Properties" (this button is located in the area of the interface parameters).
4. Establish a network assignment, that is, in the "Subnet" field, highlight an existing Ethernet subnet or click on "New" to create a new Ethernet subnet.
5. Enter the MAC address or IP address in the "Parameter" tab.
   The appearance of this part of the tab depends on the type of CP.

MAC Address

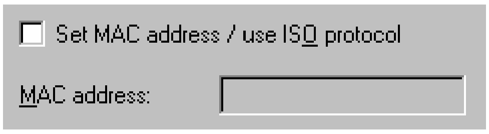
Every Ethernet CP requires a unique MAC address that must be entered when a CP is configured. As a rule, the manufacturer has imprinted this address on the module.

The box looks as follows for CP requiring the input of a MAC address:



MAC address: 08-00-06-01-00-00

The box looks as follows for new CPs equipped with a fixed factory set MAC address - and therefore not requiring the input of a MAC address:



☐ Set MAC address / use ISO protocol

MAC address:

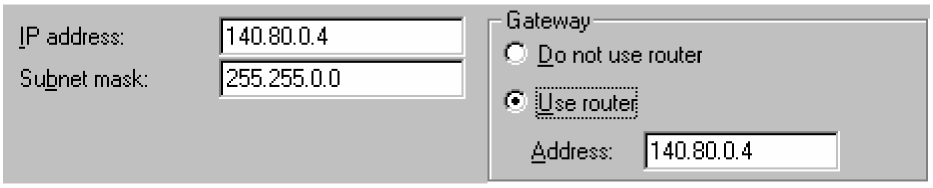
You must check the checkbox and enter the module MAC address only if you are using the **ISO** protocol (for example, for ISO network-independent transport connections), or if you are using the ISO as well as the TCP/IP protocol.

You must not activate the checkbox if you want to configure **only** communication types that require **TCP/IP** protocol (TCP connections, ISO-over-TCP connections, UDP connections). In this case, you cannot enter a MAC address and the address that is imprinted on the module is maintained.

IP Address

The IP parameters are only visible if the Ethernet CP supports the TCP/IP protocol.

The IP address consists of 4 decimal numbers, value range from 0 to 255. The decimal numbers are separated by a dot.


IP address: 140.80.0.4
Subnet mask: 255.255.0.0
Gateway
○ Do not use router
● Use router
Address: 140.80.0.4

The IP address consists of

- The (sub)net address
- The partner address (generally referred to as host or network node)

The subnet mask separates those two addresses. It determines the IP address sections used to address the network and the node IP.

The set bits of the subnet mask determine the network component of the IP address.

In the example above:

Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000

Meaning: the first 2 bytes of the IP address determine the subnet - that is, 140.80. The last two bytes address the node - that is, 2.

Generally valid is:

- The network address is the result of the logic operation IP address AND subnet mask.
- The node address is the result of the logic operation IP address AND NOT subnet mask.

Relationship between the IP address and the default subnet mask

The relationship between the IP address areas and the so-called "Default Subnet Masks" is defined. The first decimal of the (left to right) determines the (binary) structure of the default, that is, the number of "1" values as follows:

IP address (dec.)	IP address (bin.)	Address class	Default subnet mask
0 to 126	0xxxxxxx.xxxxxxxx....	A	255.0.0.0
128 to 191	10xxxxxx.xxxxxxxx...	B	255.255.0.0
192 to 223	110xxxxx.xxxxxxxx...	C	255.255.255.0

Note

The first decimal number of the IP address can also have a value between 224 and 255 (address class D etc). However, we recommend you do not use these values, as STEP 7 does not check them.

Forming further subnet masks

The subnet mask allows you to extend the structure of a subnet that is assigned to address class A, B or C and to create "private" subnets by setting the least significant bits of the subnet mask to "1". The number of "private" networks will double with each bit that is set to "1" and reduces the number of network nodes to half. Seen from the outside this still network has the same effect as a single network.

Example:

In an address class B subnet (e.g.. IP address 129.80.xxx.xxx) you change the default subnet mask as follows:

Masks	Decimal	Binary
Default subnet mask	255.255.0.0	11111111.11111111.00000000.00000000
Subnet mask	255.255.128.0	11111111.11111111.10000000.00000000

Result:

All nodes at the addresses 129.80.001.xxx to 129.80.127.xxx are assigned to the same subnet, whereas all nodes at the addresses 129.80.128.xxx to 129.80.255.xxx are assigned to a different subnet.

Network gateway (Router)

Network gateways (routers) interconnect subnets. An IP datagram sent to another network must first be passed on to a router. To enable this feature, you must enter the router address for each one of the network nodes.

The IP address of a subnet node and of the network gateway (router) may only differ at the "0" bits of the subnet mask.

10.5 Importing and Exporting Stations in the Network View

Introduction

As of STEP 7 V5.1, Service Pack 1, in the network view, you can export and import as a text file (ASCII file) those station configurations that have network data but do not have connection data.

If you export several stations of a STEP 7 project that are networked with each other and then import them into another project, these stations are again networked with each other as they were in the original project.

Applications

- Can be distributed by means of electronic media (for example, e-mail)
- Can be read into future STEP 7 versions
- Export file can be printed out with word processing systems or processed further for documentation purposes

Which Network Objects Can Be Exported Or Imported?

You can export and import SIMATIC 300, SIMATIC 400, SIMATIC 400H, and SIMATIC PC stations.

What Is Exported/Imported?

With import and export from the network view, data can be exported/imported that are required for configuration and parameter assignment of the modules, as well as the network configuration.

The network configuration includes:

- Interface parameter assignments (for example, subnet screen and router settings of an Industrial Ethernet interface, bus parameters of a PROFIBUS interface)
- Assignment to the subnets
- Connections

Not included:

- Data that are managed by means of other applications (for example, programs, global data)
- Connections
- CPU password assigned as parameter
- Data that apply to more than one station (for example, the linking of intelligent DP slaves or configuration for direct data exchange)

Note

If your configuration contains modules from older option packages, it is possible that not all data of the module are included in the export function. In this case, check to see if the module data are complete after the import is finished.

Export Files

A text file (*.cfg) is created for each exported station. The default name is "[station name].cfg".

For each station, you can set separately during export what is stored in the exported text file and in what form the information is stored (menu command **Edit > Export**):

- Readable or compact form

  Important: If you are exporting the station configuration in order to read it in with other STEP 7 versions, select the option "**Compact**"!
- Name of the file (*.cfg) for each individual station can be anything you want
- With or without symbols
- Preset values for module parameters can be left out optionally (STEP 7 "recognizes" preset values and supplements them during import from the internal module knowledge pool)
- Option "Export Subnets" - by selecting this option, you can also read in the station configuration with older STEP 7 versions (STEP 7 => V5.0).
- Option "Export Connections" **-** you can select this option to download the station configuration with older STEP 7 versions (STEP 7 => V5.0).

In order to simplify handling the import procedure already during export, you can import all exported stations together later with the option "**With Reference File.**" To do this, you must select a name for this reference file (also a *.cfg file) that contains references to all the stations that were exported together. When you select this file later during import, all the stations that were exported together are imported again automatically.

Caution

If you export a station configuration with symbols, you can no longer import this file with STEP 7 V5, SP 1, or an older STEP 7 version.

Procedure (Export)

1. Open the network view or save the network configuration that you just edited (menu command **Network > Save**).
2. Select one or more stations that you want to export.
   The Export dialog box opens. In this dialog box, you can select from a list the stations of the project to be exported.
3. Select the menu command **Edit > Export.**
4. In the dialog box that appears, enter the path and name of the export files, the format, and other options.
5. Format and options can be customized for each station separately.
   Select the "Compact" format if you intend to use the export file for other STEP 7 versions.
6. Acknowledge the settings with OK.

Procedure (Import)

1. With the network view open, select the menu command **Edit > Import**.
2. In the dialog box that appears, navigate to the text file that you want to import.
   If you exported several stations together by using the option "**With Reference File,**" select only this reference file to import all stations with their network assignment together again.
3. Acknowledge the settings with OK.
   During import, STEP 7 checks the imported files for errors and consistency and outputs messages.

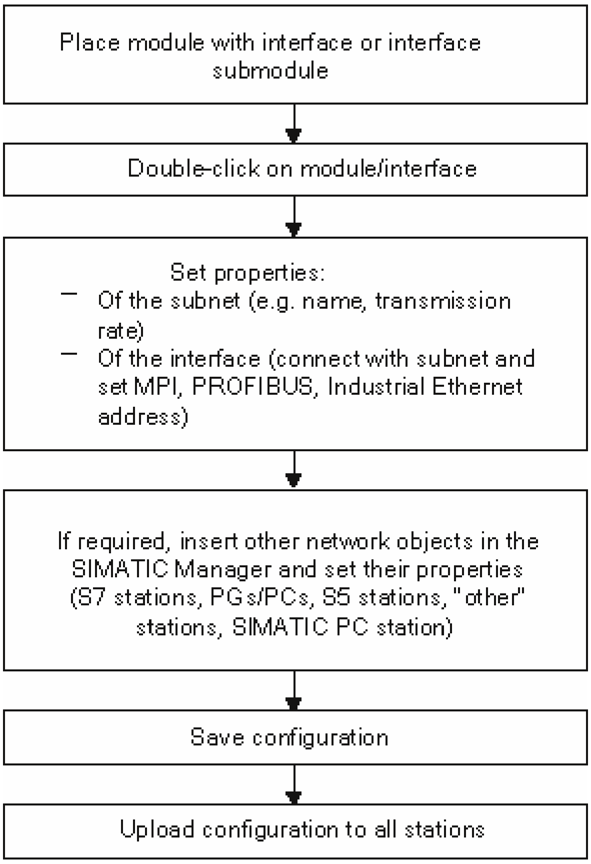
If both connection partners are imported into a project, STEP 7 will then attempt to restore the connections between these partners.

10.6 How to Configure and Save a Subnet

10.6.1 Procedure for Configuring a Subnet

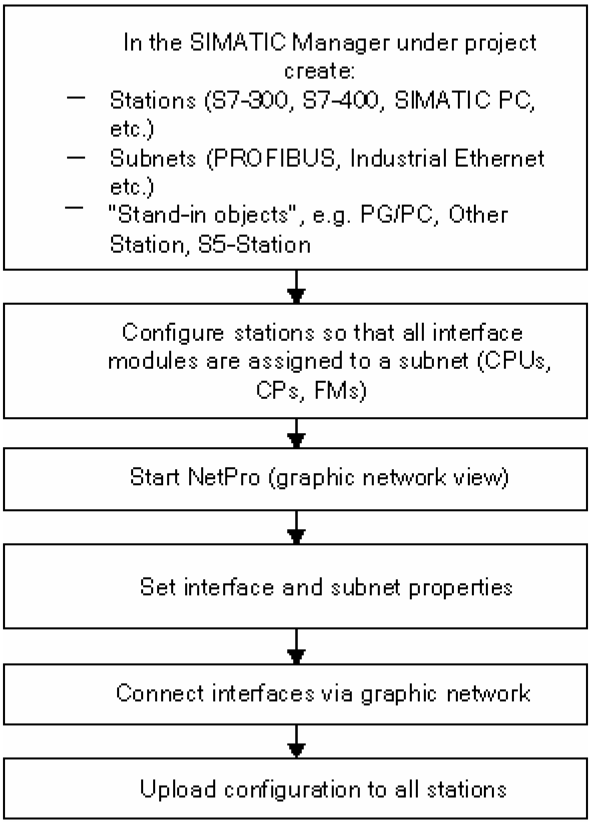
Method 1: Configuring Hardware

You have the possibility of creating subnets and connecting modules (or rather, their interfaces) to a subnet when you configure a station.


Place module with interface or interface submodule
Double-click on module/interface
Set properties:
– Of the subnet (e.g. name, transmission rate)
– Of the interface (connect with subnet and set MPI, PROFIBUS, Industrial Ethernet address)
If required, insert other network objects in the SIMATIC Manager and set their properties (S7 stations, PGs/PCs, S5 stations, "other" stations, SIMATIC PC station)
Save configuration
Upload configuration to all stations

Method 2: Configuring Networks

For complex networked systems it is more advantageous to work in the network view.



In the SIMATIC Manager under project create:

- Stations (S7-300, S7-400, SIMATIC PC, etc.)
- Subnets (PROFIBUS, Industrial Ethernet etc.)
- "Stand-in objects", e.g. PG/PC, Other Station, S5-Station

Configure stations so that all interface modules are assigned to a subnet (CPUs, CPs, FMs)

Start NetPro (graphic network view)

Set interface and subnet properties

Connect interfaces via graphic network

Upload configuration to all stations

Expanding the Network Configuration in NetPro

You have the possibility in NetPro of inserting all network objects such as subnets or stations by dragging them from a catalog and dropping them in the network view.

What else is to be done after you have inserted the objects:

- Set the object properties by double-clicking the objects
- For an inserted station: start the Hardware Configuration application by double-clicking the station and place the modules.

Opening the Graphic Network View (Starting NetPro)

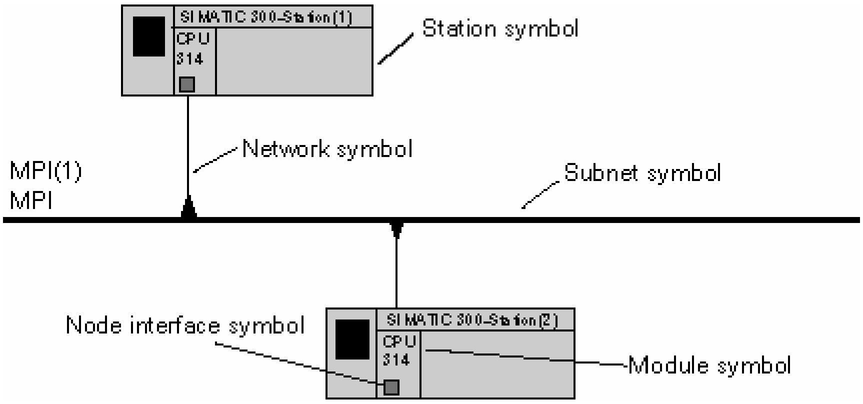
You can use the following methods to start the network configuration application:

From the SIMATIC Manager	From Configuring Hardware
1st Open the project 2nd Double-click a subnet symbol (having created a subnet using the menu command **Insert > Subnet > ...** if necessary) Alternatively you can also double-click the "Connections" object (icon found under a module that is a connection end point, for example, a CPU). In this case the connection table for the module is opened for editing when NetPro is started.	3. Menu command Options > Configure Network

Example of a Graphic Network View

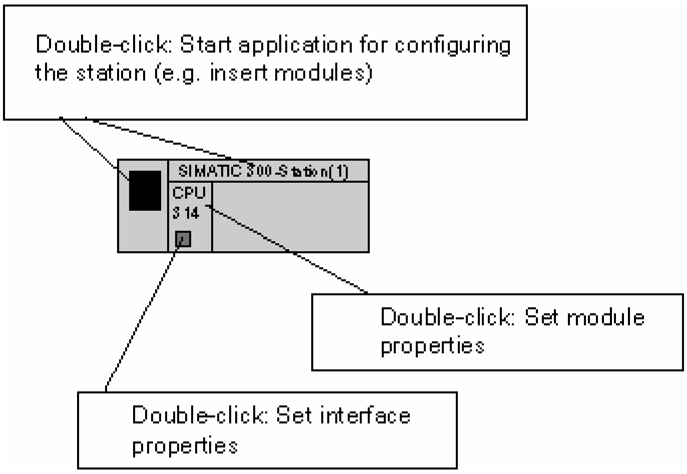
When you have opened the network configuration application, the window for the graphic view of the network is displayed. The first time you select it, the following are visible:

- All the existing subnets created in the project
- All the existing stations configured in the project



Editing a Station in NetPro

If you double-click an area of the station symbol you can further edit the station:


Double-click: Start application for configuring the station (e.g. insert modules)
SIMATIC 300-Station(1)
CPU
314
Double-click: Set module properties
Double-click: Set interface properties

Color Coding in the Network View

After their configurations have been changed (in NetPro or in HW Config), S7 stations are color-coded to indicate their status:

- An orange-colored S7 station means that the configuration was changed (i.e. that the changes have to be compiled).
- An red-colored S7 station means that the changed configuration is not consistent and thus cannot be compiled. In this case, you will have to eliminate the errors detected and logged during a consistency check.
  An inconsistent station is colored red only after a consistency check has been carried out; that is, after you have selected the **Network > Check Consistency** or **Network > Save and Compile** menu command.
  The red color coding is not saved when the station is saved. This means that this color coding will be current and complete only after a consistency check has been carried out again.

  STEP 7 does not detect every need for a compilation after the configuration has been changed:
- If more than one master is connected to a PROFIBUS subnet and a slave that causes a change in the bus parameters is being connected to the subnet: The need for a compilation is only detected for the station of the master system to which the newly added slave is assigned. The other master system that is also connected to the same PROFIBUS subnet will not be color coded.
- If you change the node address of an interface in an S7 station and this station functions as a router (S7 routing) for accesses by programming devices (PG):
  In this case, the other S7 stations on the subnet will also have to be compiled and downloaded so that they receive the routing information. However, these stations will not be color coded.

10.6.2 Creating and Assigning Parameters to a New Subnet

Requirement

NetPro must be open.

Note

STEP 7 automatically and centrally sets consistent subnet properties (e.g. transmission rate) for all nodes in a given subnet.

If you set or modify subnet properties in STEP 7, you must make sure that each node in the system subnet adopts these settings (Downloading a Configuration to a Programmable Controller).

Procedure

1. If the "Catalog" window is not visible:
   Open the "Catalog" window using the menu command **View > Catalog**.
2. Click on "Subnets" in the "Catalog" window.
3. Click on the required subnet, hold the mouse button pressed, and use the drag & drop function to copy the subnet to the window for the graphic network view.
   Invalid positions for subnets in the view window are shown by a ⊘ sign instead of the cursor.
   Result: The subnet appears as a horizontal line.
4. Double-click the symbol for the subnet.
   Result: The properties dialog box for the subnet is displayed.
5. Assign parameters to the subnet.

Tips:

You can open an information window giving details about the subnet properties by holding the mouse on the symbol for the subnet.

10.6.3 Creating and Assigning Parameters to a New Station

Requirement

NetPro must be open.

Procedure

1. If the "Catalog" window is not visible:
   Open the "Catalog" window using the menu command **View > Catalog**.
2. Click on "Stations" in the "Catalog" window.
3. Click on the required type of station, hold the mouse button pressed, and use the drag & drop function to copy the station to the window for the graphic network view.
   Invalid positions for stations in the view window are shown by a ⊘ sign instead of the cursor.
4. Double-click the station (station symbol or station name).
   You **can** now enter the whole hardware configuration for the station and assign its parameters, but you **must** assign the CPU, and any FMs and CPs to a suitable slot. Only these modules can be networked and appear in the graphic network view.
5. Save the hardware configuration.
6. Switch back to NetPro using the taskbar (in Windows).
   Result: The node interfaces available are displayed in the station.

Important:

Before you switch between the station configuration and NetPro, you must save the data you entered, otherwise the database will not be updated.

Tip

You can open an information window giving details about the station properties by holding the mouse on the symbol for the station.

10.6.4 Creating and Assigning Parameters to a Network Connection

Requirement

NetPro must be open and the existing configured stations must be visible.

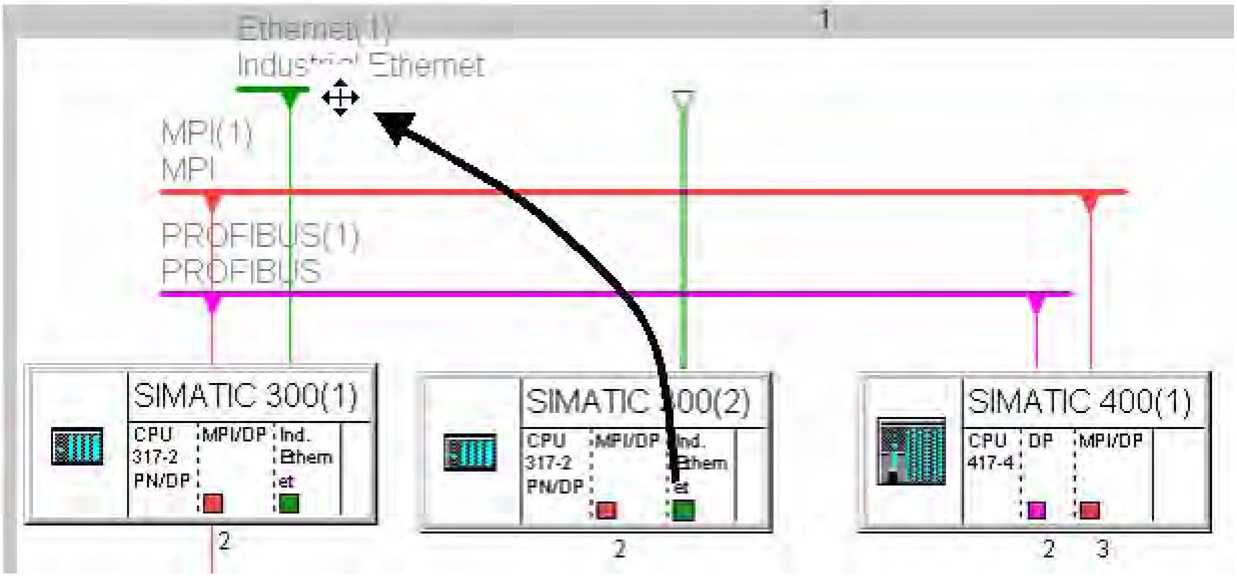
Procedure

1. Click on the symbol for the interface of a node (■), hold the mouse button pressed, and drag the mouse pointer to the subnet.
   In the case of invalid connection options (for example, connecting an MPI interface to an Ethernet-type subnet), the mouse cursor changes its shape to indicate that the action is prohibited.

   Prohibited action: ⊘

   Result: The network connection appears as a vertical line between the station/DP slave and the subnet. If the **View > Reduced Subnet Lengths** menu command is selected, the subnet automatically increases in length so that the network connection is located vertically above the interface.
2. Double-click the symbol for the network connection or the symbol for the interface.
   Result: The properties dialog box for the subnet node is displayed.
3. Assign the node properties (for example, the name and address of the node).

The following illustration shows the dragging movement of the mouse cursor when the **View > Reduced Subnet Lengths** option is selected.



Tips

You can open an information window giving details about the interface properties (module name, subnet type, and, if networked, the node address) by holding the mouse on the interface symbol.

You can show or hide this brief information ("quick info"). To do so, select the **Options > Customize** menu command. In the "Customize" dialog box that is then displayed, go to the "Editor" tab and select or clear the "Display brief information" check box as required.

10.6.5 Creating and Assigning Parameters to a New DP Slave

Requirement:

- You must have assigned a DP master to a station when you configured the hardware in the configuration table.
- DP slaves are displayed in the network view (if not: select the menu command **View > with DP Slaves**).

Procedure

1. If the "Catalog" window is not visible:
   Open the "Catalog" window using the menu command **View > Catalog**.
2. In the network view, select the DP master in a station to which you want to assign the DP slave.
3. In the "Catalog" window click on the required DP slave (under "PROFIBUS-DP"), hold the mouse button pressed, and use the drag & drop function to copy it to the window for the graphic network view.
   Invalid positions for DP slaves in the view window are shown by a 🛇 sign instead of the cursor.
   Alternatively you can also double-click the required DP slave in the "Catalog" window.
4. In the properties dialog box which opens automatically, assign a node address for the DP slave.
   Result: The DP slave appears in the network view together with its network connection.
5. To assign parameters/set addresses: double-click the DP slave.
   Result: Configuring Hardware is started and the DP slave is selected.
6. Set the properties for the DP slave.

Tips

You can open an information window giving details about the DP slave properties by holding the mouse on the symbol for the DP slave.

To find slaves faster in the NetPro catalog, use the "Find" box in the Catalog. This search tool functions just like the one in the Hardware Catalog in HW Config.

10.6.6 Creating and Assigning Parameters to Programming Devices/PCs, 'Other' Stations, and S5 Stations

Overview

What do you do with network nodes that cannot be configured in the current STEP 7 project such as programming devices (PGs), operator stations (OSs), devices made by other manufacturers with their own configuration tool, or S5 devices?

These devices are represented in NetPro by objects such as PG/PC, "other station," and S5 station.

Selecting the Correct Object

The following table shows which object should be inserted in which case:

Object	Purpose	Remarks
PG/PC	To represent in the network view your "own" programming device from which you want to access every node in the subnet online.	Use the "Assignment" tab to establish an assignment between your PG/PC (programming device) and the "PG/PC" object inserted in NetPro. In NetPro the symbol for the PG/PC is specially highlighted.
	For PGs/PCs that are the target for an S7 connection.	For PGs/PCs with S7-SAPI interface
SIMATIC PC station	For PC stations that are the end point of a (two-way) S7 connection; also suitable for redundant S7 connections. For WinLC as of V3.0	The end point of a connection in SIMATIC PC stations is an application such as S7-SAPI ** or WinCC. For a SIMATIC PC station a number of connection end points can be configured.
S5 station	For S5 stations in the subnet	-
Other station	For devices made by other manufacturers which are connected to the subnet.	-
* SIMATIC NET products on CD up to 10/98.		
** SIMATIC NET Products on CD as from 10/98, also refer to the Product Information on this CD or S7-REDCONNECT		

Procedure

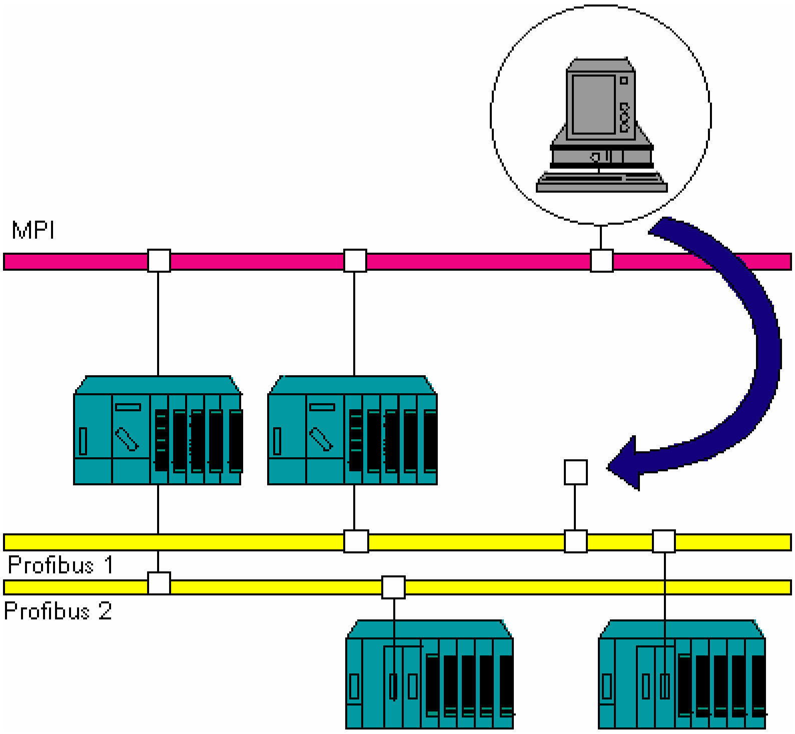
1. If the "Catalog" window is not visible:
   Open the "Catalog" window using the menu command **View > Catalog**.
2. In the "Catalog" window click on the required object (under "Stations"), hold the mouse button pressed, and use the drag & drop function to copy it to the window for the graphic network view. Invalid positions in the view window are shown by a ⃠ sign instead of the cursor.
   Alternatively you can also double-click the required object in the "Catalog" window.
3. Double-click the object.
   Result: A dialog box with tabs for setting the properties is displayed.
4. Set the properties:
   - For all objects except SIMATIC PC stations: in the "Interfaces" tab, create the type of interface that the real object has (for example, PROFIBUS). Via the "Properties" button, set the node and subnet properties.
     Result: The object receives an interface symbol for every newly created interface.
   - For the "PG/PC" object: In the "Assignment" tab, set an assignment to an existing module parameter assignment (PC card) if necessary. With this assignment you link the "PG/PC" object in the network view with the actual module parameters of your PG/PC. Advantage: If you, for example, change the transmission rate of the subnet, the module parameters for your PG/PC card change automatically.

10.6.7 Taking Connections for Programming Devices/PCs into Account in the Network Configuration

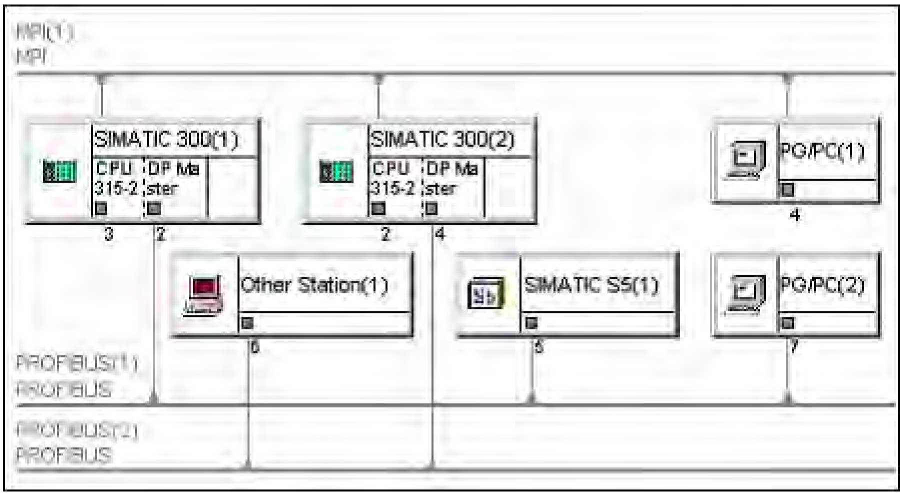
If you have a networked project with a number of different subnets, you can set up a number of wildcards in the network configuration for a PG (PC) you want to connect up later on. The "PG/PC" object in the network view assumes this "wildcard" function.

This allows you to disconnect a programming device from a subnet and connect it to another subnet. You tell STEP 7 about the PG relocation when you use the **"PLC > Assign PG/PC** menu command.

The figure below illustrates the setup:



The connection points in the network view look like this ("PG/PC(1))" and ("PG/PC(2)"):



You can now assign your PG one of the "PG/PC" symbols (to identify the device you intend to use for accessing stations online). Assignment updates the interfaces in the programming device (PG or PC) to match the configured settings. If you change your configured settings (e.g. by modifying the transmission rate or another network property), the interface in your PG or PC will be updated automatically.

Procedure

1. If the PG/PC has already been assigned: Reverse the assignment by selecting the "PG/PC" symbol and activating the **PLC > Remove PG/PC Assignment** menu command. The assigned PG/PC symbol differs from the symbol for non-assigned PGs/PCs.
2. Select a "PG/PC" symbol in the network view to represent the programming device you have connected.
3. Select the **PLC > Assign PG/PC** menu command.
4. Use the "Assignment" tab to assign a set of interface parameters in your programming device (your PG/PC) to an interface of the "PG/PC" symbol.

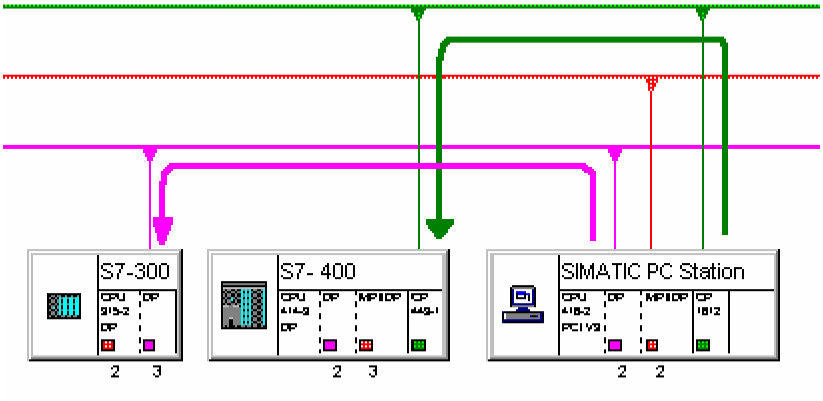
STEP 7 on a SIMATIC PC Station

SIMATIC PC stations as of STEP 7 V5.1, Service pack 2 are capable of routing.

With STEP 7 on a SIMATIC PC station you can reach all nodes online which are connected to this station. In this case, you do not require a PG/PC assignment to reach the nodes of different subnets.

Conditions to make a SIMATIC PC station routing capable:

- To configure the SIMATIC PC station:
  - STEP 7, version 5.1, Service pack 2 or
  - SIMATIC NCM PC, version 5.1 including Service pack 2 (from the SIMATIC NET CD as of 7/2001)
- Runtime conditions for the PC station
  - Installed drivers for CPs: SIMATIC NET CD, as of 7/2001
  - Installed components for WinAC Slot: WinAC Slot 41x, version 3.2



10.6.8 Checking the Consistency of the Network

Before you save, you should check the network configuration for consistency. The following are reported, for example:

- Nodes that are not connected to a subnet
(exception: non-networked MPI nodes)
- Subnets that have only one node
- Inconsistent connections

Requirement

NetPro must be open.

Procedure

- Select the menu command **Network > Check Consistency**.
Result: A window "Outputs for Consistency Check" with hints for configuring conflict-free networks and connections is displayed.
Inconsistent stations are color-coded in red.

Tip

You can select the window containing the results of the last consistency check at any time using the menu command **View > Errors and Warnings**.

Alternative Procedure

1. Select the menu command Network > Save and Compile.
2. In the dialog box, select the option "Check and compile all."

The "Outputs for Consistency Check" Window

Messages and warnings are entered in this window if the consistency check found incorrect project planning/configuration (can refer to the hardware configuration, network or connection planning). A consistency check is carried out during the following actions:

- Menu command **Network > Check Consistency**
- Menu command **Network > Network > Check Cross-Project Consistency**
- Menu command **Network > Save and Compile**
- Download to PLC (consistency check for the stations or connections to which downloading is to be carried out)

Messages in the window "Outputs for Consistency Check" are displayed as **errors**, if no system data (SDBs) could be created during saving and compiling or before downloading to the PLC. Without generated system data it is possible to download the hardware/network and connection configuration cannot be downloaded to the PLC.

Messages in the window "Outputs for Consistency Check" are displayed as **warnings**, if the signaled state allows the generation of system data (SDBs).

If you select a line in the upper section of the output window, this line is repeated in the lower section of the window in word-wrapped text. You do not need to scroll to see all the text.

Selecting a Faulty/Inconsistent Object

Double-click on the corresponding message or warning in the window "Outputs for Consistency Check", or select the **Edit > Go to Error Location** menu command for this window.

Help on a Message/Warning

Select the message or warning and press F1, or select the **Edit > Show Help on Message** menu command.

Saving Messages/Warnings

Select the **File > Save Messages** menu command in the output window.

10.6.9 Saving the Network Configuration

Introduction

To save the connection table, you can choose either of the menu commands **Network > Save** and **Network > Save and Compile**.

Saving

If you have created the network objects in NetPro or changed their properties in NetPro, NetPro saves the following using the menu command **Network > Save:**

- Node addresses
- Subnet properties (such as transmission rate)
- Connections
- Modified module parameters (for example, of CPUs)

Saving and Compiling

When you select the menu command **Network > Save and Compile** you must specify in a dialog box whether you want to compile everything or just the changes:

Independent of the selected option, NetPro checks the consistency of the configuration data throughout the project; any messages are displayed in a separate window.

- "Check and compile all" option
  The loadable system data blocks (SDBs) for the entire network configuration are created; they contain all connections, node addresses, subnet properties, input/output addresses, and module parameters.
- "Compile changes only" option
  Loadable system data blocks (SDBs) for **modified** connections, node addresses, subnet properties, input/output addresses, and module parameters are created.

10.6.10 Tips for Editing the Network Configuration

Displaying the Network Address Overview

If you want to have an overview of all the assigned node addresses, then select a subnet in the network view.

Depending on the subnet type selected, the lower section of the network view will then show a tabular overview of the MPI, PROFIBUS or Ethernet addresses. The remaining columns show the name of the station, the interface and, if a multiproject-oriented network view is open, the project name as well.

Special considerations:

- If an Ethernet subnet is selected, then, depending on the configuration, the IP address, the MAC address or both are shown. If a PROFINET IO system is configured, the device numbers of the nodes are also shown.
- If the multiproject-oriented network view is open and a merged subnet is selected, then the lower section of the window will show all the node addresses for the merged subnet.
- If doubled addresses were issued on the subnet, then they are indicated by an asterisk (*).
- To open the properties sheet for an interface, double-click the corresponding row in the address overview. You cannot change the network address here.

Configuring Connections

If you select a component in the network view that can be an end point of a connection (for example, a CPU), the connection table in which you can configure connections is displayed automatically.

Starting Global Data Configuration

1. In the network view, select an MPI subnet for which you want to configure global data communication.
2. Select the **Options > Define Global Data** menu command.

Result: The GD table for the MPI subnet is opened.

Highlighting the Communication Partners of a Module

If you have already configured connections:

1. Select a programmable module (CPU, FM) in the network view.
2. Select the **View > Highlight > Connections** menu command.

Note: The communication partners of only one programmable module can be highlighted at any one time.

Displaying/Changing the Properties of Components

To display or change the properties of stations or modules, proceed as follows:

1. Select the component (station symbol or module).
2. Select the **Edit > Object Properties** menu command.

Copying Subnets and Stations

1. Select the network objects you want to copy by clicking them with the left mouse button. If you want to copy more than one network object at the same time, select additional network objects with SHIFT + left mouse button.
2. Select the **Edit > Copy** menu command.
3. Click the position in the network view (place the cursor there) where you want to place the copy and select the **Edit > Paste** menu command.

   Note: You can copy individual network objects or whole subnets with network connections, stations, and DP slaves. When copying, remember that every node in a subnet must have a unique node address. You should therefore change the node addresses if necessary.

Deleting Network Connections, Stations, and Subnets

1. Select the symbol for the network connection, the station, the DP slave or the subnet.
2. Select the **Edit > Delete** menu command.
   When you delete a subnet, the stations which were previously connected to the subnet are retained and can be connected to another subnet if required.

Positioning Stations and Subnets

You can move subnets, stations, and DP slaves (with or without a network connection) anywhere in the view window. This means you can replicate your hardware structure visually on the screen.

- Click the subnet or station/DP slave, hold the mouse button down, and use "drag & drop" to move the subnet or station/DP slave to the required position.
  Invalid positions for subnets or stations/DP slaves in the view window are shown by a ⊘ sign instead of the cursor.

You can also move stations/DP slaves that are already connected to a subnet. The network connections of the stations/DP slaves are retained.

Reducing Subnet Lengths

In NetPro, subnets are displayed in a horizontal line that extends "infinitely." To reduce the length of the subnets so that stations can be grouped with their subnets and arranged in a clearer and more understandable manner, select the **View > Reduced Subnet Lengths** menu command. The length shown is determined by the distance between the networked interfaces and is adapted automatically. The subnet will extend a little bit beyond the networked modules.

If the subnet does not yet have a node connected to it, such as after a new subnet has been inserted, the length shown is "infinite" and is dependent on the setting selected in the "View" menu.

Note

If you change the view, connect stations to subnets or arrange subnets and stations, then it may happen that the subnets overlap each other. In this case, you will have to arrange the stations and subnets again.

Arranging the Network View - Assigning DP Slaves to Their DP Masters

In the graphical network view, in order to rearrange a network configuration that has become confusing, you can assign the DP slaves to their respective DP master stations visually:

Requirement: The "with DP Slaves" view is activated (menu command: **View > With DP Slaves**).

Select the **View > Rearrange** menu command (as of STEP 7 V5.1, Service Pack 1).

Selecting a Master System

You can select a whole master system to copy it, for example.

1. Select a master or a slave in the network view.
2. Select **the Edit > Select > Master System** menu command.

Highlighting a Master System

1. Select a DP master or a DP slave in the network view.
2. Select the **View > Highlight > Master System** menu command.

Online Access to Modules

You have access to the following functions via the "PLC" menu:

- Display module information
- Change the operating mode of a module
- Clear/reset a module
- Set the date and time for a module
- Download and upload

10.7 Networking Stations that Represent Network Gateways

10.7.1 Networking Stations that Represent Network Gateways

Overview

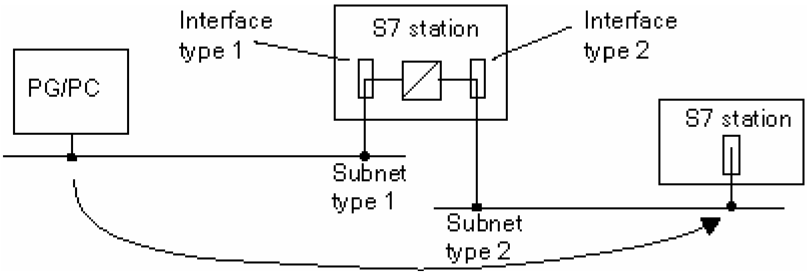
For most programmable logic controller systems, you can operate the programming device exclusively by means of the bus cable (subnet) that is connected to the programmable logic controller directly. For relatively large networked systems, this situation requires that the programming device be connected to various bus cables (subnets) before a remote programmable logic controller can be reached online.

From STEP 7 V5 on, it is possible to access programmable logic controllers online with the programming device/PC beyond the limits of a subnet in order, for example, to download user programs or a hardware configuration or for test and commissioning functions.

The "PG Routing" function makes it possible to reach programmable logic controllers from a fixed location in the system beyond the different subnets without having to reconnect the bus connectors. For this purpose, special "routing tables" are generated automatically for the network gateways during network configuration with STEP 7. These routing tables are special system data and must be loaded to the individual network gateways, that is, S7 CPUs or communication processors (CPs). Afterward, when the programming device goes online, the way to the selected programmable logic controller can be found by means of the network gateways.

Network Gateway

The gateway from a subnet to one or more other subnets lies in a SIMATIC station that has interfaces to the respective subnets.



Requirements

- STEP 7 as of Version 5
- The modules with communication capability (CPUs or CPs) that are supposed to establish network gateways between the subnets must be capable of routing (the info text on the respective component can be found in the hardware catalog).
  As of STEP 7 Version 5.1, service pack 2, SIMATIC PC stations can also be capable of routing.
- All reachable programmable logic controllers or communication partners in a system network must be configured and loaded within an S7 project.
- The desired S7 station is networked and can actually be reached by means of the network gateways.
- The modules must be loaded with the configuration information that contains the current "knowledge" of the entire network configuration for the project. Reason: All the modules involved in the network gateway must receive information about which subnets can be accessed via which routes (routing tables).
- The programming device/PC with which you want to establish an online connection by means of a gateway must be configured and assigned to your programming device in the network configuration.

Additional Information for Network Gateways

In addition to node address, subnet properties, and connections, STEP 7 V5 also generates routing information that must be loaded onto the corresponding modules.

Routing information includes:

- Module interfaces
- Assignment to connected subnets
- The next network gateways in order to be able to access a remote subnet from one of the connected subnets

This information is generated automatically by STEP 7 when the network or station configuration is compiled (menu command: **... > Save and Compile**).

Which Modules or Stations Must Be Loaded After a Change to a Network Configuration?

If you change a configuration as follows ...	... you have to reload
Delete or add a network connection to a station (station equals network gateway)	All network gateways
Modify an interface address on the subnet (station equals network gateway) or plug a module with its own MPI address into an S7-300 station in such a way that the MPI address of a network gateway (next module inserted) is changed	Network gateways on the same subnet
Add or delete a network gateway	All network gateways
Insert a module with a network connection in another slot (station equals network gateway)	All modules for this station
Add a subnet	-
Delete a subnet (where network gateways are configured on this subnet)	All network gateways
Change a subnet ID	If any network gateways are connected to this subnet: All network gateways

S7 Subnet ID for an Online Connection via Network Gateways

If the network configuration together with all the routing information was downloaded to the affected stations, you must also specify an S7 subnet ID to access the remote station.

The S7 subnet ID that is requested by STEP 7 using dialog boxes is formed from two numbers:

- A number for the project
- A number for the subnet

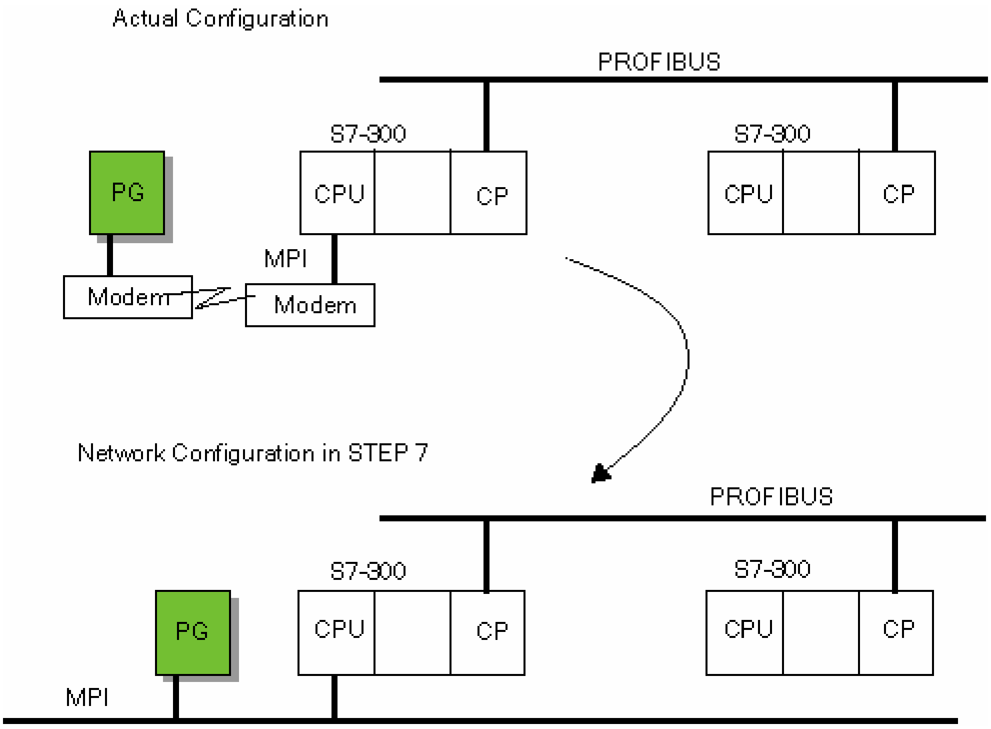
Both numbers can be determined via the properties dialog box of the subnet with an existing network configuration. If you want to go online with a programming device without a consistent project, you must know the subnet ID. The subnet ID is printed out with the network configuration.

10.7.2 Programming Devices / PCs Connected to a Subnet via TeleService or WAN

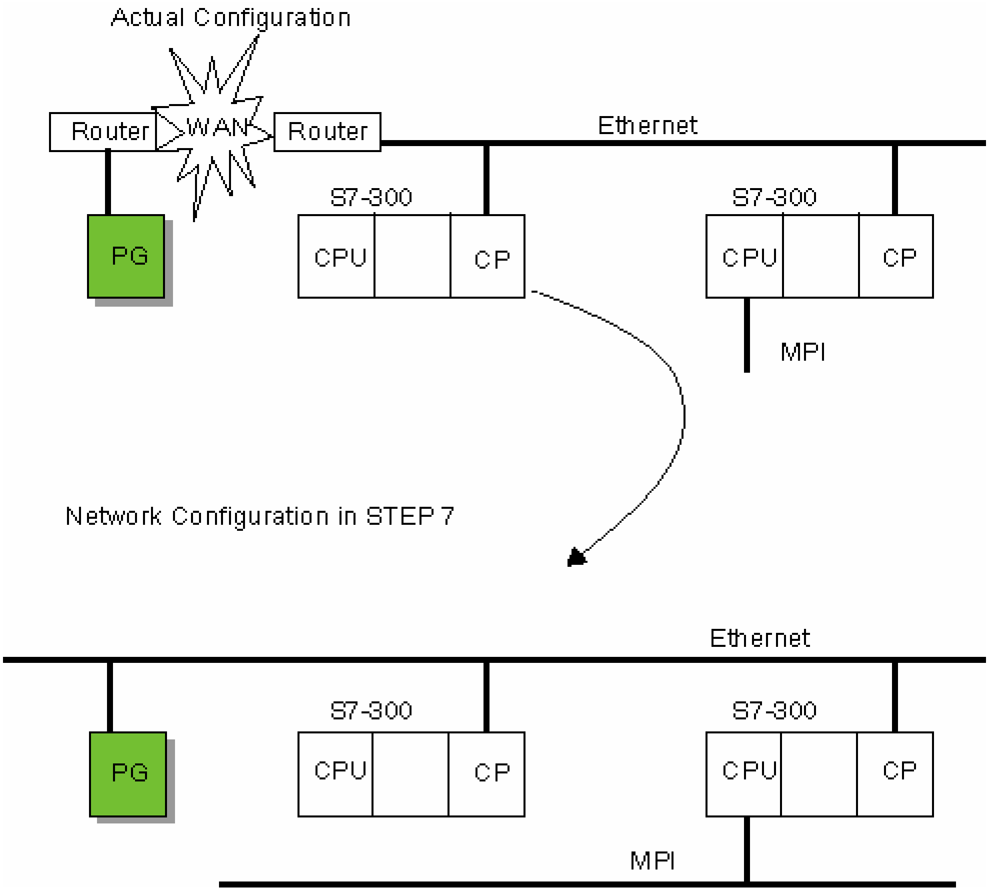
A programming device or PC that accesses nodes in a remote subnet via TeleService or WAN (Wide Area Network) is dealt with as follows in the network configuration:

The "PG/PC" object is connected directly to the remote subnet in the network configuration of STEP 7. The network gateway via a TS adapter or router is not visible in the configuration.

Example: Connecting a Programming Device via TeleService



Example: Connecting a Programming Device via WAN



10.8 Networking Stations from Different Projects

Introduction

In complex networked systems, it may be advantageous to manage the stations in more than project.

As of STEP 7 V5.2, you can configure several projects by using a multiproject. This procedure is recommended for new projects.

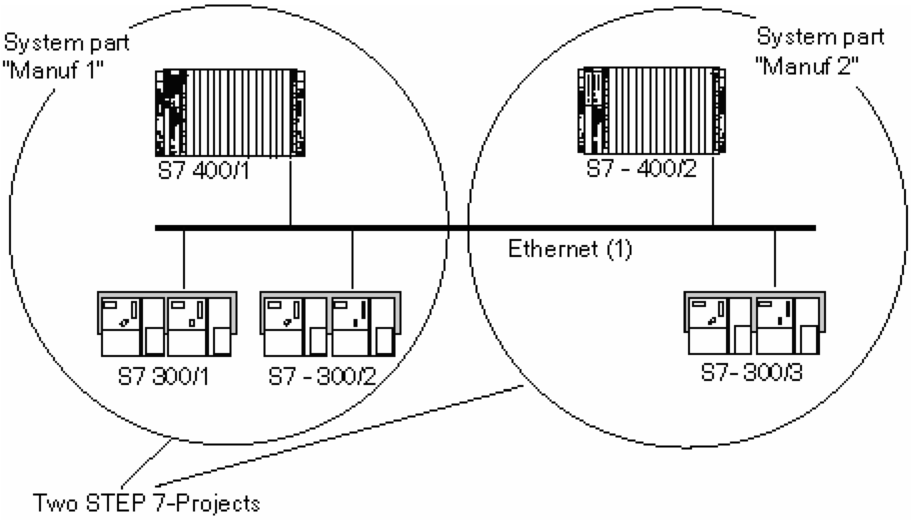
In the figure below, a networked system is divided into two system parts (projects) "Manuf1" and "Manuf2."

Without a multiproject, the following problem arises: for the project "Manuf1," stations that were configured in the project "Manuf2" are unknown.

Configuring Without a Multiproject

- You must insert the symbol "Other Station" in the project "Manuf1" to represent a station in the project "Manuf2." The "Other Station" as a "stand-in object" is restricted to the properties that are relevant for the network view.
- You must program the subnet to which both parts of the system are connected twice in identical form: once in the project "Manuf1" and once in the project "Manuf2."

The responsibility for the consistency of the network data in the different projects lies with you in this case; STEP 7 cannot ensure consistency "beyond project boundaries."



11 Configuring Connections

11.1 Introduction to Configuring Connections

Introduction

Communication connections are always required when you want to exchange data using special communication blocks (SFBs, FBs, or FCs) in the user program.

This section describes how you define connections with *STEP 7*, which particular features you should note, and which communication blocks you can use in the user program.

What is a Connection?

A connection is a logical assignment of two communication partners to execute communications. A connection determines the following:

- The partners involved in communication
- The type of connection (for example, S7, point-to-point, FDL, or ISO transport link)
- Special properties (such as whether a connection remains permanently configured, or whether it is established and disconnected dynamically in the user program, and whether operating mode messages should be sent).

What Happens When You Configure Connections?

When you configure connections, a unique "local ID" is assigned for each connection. Only this local ID is required when you assign parameters to communication blocks. Every programmable module that can be an end point in a connection has its own connection table.

11.2 Configuring Communication with an Ethernet CP

Type of Communication

The Ethernet CP supports the following types of communication, depending on the CP type:

- S7 communication
  S7 communication forms a simple and efficient interface between SIMATIC S7 stations and PGs/PCs using communication function blocks.
  The CP functions as an "S7 communication relay", that passes on the block communication on Industrial Ethernet.
- Open communication services
- SEND-RECEIVE interface
  Depending on the CP type, the SEND-RECEIVE interface allows program-controlled communication on a configured connection from a SIMATIC S7 PLC to another SIMATIC S7 PLC, to a SIMATIC S5 PLC, to PCs/PGs, and to any other station.
- FETCH/WRITE services (Server)
  FETCH/WRITE services (Server) allow direct access to system memory areas of a SIMATIC S7 CPU from a SIMATIC S5 system or from third-party devices.
- HTML Process Control
  With the IT-CP, you use the included functions and HTML pages to enquire about important system data by means of a web browser (see the guide to the IT-CP).
- File management and file access via FTP
  The IT-CP provides additional functions for FTP services.

Communication Services on the SEND/RECEIVE Interface

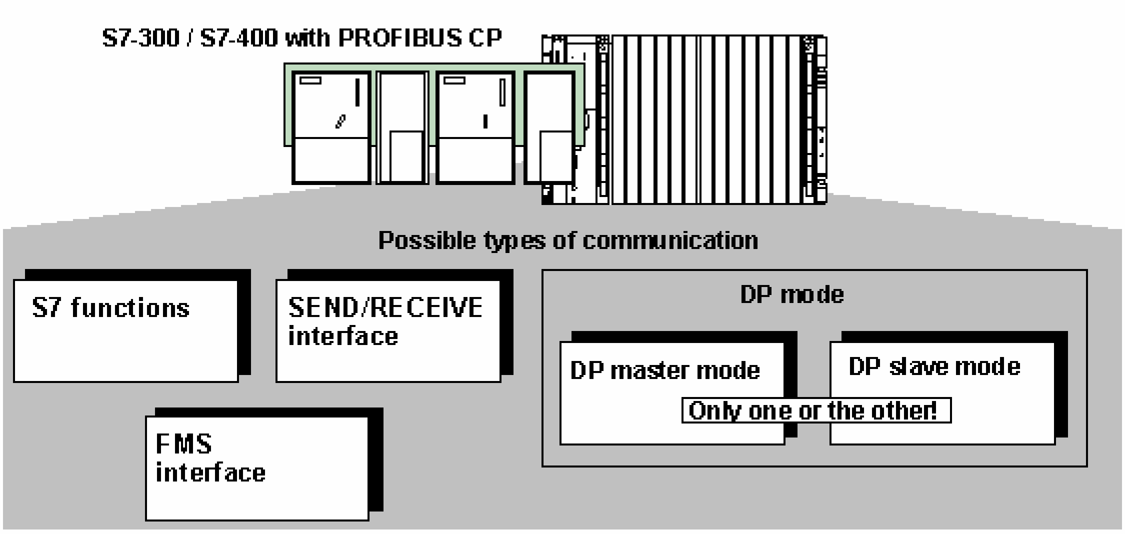
Depending on the CP type, the following communication services are available:

- ISO Transport
  optimized for top performance at the self-contained manufacturing level
- TCP/IP for internet communication with
  - ISO-on-TCP connections (RFC 1006) and
  - UDP datagram service.
- Sending E-Mail
  The controller can send messages depending on process events (see the guide to the IT-CP).

11.3 Configuring Communication with a PROFIBUS CP

Type of Communication

The PROFIBUS CP supports the following types of communication (depending on the CP being used):



- PG/OP communication
  PG/OP communication is used to download programs and configuration data, to perform testing and diagnostics functions as well as for operator control and monitoring of a system via OPs.
- S7 communication
  The S7 communication forms a simple and efficient interface between SIMATIC S7 stations and PGs/PCs using communication function blocks.
  The CP functions as an "S7 communication relay", that passes on the communication on PROFIBUS.
- Open communication services (SEND-RECEIVE interface)
  The SEND/RECEIVE interface allows program-controlled communication on a configured connection from a SIMATIC S7 PLC to another SIMATIC S7 PLC, a SIMATIC S5 PLC and to PCs/PGs.
- Standard communication (FMS interface)
  (complying with EN 50170 Vol. 2; FMS client and server function)
  The FMS interface allows the program-controlled, neutral transmission of structured data via a configured connection from SIMATIC S7 PLCs to devices that support the FMS protocol.
- PROFIBUS DP
  (complying with EN 50170 Vol. 2, DP master or DP slave)
  The distributed peripheral I/Os (DP) allow you to use a large number of analog and digital input/output modules in the immediate vicinity of the process in a distributed configuration.

11.4 What You Should Know About the Different Connection Types

Introduction

The following sections provide a brief overview of the connection types you can configure with STEP 7. The "Communicating with SIMATIC" manual provides a more detailed description of the communications possibilities in SIMATIC.

S7 Connections

S7 connections offer the following features:

- Can be configured in all S7/M7 devices
- Can be used in all types of subnet (MPI, PROFIBUS, Industrial Ethernet)
- When using the SFBs BSEND/BRCV: secure transfer of data between SIMATIC S7/M7-400 stations; for example, exchanging data block contents (up to 64 Kbytes)
- Using CPU 317-2 PN/DP as well as CPU 31x and a CP with the FBs BSEND/BRCV from the SIMATIC_NET_CP library or Standard Library also allows secure transmission of data to S7-300 and S7-400.
- When using the SFBs USEND/URCV: fast, unsecured transfer of data independent of the time processing of the communication partner, for example, for event messages and messages.
- Acknowledgement of data transfer from the communication partner via layer 7 of the ISO reference model

S7 Connections, Redundant

- Properties as for S7 connections; however, restricted to S7 H CPUs as well as SIMATIC PC stations (such as OPC servers) and not on MPI subnets
- Depending on the network topology, at least two connection paths between the connection end points are possible with a redundant S7 connection.

Point-to-Point Connection

The local CP 441 provides the link for the connection between an S7-400 CPU and a communication partner connected via a point-to-point connection. On the CP, a conversion is made to the addressing mechanisms of the selected transfer procedure. For this reason, the point-to-point connection ends in the CP 441 and not at the communication partner as for other connection types.

The number of connections to the CP depends on the set procedure.

FMS Connection

PROFIBUS FMS (Field bus Message Specification) has the following features:

- For the transfer of structured data (FMS variables)
- Fulfills the European standard EN 50170 Vol.2 PROFIBUS
- For open communication with non-Siemens devices on PROFIBUS
- Application on the remote communication partner acknowledges receipt of the data
- Can be ordered in layer 7 of the ISO reference model
- FMS services are provided on the PC as C functions

FDL Connection

PROFIBUS FDL (Field bus Data Link) has the following features:

- For the transfer of data to a communication partner (for example, SIMATIC S5 or PC) that supports sending and receiving in accordance with the SDA function (Send Data with Acknowledge)
- The receipt of data is confirmed by the FDL service of the communication partner with an acknowledgement
- For PROFIBUS subnets only
- Fulfills the European standard EN 50170 Vol.2 PROFIBUS
- Can be ordered in layer 2 of the ISO reference model
- FDL services are provided on the PC as C functions

ISO Transport Connection

The ISO Transport connection has the following features:

- Suitable for large quantities of data owing to "data blocking"
- Allows communication to a partner (for example, SIMATIC S5 or PC) that supports sending and receiving of data in accordance with ISO Transport
- You can use the Send/Receive utilities and the Fetch and Write utilities to transmit data.
- For Industrial Ethernet only
- The receipt of data is confirmed by the ISO Transport service of the communication partner with an acknowledgement
- ISO Transport (ISO 8073 class 4) corresponds to layer 4 of the ISO reference model
- ISO Transport services are provided on the PC as C functions

ISO-on-TCP Connection

The ISO-on-TCP connection has the following features:

- Fulfills the TCP/IP (Transmission Control Protocol/Internet Protocol) standard with the extension RFC 1006 in accordance with layer 4 of the ISO reference model. RFC 1006 describes how the services of ISO layer 4 can be mapped onto TCP
- Allows communication to a partner (for example, SIMATIC S5 or PC) that supports sending and receiving of data in accordance with ISO-on-TCP
- You can use the Send/Receive utilities and the Fetch and Write utilities to transmit data.
- The receipt of data is confirmed with an acknowledgement
- For Industrial Ethernet only
- ISO-on-TCP services are provided on the PC as C functions

TCP Connection

The TCP connection has the following features:

- Complies with the TCP/IP standard (Transmission Control Protocol/Internet Protocol)
- Enables communication to a partner (such as a PC or non-Siemens device) that supports sending or receiving data according to TCP/IP.
- You can transmit data by means of the utilities Send/Receive or Fetch and Write.
- For Industrial Ethernet only
- As a rule, you can use the TCP/IP implementation in the operating system of the PC.

UDP Connection

The UDP (User Datagram Protocol) connection has the following features:

- For Industrial Ethernet (TCP/IP protocol)
- Allows unsecured transfer of contiguous blocks of data between two nodes

E-Mail Connection

The e-mail connection has the following features:

- For Industrial Ethernet (TCP/IP protocol)
- Enables process data, for example, to be sent from data blocks via e-mail using an IT communications processor
- Using the e-mail connection, you define the mail server via which all e-mails sent by an IT communications processor are delivered.

11.5 What You Should Know About Assigning Connection Resources

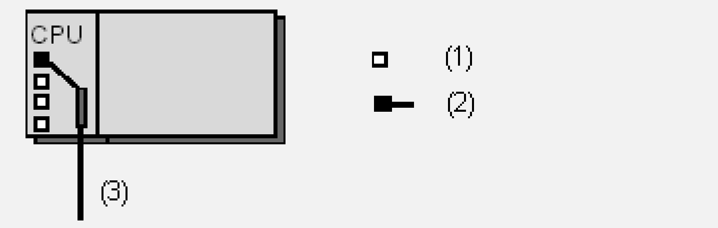
Introduction

At the stations involved, each connection requires connection resources for its endpoint and its transition point (i.e. CP). The actual number of connection resources available depends on the CPU/CP in use.

If all the connection resources for a communication partner occupied, then a new connection will not be able to be established. The following discussion treats each type of communication individually. Within the limits imposed by the amount of resources available, combinations of any type are possible.

S7 Connections

For S7 connections established through the **integrated** MPI-/PROFIBUS DP/PN interface, one connection resource for the end point is occupied at the CPU for each S7 connection. This applies to all S7/M7-300/400 and C7-600 CPUs.



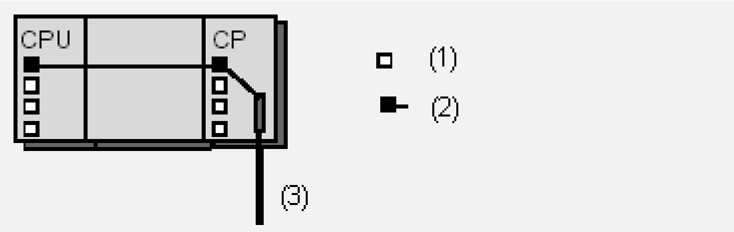
(1) Available connection resource

(2) Assigned connection resource

(3) MPI, PROFIBUS DP or Industrial Ethernet (PROFINET)

For S7 connections established through an **external** CP interface, a connection resource is occupied at the CPU for (the end point) and at the CP (transition point) for each STEP 7 connection. This applies to all S7/M7-300/400 and C7-600-CPUs.

Tip: Connection resources for an S7-300 CPU can be reserved for OP/ programming device (PG) communication and S7 basic communication in the properties dialog of the CPU ("Communication" tab). The already configured S7 connections are displayed there as well ("S7 Communication").



(1) Available connection resource

(2) Assigned connection resource

(3) Industrial Ethernet, PROFIBUS

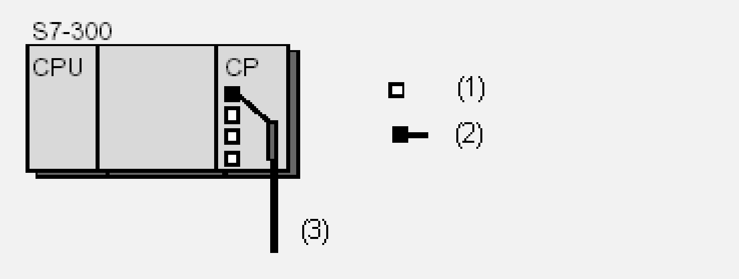
Note

To execute programming device functions on CPs for the S7-400 through the MPI or through the integrated DP interface, two connection resources are needed at the CPU (for two transition points). This requirement must be taken into account in determining the total number of configured STEP 7 connections.

SEND/RECEIVE Interface

Communication through the SEND/RECEIVE interface takes place only through CPs. Each connection (i.e. FDL, ISO-transport, ISO-on-TCP, UDP and TCP connections) takes up one resource at the CP for the end point.

At an S7-300 CPU and a C7-600 CPU, no connection resources are required for the connection.

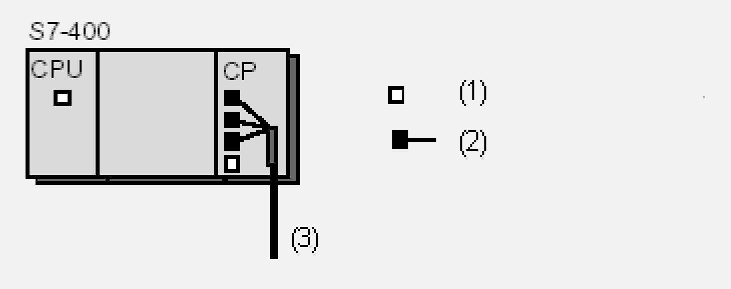


(1) Available connection resource

(2) Assigned connection resource

(3) Industrial Ethernet, PROFIBUS

At a S7-400 CPU, no connection resources are required for SEND/RECEIVE connections (i.e. FDL, ISO-transport, ISO-on-TCP connections.



(1) Available connection resource

(2) Assigned connection resource

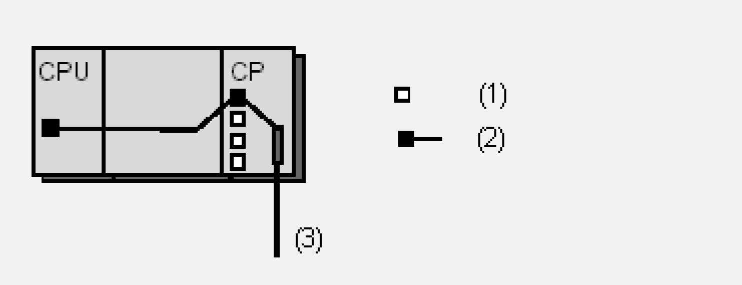
(3) Industrial Ethernet, PROFIBUS

Note

To execute programming device functions on CPs for the S7-400 through the MPI or through the integrated DP interface, two connection resources are needed at the CPU (for two transition points). This requirement must be taken into account in determining the total number of configured STEP 7 connections.

FMS Interface

Communication through the FMS interface takes place only through CPs. Each FMS connection takes up one resource at the CP for the end point. At the CPU, each CP requires a connection resource for communication to it.



(1) Available connection resource

(2) Assigned connection resource

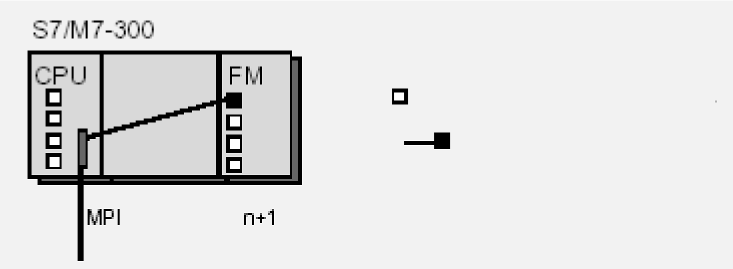
(3) PROFIBUS

Note

To execute programming device functions on CPs for the S7-400 through the MPI or through the integrated DP interface, two connection resources are needed at the CPU (for two transition points). This requirement must be taken into account in determining the total number of configured STEP 7 connections.

S7 Connections through S7/M7-300 and C7-600

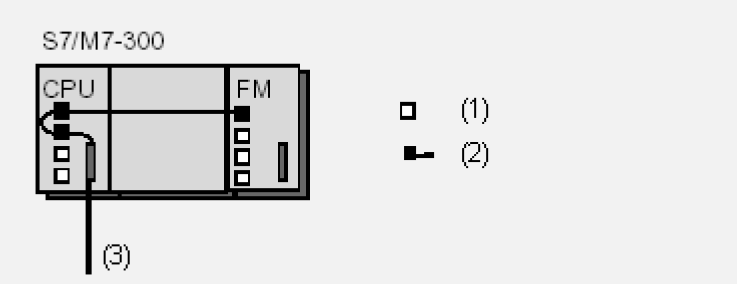
For S7 connections established via the MPI interface, only one connection resource for the end point is occupied at the FM for an S7/M7-300 (for CPU 312-316 only) and C7-600.



(1) Available connection resource

(2) Assigned connection resource

For S7 connections established via the internal MPI-/PROFIBUS DP interface, two connection resources (for two transition points) at the CPU are occupied; at the FM, one is required in each case (for the end point).



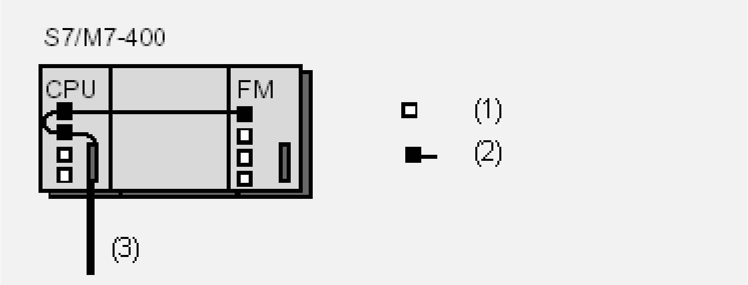
(1) Available connection resource

(2) Assigned connection resource

(3) PROFIBUS DP

S7 Connections through S7/M7-400

For S7 connections established through the internal MPI-/PROFIBUS DP interface, two connection resources (for two transition points) at the CPU are occupied; at the FM, one is required in each case (for the end point). This also applies for each additional CPU (multicomputing) within the same station since this node is on the MPI.



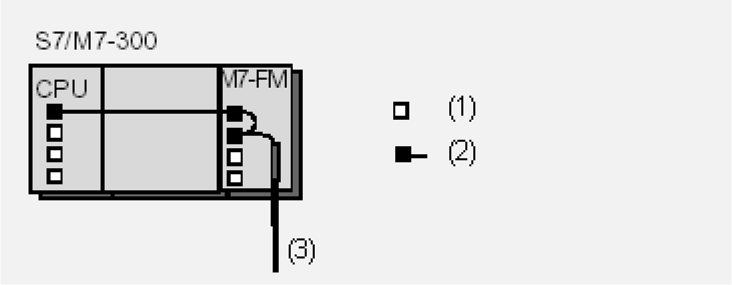
(1) Available connection resource

(2) Assigned connection resource

(3) MPI or MPI/PROFIBUS DP

S7 Connections through M7 FMs

For S7 connections established through the internal MPI-/PROFIBUS DP interface, each S7 connection at the FM requires two connection resources (for two transition points); at an S7/M7 CPU or C7-600 CPU, one is required in each case (for the end point).



(1) Available connection resource

(2) Assigned connection resource

(3) PROFIBUS DP

11.6 Using Connection Resources with Fault-Tolerant S7 Connections

For H-systems, there are a number of possible configurations that differ with respect to the number of H-CPUs, number of CPs and number of subnets. Depending on the configuration, two or four partial connections are possible for each fault-tolerant S7 connection. These connections ensure communication is maintained even if a component should fail.

The following sections describe the most common configurations and how they use connection resources for a fault-tolerant connection.

Basic Information

At each H-CPU, a connection resource is occupied for each of the endpoints of a fault-tolerant S7 connection (that is, for redundant connections on both H-CPUs concerned).

For each fault-tolerant S7 connection, STEP 7 establishes two partial connections as alternative communication paths. To ensure the availability of both paths, each path must have resources reserved for it. If both partial connections run through the same intermediate CP, then two connection resources are also reserved on this CP.

The following sections explain this situation in detail.

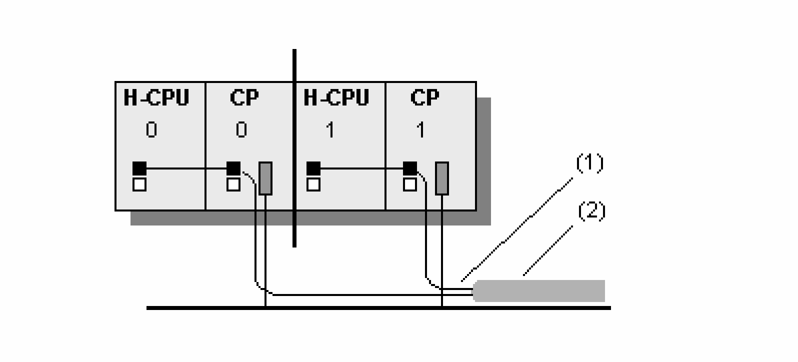
Notes on the Illustrations

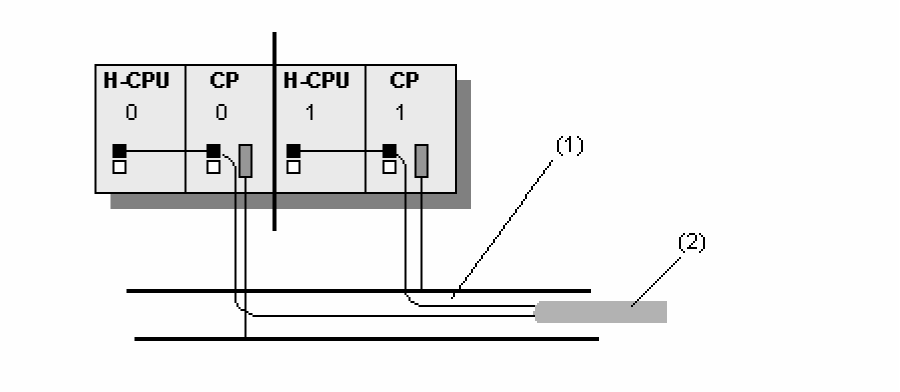
For redundant H-stations, the two CPUs are labeled "H-CPU 0" and "H-CPU 1".

The CPs are numbered sequentially (CP 0, CP1, ...).

Case 1: Configuration with redundant H-stations (local and remote), two partial connections possible

As shown in the following illustrations, each of the two CPUs has one resource assigned on it, and each of the CPs involved has one connection resource assigned on it as well.





(1) Partial connection

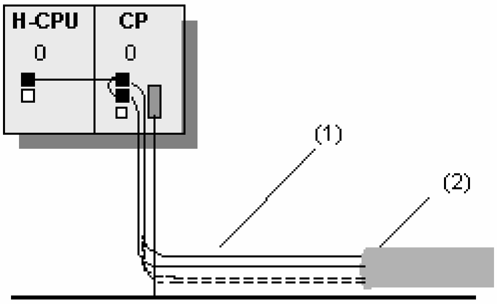
(2) Fault-tolerant S7 connection

Case 2: Configuration with non-redundant H-Station (local) and redundant H-Station (remote)

In this case, there is a difference whether the redundant partner station is fitted with a total of two or four CPs.

If the partner station is fitted with a total of two CPs, then one connection resource is reserved on the local H-CPU and two connection resources are reserved on the local CP.

If the partner station is fitted with a total of four CPs and the "Enable max. CP redundancy" option was selected, then a total of four connection paths is possible. Even so, one connection resource is reserved the local H-CPU, and two connection resources are reserved for the local CP. The reason for this is that a maximum of two partial connections can be active.



(1) Partial connection

(2) Fault-tolerant S7 connection

Case 3: Configuration with redundant H-stations (local and remote), four partial connections possible

If redundant H-stations are connected in a subnet, then a maximum of four partial connections is possible. One connection resource is reserved for each of the H-CPUs, and two connection resources are reserved for each of the CPs involved

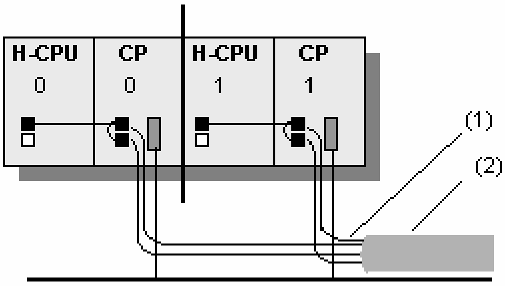
The following partial connections are possible:

From H-CPU0/CP0 (local station) to CP0 /H-CPU0 (partner station)

From H-CPU1/CP1 (local station) to CP1 /H-CPU1 (partner station)

From H-CPU0/CP0 (local station) to CP1 /H-CPU1 (partner station)

From H-CPU1/CP1 (local station) to CP0 /H-CPU0 (partner station)

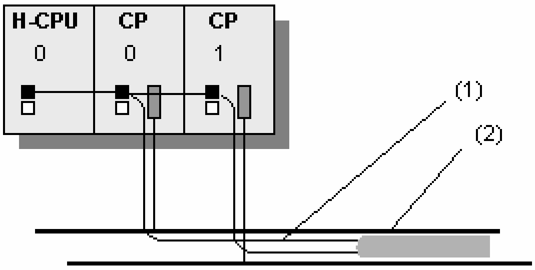


(1) Partial connection

(2) Fault-tolerant S7 connection

Case 4: Configuration with non-redundant H-Station and two CPs (local) and a redundant H-Station (remote)

If the local station is fitted with a total of two CPs, then one connection resource is reserved on the local H-CPU and one connection resource is reserved on each of the local CPs.



(1) Partial connection

(2) Fault-tolerant S7 connection

Call 5: Configuration with redundant H-Station (local) and redundant H-Station (remote) with maximum CP redundancy

If the local station and the partner station are fitted with a total of four CPs and the "Enable max. CP redundancy" option was selected, then a total of four connection paths is possible. STEP 7 reserves one connection resource for the local H-CPU and connection resource for each of the local CPs.

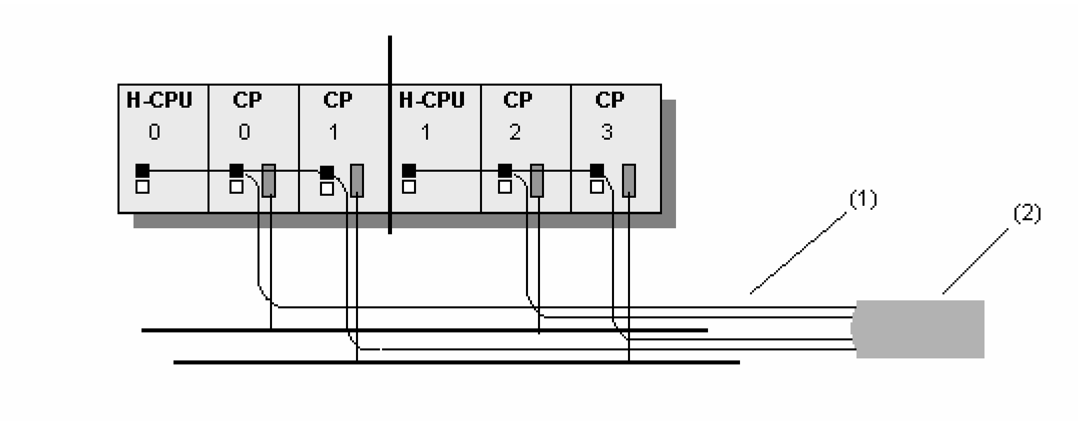
The following partial connections are possible:

From H-CPU0/CP0 (local station) to CP0 /H-CPU0 (partner station)

From H-CPU1/CP2 (local station) to CP2 /H-CPU1 (partner station)

From H-CPU0/CP1 (local station) to CP1 /H-CPU0 (partner station)

From H-CPU1/CP3 (local station) to CP3 /H-CPU1 (partner station)



(1) Partial connection

(2) Fault-tolerant S7 connection

11.7 Blocks for Different Connection Types

Blocks for Use with S7 Connections

The system function blocks are integrated in the CPUs of the S7-400.

For S7-300, with newer CPUs and CPs, you can operate S7 communication actively (that is, as client) by means of the interface of the CP. The blocks (FBs) have the same number and designation as the SFBs of the S7-400; however, they must be called up cyclically in the user program of the S7-300-CPU. You can find the blocks in the SIMATIC_NET_CP library.

The CP must support the client function for S7 communication.

The CPU 317-2 PN/DP with PROFINET interface can also be configured as a client for S7 communication. In this case, the same blocks are used as in the case for the S7-300 with CP mentioned above. These blocks are also in the Standard Library (Communication Blocks/CPU_300). The client function is only available with the PROFINET interface.

SFB/FB/FC	Designation	Brief Description
SFB8/FB8 SFB9/FB9	USEND URCV	Uncoordinated data exchange using a send and a receive SFB Max. length SFB 8/9: 440 bytes, split into 4x100 bytes. Max. length FB 8/9: 160 bytes
SFB12/FB12 SFB13/FB13	BSEND BRCV	Exchange blocks of data of variable length between a send SFB and a receive SFB Max. length SFB 12/13: 64 KB Max. length FB 12/13: 32 KB
SFB14/FB14	GET	Read data from a remote device Max. length SFB 14: 400 bytes, split into 4x100 bytes Max. length FB 14: 160
SFB15/FB15	PUT	Write data to a remote device Max. length SFB 15: 400 bytes, split into 4x100 bytes Max. length FB 15: 160
SFB19	START	Execute a restart (warm restart) on a remote device
SFB20	STOP	Switch a remote device to STOP mode
SFB21	RESUME	Execute a hot restart in a remote device
SFB22	STATUS	Specific query of the status of a remote device
SFB23	USTATUS	Receive status messages from remote devices
SFC62	CONTROL	Query the status of the connection that belongs to an SFB instance
FC62	C_CNTRL	Query the status of a connection (for S7- 300 CPUs)

Blocks for Use with Point-to-Point Connections

For the PtP connection type you can use the SFBs BSEND, BRCV, GET, PUT, and STATUS (see above table).

You can also use the SFB PRINT:

SFB	Designation	Brief Description
SFB16	PRINT	Send data to a printer

Blocks for Use with FMS Connections

FB	Designation	Brief Description
FB 2	IDENTIFY	Identify the remote device for the user
FB 3	READ	Read a variable from a remote device
FB 4	REPORT	Report a variable to the remote device
FB 5	STATUS	Provide the status of a remote device on request from the user
FB 6	WRITE	Write variables to a remote device

Blocks for Use with FDL, ISO-on-TCP, UDP and ISO Transport Connections as well as Email Connections

FC	Designation	Brief Description
FC 5	AG_SEND	Send data by means of a configured connection to the communication partner (<= 240 bytes).
FC 6	AG_RECV	Receive data by means of a configured connection from the communication partner (<= 240 bytes, not email).
FC 50	AG_LSEND	Send data by means of a configured connection to the communication partner.
FC 60	AG_LRECV	Receive data by means of a configured connection from the communication partner (not email).
FC 7	AG_LOCK	Lock the external data access by means of FETCH/WRITE (not for UDP, email).
FC 8	AG_UNLOCK	Unlock the external data access by means of FETCH/WRITE (not for UDP, email).

11.8 Working with the Connection Table

Displaying and Hiding Columns of the Connection Table

1. Place the pointer in the connection table and click with the right mouse button to display a pop-up menu.
2. In the pop-up menu, select **Show/Hide Columns** >…. In the next pop-up menu, select the name of the column that you want to show or hide.

   The names of the visible columns are indicated with a check mark. If you select a visible column, the check mark disappears and the column is hidden.

Optimizing the Column Width

In order to adapt the width of a column to its contents so that all the text in the rows is legible, proceed as follows:

1. Place the pointer **in the header** of the connection table to the right of the column that you want to optimize until the pointer assumes the shape of two parallel lines (as though you wanted to change the width of the column by dragging with the pointer).
2. Double-click this position.

Tip: For columns that are set too narrowly, you can display the complete contents if you allow the pointer to rest for a short time on the filed in question.

Sorting the Connection Table

In order to sort the connection table in ascending order according to a column, click the heading of the column.

An additional click on the heading sorts the connection table in descending order.

Note

The column width and which columns are visible are saved for each project when you exit the project. This means that when you open the project on another computer, the settings are also valid on that computer.

Further Information

For further information on the columns of the connection table please refer to the context help (for example for the menu command **View > Display/Hide Columns** ...).

Navigating with Cursor Keys and Calling Up Dialogs for Editing

You can use the UP ARROW and the DOWN ARROW keys to select a connection in the connection table. The selected connection is highlighted.

If you use the RIGHT ARROW key or the LEFT ARROW key to navigate to the field in the "Partner" column and you press ENTER, the Change Connection Partner dialog appears. If you navigate to a different field (such as Local ID) and you press ENTER, the "Properties - Connection dialog appears.

If you have selected multiple connections, that is, if you have highlighted several columns (by holding down the CTRL key and selecting one column after another), and by so doing have called up the accompanying dialog (Change Connection Partner or Properties of the Connection), the dialogs of the highlighted connections appear one after the other.

Changing the Properties of the Connection

If you want to change a connection that has already been configured, for example, to set a different connection path (interface), proceed as follows:

1. Select the connection that you want to change.
2. Select the menu command **Edit > Object Properties**.
   In the dialog box that appears, you can change the properties that can be edited.

Changing local data only: As of STEP 7 V5.1, Service Pack 1, you can change the local ID directly in the Local ID column of the connection table.

Go To Partner Station

As of STEP 7 V5.2 you can easily move to the connection table of a connection partner when you edit the connection table:

1. Select a connection from the connection table.
2. Select menu command Edit > Go To Partner Connection.

This function is also available for interproject connections in the multiproject. The project that contains the connection partner must be open.

11.9 Inconsistent Connections

If a connection is inconsistent, the structure of the connection data is disrupted, or the connection would not be capable of functioning in the context of the project.

Inconsistent connections cannot be compiled and loaded - no operation is possible with such a connection.

Inconsistent connections are identified in the connection table by the color **red** and ***italics***.

Possible Causes of Inconsistent Connections

- Deleting or altering the hardware configuration
- Missing networking of interfaces in the project that are necessary for a connection
- Exceeding connection resources
- Error in saving data due to insufficient memory
- Connections to an unspecified partner without specifying the partner address
- Connections to a "partner in an unknown project ", whereby the connections have yet to be configured

Detailed information about the causes of inconsistent connections is listed in the Outputs on Consistency Check window (after the consistency check has been run, menu command **Network > Check Consistency** or **Network > Check Cross-Project Consistency**).

Detailed information on the cause of the inconsistency can be obtained by editing the connection properties (highlight the connection and select the menu command **Edit > Object Properties**).

Remedies

For many causes, it is enough to edit the connection properties to reestablish consistency, that is, to correct the connection data. The inconsistent connection can be corrected when the new properties are assumed.

If the connection cannot be repaired by opening the connection properties, changing or reversing the configuration, it may be necessary to delete and recreate the connection.

11.10 Displaying the Connection Status

Connection Status

In order to start up a system or to diagnose errors, you can display the status of the communication connections for a module in the connection table.

Possible Status of a Connection:

- Established
- Not established
- Is being established now
- Not available

Note

The "Connection Status" column has a **yellow** background if the connection is only available online, meaning that it is not available offline in the project. The cause can be that this connection was downloaded to the module without first having been saved in the project.

Requirements

- Displaying the connection status is possible only for the local endpoint of a connection, for example, for a CPU that is selected in the network view.
- The module must support the connection status.

  (Possible as of 10/99, this performance feature is documented in the instruction list of the CPU. The CPU must support the SSL ID 0x36 "Connection-Specific Diagnostics.")
- There is an online connection to the connection end point.
- If the connections that are involved are connections that were configured with an optional package, you will need the optional package in order to display the status.
- There is a project for the network configuration on the programming device, or you have uploaded the station to the programming device (menu command **PLC > Upload**).

Procedure

1. Select the module for which you want to display the connection status.
2. Select the menu command PLC > Activate Connection Status.

   The window becomes an ONLINE window and the Connection Status column is placed in front of the connection table.

   The status information for each connection is displayed in this column.

   The text of the executed menu command changes to "Deactivate Connection Status." You use this menu command to return to the (offline) connection table so that you can configure connections.
3. To get more detailed information concerning the current connection status, double-click the appropriate row of the connection table or select the menu command **Edit > Object Properties** (alternatively, click with the right mouse button to get a pop-up menu and select Object Properties or Status Information).

   The Status Information tab appears with additional information on the connection status.

Exiting Connection Status

Use the menu command **PLC > Deactivate Connection Status** to exit the function and return to the (offline) connection table.

11.11 Access to Connection IDs during Programming

When programming (for example, in the Statement List Editor), you can perform the following functions while being prompted by dialogs:

- Access already configured connections
- Create new connections
- Change the connection partner

To call up the dialog, proceed as follows:

1. Program the block call (for example, SFB12, BSEND)
2. Position the mouse pointer on a block parameter (for example, "ID")
3. Click the right mouse button to get a pop-up menu. In the pop-up menu, select "Connections."
   A dialog with all the connections that apply in this context appears (depending on the associated stations, already configured connections)

11.12 Connection Types for Connection Partners in the Same Project

11.12.1 Connection Types for Connection Partners in the Same Project

Selecting the Connection Type for Connection Partners in the Same Project

The connection type is dependent on the subnet and the transfer protocol via which the connection is established, and on the automation family to which the connection partners belong.

The blocks (SFBs, FBs, or FCs) you can use depend on the connection type.

The following table should make it easier for you to select the connection type for the connection you want to establish.

Connection Type	Subnet Type	Connection between SIMATIC...	SFB/FB/FC
S7 connection	MPI, PROFIBUS, Industrial Ethernet	S7 - S7, S7 - PG/PC, S7 - PG/PC with WinCC with MPI also: M7 - M7, M7 - S7, M7 - PG/PC S7 - partner in another project (S7, PG/PC with WinCC)	**SFBs** USEND, URCV, BSEND, BRCV, GET, PUT, START, STOP, RESUME, STATUS, USTATUS
S7 connection, fault-tolerant	PROFIBUS, Industrial Ethernet	S7(H) - S7(H), S7(H) - PC station (H)	**SFBs** USEND, URCV, BSEND, BRCV, START, STOP, RESUME, STATUS, USTATUS
PTP Connection	Point-to-point (computer protocol RK 512/3964(R))	S7 - S7, S7 - S5, S7 - non-Siemens device S7 - partner in another project (S7, no Siemens device)	**SFBs** BSEND, BRCV, GET, PUT, STATUS, PRINT
FMS connection	PROFIBUS (FMS protocol)	S7 - S7, S7 - S5, S7 - PG/PC, S7 - no Siemens device, S7 - broadcast to all nodes S7 - partner in another project (S7, S5, PG/PC, no Siemens device)	**FBs** READ, WRITE, IDENTIFY, OSTATUS, REPORT
FDL connection	PROFIBUS (FDL protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 -non-Siemens device S7 - partner in another project (S7, S5, PG/PC, non-Siemens device)	**FCs** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV
ISO transport connection	Industrial Ethernet (ISO Transport protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - third-party device, S7 - unspecified S7 - partner in other project (S7, S5, PG/PC, third-party device, unspecified)	**FCs** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV AG_LOCK AG_UNLOCK SSEND***, SRECV***

Connection Type	Subnet Type	Connection between SIMATIC...	SFB/FB/FC
ISO-on-TCP connection	Industrial Ethernet (TCP/IP protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - third-party device, S7 - unspecified S7 - partner in other project (S7, S5, PG/PC, third-party device, unspecified)	FCs AG_SEND, AG_RECV, AG_LSEND, AG_LRECV, AG_LOCK, AG_UNLOCK, SSEND***, SRECV***
TCP connection	Industrial Ethernet (TCP/IP protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - third-party device, S7 - unspecified S7 - partner in other project (S7, S5, PG/PC, third-party device, unspecified	**FCs** AG_SEND*, AG_RECV*, AG_LSEND**, AG_LRECV**, AG_LOCK, AG_UNLOCK, SSEND***, SRECV***
UDP connection	Industrial Ethernet (TCP/IP protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - third-party device, S7 - unspecified S7 - partner in other project (S7, S5, PG/PC, third-party device, unspecified)	**FCs** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV, SSEND***, SRECV***
E-mail connection	Industrial Ethernet (TCP/IP protocol)	S7 - unspecified (S7 - mail server)	**FCs** AG_SEND, AG_LSEND

***** The FCs AG_SEND and AG_RECV here can only be used for S7-300 (this depends on release number of the CP, see the documentation for the CP).

****** The FCs AG_LSEND and AG_LRECV here can generally be used for S7-400, and in addition for S7-300 (this depends on release number of the CP, see the documentation for the CP).

******* FCs SSEND and SRECV can be used for S7-400 as of software version V5.1 and CP 443-1EX20/GX20.

Special Feature: Connection to Broadcast and Multicast Nodes

For special connection types there is the option of selecting not just one connection partner, but several ones (broadcast and multicast nodes). These options are described in the SIMATIC NET (NCM S7) manuals. The connection partners "all broadcast nodes" or "all multicast nodes" are listed in the dialog box for entering a new connection.

- You can set up a connection to "all broadcast nodes" (simultaneous sending to all broadcast receivers) for the connection types FDL connection and UDP connection.
- You can set up a connection to "all multicast nodes" (simultaneous sending to several nodes) for the connection types FDL connection and UDP connection.

11.12.2 Rules for Creating Connections

Selecting a Connection Path with More Than One Subnet in the Project

If stations are connected to a number of subnets, STEP 7 chooses a connection path via one of the subnets. This connection path was found by STEP 7 to be more efficient than the others. STEP 7 selects a preferred subnet in the following order: Industrial Ethernet before Industrial Ethernet/TCP/IP before MPI before PROFIBUS.

Example: Two stations are networked together via MPI and Industrial Ethernet. STEP 7 chooses the route via Industrial Ethernet.

Note that the route selected by STEP 7 remains even if the connection to the subnet is lost. STEP 7 does not select an alternative route via another subnet (exception: redundant S7 connections).

In an S7 connection, the connection path set automatically by STEP 7 can be changed by the user in the properties dialog box for the connection, for example, from MPI to PROFIBUS.

Number of Possible Connections

The number of possible connections which can be entered in the connection table depends on the resources of the selected module and is monitored by STEP 7.

Information on the connection resources of a module is available in the "Module state" dialog box, "Communication" tab card.

11.12.2.1 Configuring Connections for Modules in a SIMATIC Station

The following section shows you how to create connections for a connection end point (for example, a CPU) in the network view.

Special Feature

STEP 7 automatically assigns a local ID for each end point of the connection under either of the following conditions:

- If both communication partners are S7-400 stations
- If one communication partner is an S7-400 and the other is a SIMATIC station

In this case, you configure only the connection in the connection table of a partner that has other communication partners, then automatically the matching entry in its connection table.

11.12.2.2 Inserting a New Connection

A connection defines the communication relationship between two nodes. It defines the following:

- The two communication nodes involved
- The connection type (for example, S7, point-to-point, FMS, ISO-on-TCP, FDL, or ISO Transport)
- Special properties which depend on the type of connection (such as whether a connection remains permanently configured, or whether it is set up/broken dynamically in the user program)

Requirement

You must be in the network view of NetPro.

Procedure

1. In the network view, select the module for which you want to create a connection.
   Result: The connection table for the selected module is displayed in the lower half of the network view.
2. Double-click an empty row in the connection table, or select a row and select the menu command **Insert > Connection**.
3. In the "New Connection" dialog box select the required connection partner. You can obtain help with selecting a connection partner in the online help for this dialog box.
4. Specify the type of connection.
5. Activate the "Show Properties dialog box" check box if you want to view or change the properties for the connection after clicking "OK" or "Add:"
   The contents of the "Properties" dialog box is dependent on the connection selected; you can obtain help with filling it out in the online help for the dialog box.
   Result: STEP 7 enters the connection in the connection table of the local (selected) node and assigns the local ID and a partner ID if required for this connection which you will need when programming the communication function blocks (value for the block parameter "ID").

11.12.2.3 Changing the Connection Partner

You can change the connection partner for a previously configured connection. The local ID and the connection type are retained.

Requirement

You must be in the network view of NetPro.

Procedure

1. In the network view, select the module for which you want to modify a connection.
2. Select the row in the connection table containing the connection you want to modify.
3. Double-click the selected area of the "Partner" column or select the menu command **Edit > Connection Partner.**
   Result: The "Modify Connection" dialog box appears.
4. In the "Connection Partner" box, select the end point (i.e. a module in the current project or in another project in the multiproject). If the new connection partner that you have selected is a partner "in an unknown project", you will then be prompted to enter a connection name (reference) for this connection.
5. Confirm your entries by clicking "OK."

Notes:

Note that the configured properties of the connection will be returned to the default setting when the connection partner is modified. Use the menu command **Edit > Object Properties** if you wish to change the properties of the connection.

The following applies to S7 connections: You can change an "unspecified" connection partner from STEP 7 V5 onwards (for example, to a SIMATIC 300/400 station). You can also change the SIMATIC station partner to "unspecified."

11.12.2.4 Reserving a Connection

If you wish to reserve the communication resources of a node to expand your project at a later date, or if you do not wish to specify a connection partner yet, you can enter "Unspecified" for the connection partner. Reserving connections is currently not possible for all connection types.

Requirement

You must be in the network view of NetPro. The properties dialog box for the connection is open.

Procedure

1. In the "Station" box, select the option "Unspecified."
   Result: The entry in the "Module" box is deactivated.
2. Select the connection type in the "Type" box.
3. Activate the "Show Properties dialog box" check box if you want to view or change the properties for the connection after clicking "OK" or "Add."
4. Confirm your entries by clicking "OK."
   Result: STEP 7 enters the connection in the connection table of the local node and assigns the local ID for this connection which you will require for programming the communication blocks.

Note

Note that you can also assign special properties for each connection. To do this, select the menu command **Edit > Object Properties**.

11.12.2.5 Deleting One or More Connections

Requirement:

You must be in the network view of NetPro.

Procedure

1. Select the connections you wish to delete.
2. Select the menu command **Edit > Delete.**
   If local ID and partner ID for the connection were entered in the connection table, STEP 7 also deletes the connection in the table for the connection partner.
3. Download the connection tables with the deleted connections to the relevant programmable module. (To delete all the connections of a programmable module, you must download an empty connection table to the module.)

11.12.2.6 Copying Connections

Introduction

Connections are not copied individually but always in context with the project or the station.

You can copy the following:

- Entire projects
- One or more stations within a project or beyond the boundaries of a project

Requirement

The SIMATIC Manager must be open.

Copying a Project

When you copy a project, all configured connections are also copied. There are no settings required for the copied connections because the connections remain consistent.

Copying a Station

When you copy one or more stations within the project boundaries or beyond them, you may have to reassign the connection partners to the local node (modifying a connection).

If no connection partner exists for a connection, you will see this in the connection table, since the row of the connection partner will be displayed in bold.

Connections between stations that are copied beyond the project boundaries remain in tact and consistent if the relevant subnets between these stations are copied along with them.

11.12.3 Configuring Connections for a SIMATIC PC Station

You can configure connections for SIMATIC PC stations. Depending on which interfaces are available on your PC (communication processors, CPs), you can configure the corresponding connection types.

If you want to create or modify connection types other than S7 connections (such as fault-tolerant S7 connections), you must make sure you have installed the corresponding optional package.

Requirements

You have configured a SIMATIC PC station with all the communication end points (applications, OPC servers, software PLCs or slot CPUs) and PC communication cards. You have also configured all the stations that are supposed to be the communication end points of the connections.

To configure connections, you must note the following, depending on the components that you are using:

- WinLC (V3.0): This version does not support any configured connections.
- Slot PLCs (CPU 41x-2 PCI): You can configure connections for these CPUs only by means of exactly one of the configured CPs.
- Applications and OPC servers: Please see the documentation for the applications or programming interfaces that you are using.

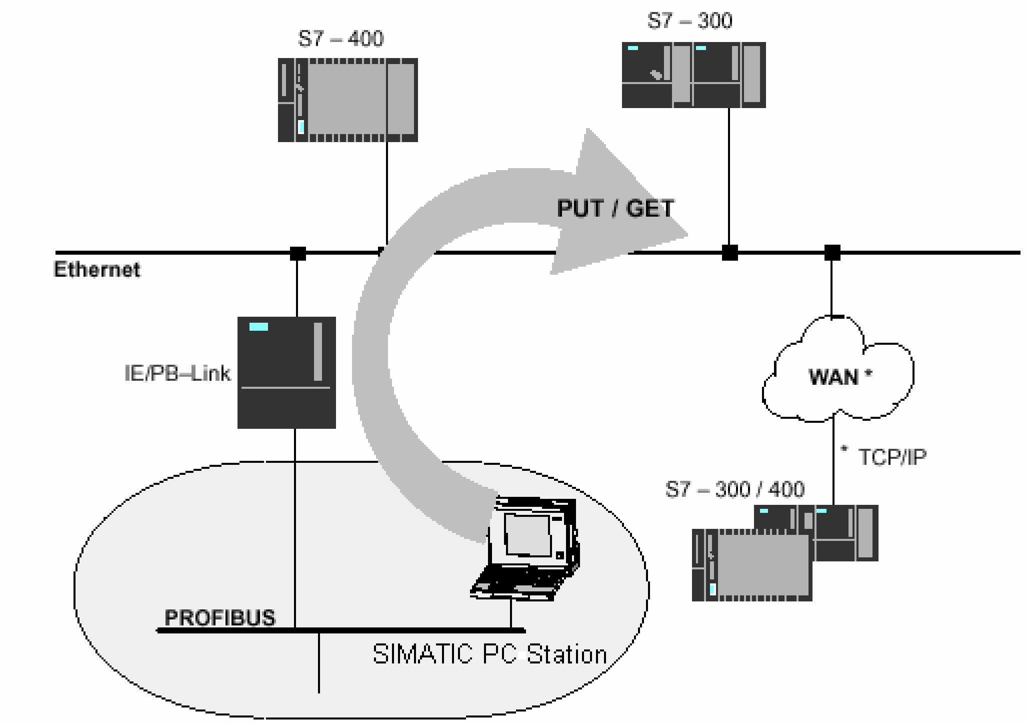
How to Configure a Connection

1. Select the communication end point in the SIMATIC PC station in the network view, so that the connection table is visible.
2. Double-click an empty row in the connection table or highlight a row and select the menu command **Insert > Connection**.
3. Select the required connection partner in the "New Connection" dialog box. You can find help on selecting the connection partner in the online help for this dialog box.
4. Define the type of connection.
   Special property of the connection: In contrast to S7 stations, STEP 7 assigns a name instead of a numerical value for the local ID (connection ID). You can change this name in the object properties of the connection.
   New connections are always two-way connections; that is, STEP 7 automatically enters a connection to the local station in the connection table of the partner.
5. Select the menu command Network > Save and Compile.

   On compilation, a configuration file ("XDB file") is generated for the PC station. This contains the name of the PC station, the connection descriptions, and the assigned parameters and subnet information for the PC communication cards.
   You enter the path for the configuration file in the "Configuration" tab (object properties of the SIMATIC PC Station).
6. If the PC stations are configured with the Station Configuration Editor, you can load the PC station (see Downloading a PC Station).
   Otherwise, continue with step 7.
7. Copy the configuration file to the PC station (programmable logic controller).
   The location of the configuration file is to be set on the PC station by means of the "Set PG/PC Interface" program ("STEP 7 Configuration" tab).
   You can find additional information concerning installation on a programming device/PC in the description "SIMATIC NET, S7 Programming Interface" and in the online Help for the "STEP 7 Configuration" tab.

11.12.3.1 Configuring Routed S7 Connections for a SIMATIC PC Station

For OPC servers (part of the CD SIMATIC NET PC software products as of Version 6.1) you can configure S7 connections between a SIMATIC PC station/SIMATIC HMI station and an S7 station that is connected to a different subnet.



Requirements

Both of the subnets must be interconnected across a router, for example IE/PB Link. Routing functions can also be performed by an S7 station or a SIMATIC PC station that is interconnected to both subnets via CPs or CPU.

The end point in the SIMATIC PC/HMI station (currently only an OPC server) must support routed subnets.

The connection partner must be located in the same project.

What to do if only one-way S7 connections can be configured

The S7 station can only be operated as communication server for configured one-way S7 connections on interconnected subnets.

In this case, use NetPro to configure a one-way S7 connection for a SIMATIC PC station/HMI station at the other subnet. This enables you to access data of the S7 station in the user program of the SIMATIC PC station, via the functions PUT (write access) and GET (read access).

11.12.4 Programming Devices / PCs as Connection Partners

If you want to configure connections for an end point on a programming device or PC, you have various options. Select either the "PG/PC" object or the "SIMATIC PC Station" object in the network view:

- "SIMATIC PC Station" for applications that use configuration files (*.xdb files) for communication to a SIMATIC S7 station, for example:
  - For redundant connections with the optional package S7-REDCONNECT (fail-safe S7 communication) or
  - For SAPI-S7 interface
  - For OPC server

  Please check in the product information of the SIMATIC NET product to see if configuration by means of a configuration file is possible.
- "PG/PC" for applications that use LDB files for communication to a SIMATIC S7 station (SAPI-S7 interface)

11.12.4.1 S7 Connection to a PG/PC

An S7 connection from an S7 station to a PG/PC is only possible if the PG/PC has an SAPI-S7 programming interface (C programming interface for access to SIMATIC S7 components) or an OPC Server.

You will find more information on filling out the dialog box in the online help for this dialog.

For the Connection Partner "PG/PC"

One method of configuring connections is by creating an LDB (local database):

1. Configure the PG/PC in the network view.
2. Configure the station from which a connection should go to the PG/PC.
3. When you create the S7 connection you must select "PG/PC" as the connection partner.
4. Edit the "Address Details" dialog box (accessible via the properties of the connection). In this dialog box you enter the name of the connection and the virtual field device name of the programming device/PC. You will find more information in the online help for this dialog box.
5. Then double-click the configured programming device/PC and generate the local database (LDB).
6. Transfer the database to the programming device/PC.
7. Download the connection(s) to the station.

For the Connection Partner "SIMATIC PC Station"

For the programming device/PC you create a SIMATIC PC station in the network view. The "Application" forms the end point of a connection with this station type.

11.12.4.2 S7 Connection to a PG/PC with WinCC

For the Connection Partner "Unspecified"

You can create S7 connections to PGs/PCs with *WinCC* within a project or beyond project boundaries. When you create the S7 connection you must select "Unspecified" as the connection. You enter special address information for *WinCC* in the "Address Details" dialog box.

For the Connection Partner "SIMATIC PC Station"

For the programming device/PC you create a SIMATIC PC station in the network view. WinCC forms the end point of a connection with this station type.

11.13 Configuring Connections to Partners in Other Projects

Selecting the Connection Type to Communication Partners in Other Projects

When you have set up a multiproject, you can then configure interproject connections.

The connection type is dependent on the subnet and the transfer protocol through which the connection is established and on the automation family to which the connection partners belong.

The blocks (SFBs, FBs, or FCs) you can use depend on the connection type.

The following table should make it easier for you to select the connection type for the connection you want to establish.

Connection Type	Subnet Type	Connection Between SIMATIC ...	SFB/FB/FC
S7 connection	MPI, PROFIBUS, Industrial Ethernet	S7 - S7, S7 - PG/PC, S7 - PG/PC with WinCC with MPI also: M7 - M7, M7 - S7, M7 - PG/PC	**SFBs** USEND, URCV, BSEND, BRCV, GET, PUT, START, STOP, RESUME, STATUS, USTATUS
S7 connection, fault-tolerant	PROFIBUS, Industrial Ethernet	S7(H) - S7(H), S7(H)- PC-Station (H)	**SFBs** USEND, URCV, BSEND, BRCV, START, STOP, RESUME, STATUS, USTATUS
FDL connection	PROFIBUS	S7 - S7, S7 - S5, S7 - PC/PG, S7 – non-Siemens device, S7 - broadcast to all nodes	**FBs** READ, WRITE, IDENTIFY, OSTATUS, REPORT
ISO Transport connection	Industrial Ethernet (ISO-transport protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 – non-Siemens device, S7 – unspecified	**FCs** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV
ISO-on-TCP connection	Industrial Ethernet (TCP/IP protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 – non-Siemens device S7 – unspecified S7 - partner in another project (S7, S5, PG/PC, non-Siemens device, unspecified)	**FCs** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV AG_LOCK AG_UNLOCK
TCP connection	Industrial Ethernet (TCP/IP protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - non-Siemens device, S7 - unspecified	**FCs** AG_SEND*, AG_RECV*, AG_LSEND**, AG_LRECV** AG_LOCK AG_UNLOCK

Connection Type	Subnet Type	Connection Between SIMATIC ...	SFB/FB/FC
UDP connection	Industrial Ethernet (TCP/IP protocol)	S7 - S7, S7 - S5, S7 - PC/PG, S7 – non-Siemens device, S7 – unspecified	**FCs** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV

***** The FCs AG_SEND and AG_RECV here can only be used for S7-300 (this depends on release number of the CP, see the documentation for the CP).

****** The FCs AG_LSEND and AG_LRECV here can generally be used for S7-400, and in addition for S7-300 (this depends on release number of the CP, see the documentation for the CP).

Special Feature: Connection to Broadcast and Multicast Nodes

For special connection types there is the option of selecting not just one connection partner, but several ones (broadcast and multicast nodes). These options are described in the SIMATIC NET (NCM S7) manuals. The connection partners "all broadcast nodes" or "all multicast nodes" are listed in the dialog box for entering a new connection.

- You can set up a connection to "all broadcast nodes" (simultaneous sending to all broadcast receivers) for the connection types FDL connection and UDP connection.
- You can set up a connection to "all multicast nodes" (simultaneous sending to several nodes) for the connection types FDL connection and UDP connection.

11.13.1 Basic Procedure for Configuring Connections between Projects

Configuring Connections with Multiproject

AS of STEP 7 V5.2 you can easily create interproject connections with the help of the multiproject (see also the section: What you Should Know About Multiprojects).

Possible Connection Partners in the Other Project (without Multiproject)

There are two methods of setting up connections to connection partners in other STEP 7 projects (without multiproject):

- Set up a connection to an "other station," a "PG/PC," or a "SIMATIC S5 station"
- Set up a connection to an "unspecified" connection partner

Note:

Connections to "other stations," "SIMATIC S5 stations," "PG/PC," and "unspecified" connection partners are also possible within a STEP 7 project. To which of these connection partners connections can be set up depends, among other things, on the connection type.

Differences Between the Two Methods

- You must configure an "other station", a "PG/PC", or a "SIMATIC S5 station" as subnet nodes in the current STEP 7 project. **Restriction:** For "other stations" and SIMATIC S5 stations you cannot configure S7 connections. All other connection types are possible.
- No subnet node needs to be configured in the current STEP 7 project for an unspecified partner. You can set up S7 connections, PTP connections, ISO Transport connections, and ISO-on-TCP connections to unspecified connection partners.

Special Case for PtP Connections

In contrast to the case for STEP 7 connections, the configuration of PtP connections to an unspecified partner does not require networking of the local node. You only have to network the communication partners in an actual plant before you place the connection in operation.

11.13.2 Creating a New Connection to an Unspecified Partner

FMS, FDL, ISO-Transport, and ISO-on-TCP Connections

FMS, FDL, ISO-Transport, and ISO-on-TCP connections are described in the SIMATIC NET, and SIMATIC NCM manuals.

S7 Connections

You can use unspecified connection partners in order to configure S7 connections to a connection partner in a different project.

Requirement

You must be in the connection table view (NetPro).

Procedure

You create an S7 or point-to-point connection to an "unspecified" connection partner as follows:

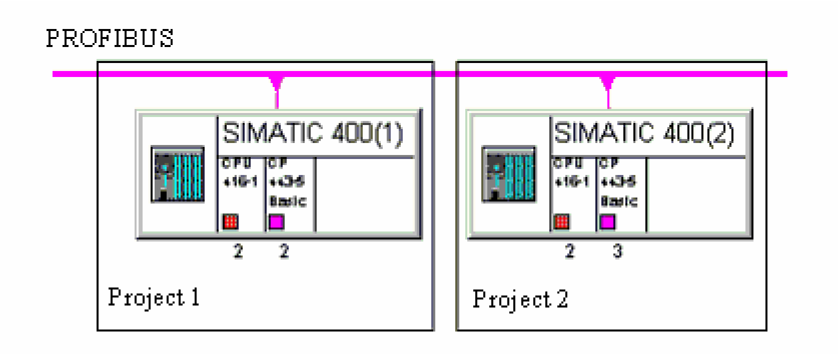
1. Select a module from which you want to run the connection (local node).
2. Double-click an empty row in the connection table, or select a row and select the menu command **Insert > Connection**.
3. In the "New Connection" dialog box select "Unspecified" as the connection partner.
4. Set the properties for the connection:
   For PtP connections: in the properties dialog box for the PtP connection change the name of the partner from "unspecified" to a suitable name (the name is also entered in the connection table).
   For S7 connections: enter the address of the interface of the connection partner and click the Address Details button in the properties dialog box.
   Depending on the connection partner, different settings are necessary in the "Address Details" dialog box. You will find more information on filling out the dialog box in the online help.

"Address Details" Dialog Box for S7 Connections to a Partner in a different Project

In the dialog boxes for the connection properties (Properties S7 Connection dialog box and Address Details dialog box) of an unspecified partner, you have to enter or select the address (of the interface), the end point of the connection (rack/slot), and the connection resources of each "Other Side" (that is, of the connection partner).

Example: The following configuration is a prerequisite:

The stations are in different projects. The CPU as connection end point is in the central controller (rack 0) slot 4 in both projects. Both stations are networked by means of the PROFIBUS interface of a CP.



The following table shows which supplementary information is required for the configuration shown to enable a connection setup in operation.

Information in Dialog Boxes on the Unspecified Connection Partner...	**... for Project 1**	**...for Project 2**	**Comments**
Active connection setup	Yes	No	Dialog box: Properties S7 Connection
Address (interface) Local/Partner	Local: 2 Partner: 3	Local: 3 Partner: 2	Dialog box: Properties S7 Connection, under Connection Path; the addresses have to match "crosswise".
Rack/slot	Local: 0/4 Partner: 0/4	Local: 0/4 Partner: 0/4	Dialog box: Address Details"
Connection resource (hex) (Local/Partner)	Local: 10 Partner: 11	Local: 11 Partner: 10	Dialog box: Address Details; connection resources cannot be assigned twice and must match "crosswise".
TSAP (Transport Service Access Point)	Is displayed (cannot be changed)	Is displayed (cannot be changed)	Dialog box: Address Details; for station-internal addressing of the interface, by means of the connection "Running" (is made up of the address details)

11.13.3 Creating a Connection to an 'Other Station', a 'PG/PC', or a 'SIMATIC S5 Station'

Requirement

You must have created the entire network configuration in both projects.

Stations that were configured in one project must be entered in the other project as an "other station."

The network view (NetPro) must be open.

Procedure

The procedure is the same as for creating connections to a partner (PG/PC, "other station," and SIMATIC S5) within one project.

11.14 Saving Connections

NetPro saves connections (menu command **Network > Save**) implicitly with all network and station data that are relevant for a functioning network configuration. For further information please refer to the section "Saving and Downloading the Network Configuration and Consistency Check".

12 Configuring Global Data Communication

12.1 Overview: Global Data Communication

Introduction

Global data communication (GD communication) is a simple method of communication integrated in the operating system of the S7-300 and S7-400 CPUs.

GD communication allows the cyclic exchange of data between CPUs via the multipoint interface. Cyclic data exchange takes place with the normal process image.

Global data communication is configured with STEP 7; transferring global data is done by the system and is not therefore programmed.

The following sections show how you can estimate using the technical specifications as given for every CPU (number of GD circles, size and number of GD packets etc.) what quantities of data can be exchanged between CPUs using the "GD communication" process.

The following are also listed:

- Send and receive conditions to be observed
- The formula for estimating the approximate response time

What Are Global Data?

Global data as they are used in the GD communication process are the following address areas in the CPU:

- Inputs, outputs (from the process image)
- Bit memory
- Data block areas
- Timers, counters (not recommended because the values are no longer current at the receiver; can only be configured as send address areas)

Peripheral areas (PI and PQ) and local data cannot be used for global data communication.

Data Transfer Process

Global data communication functions according to the broadcast method, meaning the receipt of the global data is not acknowledged. The sender does not receive information as to whether a receiver and which receiver has received the sent global data. If the process requires secure data exchange, use another service such as S7 functions.

Subnets for Global Data Communication

GD communication is possible:

- Via an MPI subnet only (between different stations) or
- Via the backplane bus only (for example, between S7 CPUs in a rack in multicomputing)

How Does an Address Area Become a Send/Receive Area?

The address areas that are involved in global data communication are configured in a global data table (GD table) using STEP 7:

- Every column is assigned to exactly one CPU which means the columns represent the CPUs involved in the data exchange (**maximum 15 CPUs**)
- Every row (or more exactly: every cell that can be edited in a row) represents the address areas via which one CPU only sends and one or more CPUs receive

When you have filled out the table, compiled it, and downloaded it to the CPUs involved, these CPUs send and receive cyclically via these address areas at the scan cycle checkpoint (meaning at the time at which the process image update takes place).

Special Feature: With the S7-400, configured global data can also be sent using SFC60 (GD_SND) for event-driven sending and received using SFC61 (GD_RCV).

12.2 Determining the Communication Capabilities from the GD Resources

You can determine how high the performance of an S7 CPU is with regard to GD communication using the following technical specifications ("GD resources"):

- The number of GD circles (in which the CPU can participate)
- The maximum number of net data per GD packet
- The maximum number of receive GD packets per GD circle
- The length of the consistent data per packet

The remaining documented GD resources are identical for all S7 CPUs and are not therefore relevant for selecting a CPU.

The technical data listed above provide indirect information about how much data the CPUs that are connected together via an MPI subnet or S7-400 backplane bus can exchange cyclically. How send data are "bundled up" to form GD packets and how many GD circles are used is illustrated in the following sections.

Tip

If you only want to transfer small amounts of data (only a few bytes) between a few CPUs, simply enter the address areas in the GD table and compile the table.

STEP 7 packs the data and allocates the resources automatically. You can see the total resources (GD circles and GD packets) used up after compilation in the first column ("GD Identifier") of the GD table.

The following sections outline the principle according to which GD packets and GD circles are used up.

12.2.1 Required Number of GD Packets

A GD packet is a data frame that is sent from one CPU only to one or more other CPUs "in one shot."

A GD packet contains the following maximum number of net data (see also the technical specifications of the CPUs):

- A maximum of 22 bytes in S7-300
- A maximum of 54 bytes in S7-400

Example 1:

You want to use the maximum send area for an S7-300 CPU to send from a data block. The bit memory area is to be used for the receiver CPU.

You enter the following in the GD table as the send area for an S7-300 CPU:

- DB8.DBB0:22 (that is an area of 22 data bytes in DB8 from data byte 0)

You enter the following in the GD table as the receive area for another CPU (must always be exactly the same size as the send area):

- MW100:11 (that is 11 memory words from MW100)

Rules

- If you want to send from more than one address area, you must take off two bytes from the maximum number of net data for each additional address area used.
- A bit address (for example, M 4.1) uses one byte of net data in the GD packet.

Example 2:

You want to send from a data block and from the process image of the outputs. The GD packet can then only be 20 bytes in size.

You enter the following in the GD table as the send areas for an S7-300 CPU:

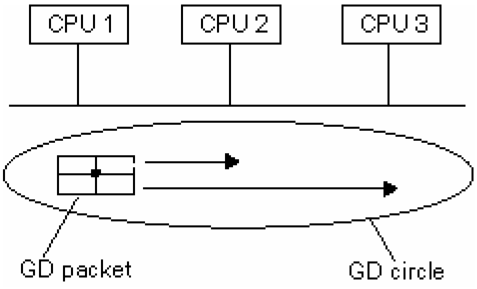
- DB8.DBB0:10 (that is an area of 10 data bytes in DB8 from data byte 0)
- QW0:10 (that is an area of 10 output words from QW0)

You enter the receive areas for the other CPUs similarly to the first example; the width of the receive area must be identical to the send area.

12.2.2 Required Number of GD Circles

What is a GD Circle?

All CPUs participating as sender or receiver in exchanging a common data packet form a GD circle.

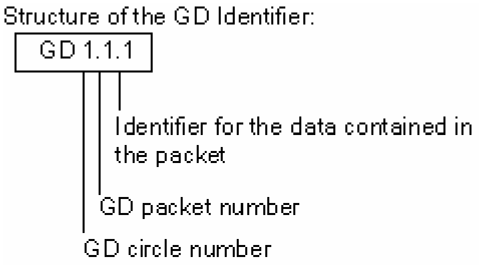


Example for Converting to a GD Table (After Compiling)

GD Identifier	CPU 1	CPU 2	CPU 3
GD 1.1.1	> MW0	IW0	IW0

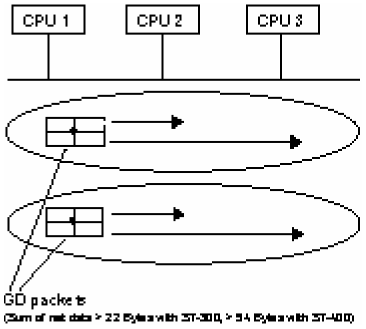
Legend for GD table:

"**>**" indicates the sender


Structure of the GD Identifier:
GD 1.1.1
Identifier for the data contained in the packet
GD packet number
GD circle number

When is Another GD Circle Required? (Case 1)

If more data are to be sent and received than can fit in one GD packet, an additional GD circle is required.

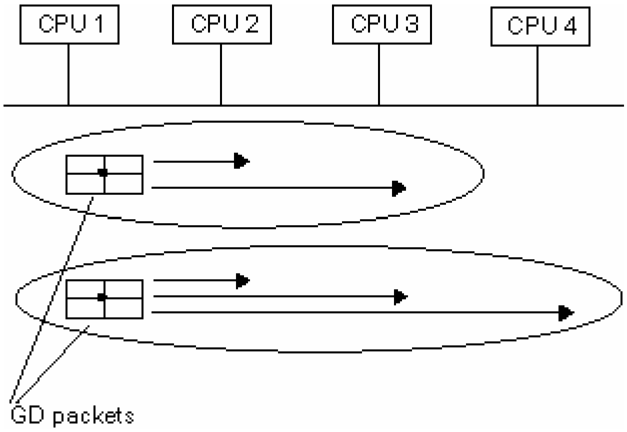

CPU 1
CPU 2
CPU 3
GD packets
(Sum of net data > 22 Bytes with ST-300, > 54 Bytes with ST-400)

Example for Converting to a GD Table (After Compiling)

GD Identifier	CPU 300 (1)	CPU 300 (2)	CPU 300 (3)
GD 1.1.1	>MW0:10	IW0:10	IW0:10
GD 2.1.1	> MW100:4	IW30:4	IW20:4

When is Another GD Circle Required? (Case 2)

An additional GD circle is also required if the send and receive CPUs are not the same (then a new GD packet must be created).



Example for Converting to a GD Table (After Compiling)

GD Identifier	CPU 300 (1)	CPU 300 (2)	CPU 300 (3)	CPU 300 (4)
GD 1.1.1	> MW0	IW0	IW0	
GD 2.1.1	> MW100:4	IW30:4	IW20:4	IW30:4

Tip

It may be advisable to determine a CPU as the receiver for a GD packet although the packet is not required for these CPUs (as CPU 4 in the example above). If the sender and the receiver CPUs are the same, the number of GD circles can be reduced in this way, for example, for the sender CPU. In the example above, CPUs 1, 2, and 3 would then only form one GD circle because both GD packets would be grouped together to form one GD packet.

Maximum Number of GD Circles

The number of GD circles is limited to 16. CPUs that have a GD circle number greater than 16 cannot be loaded with the global data configuration. However, STEP 7 does not prevent saving and compiling GD tables with a greater number of GD circles.

12.2.3 Exceptions for Calculating the GD Circles Required

Under certain conditions, the number of GD circles required can be reduced:

For S7-300:

If an S7-300 CPU (”sender CPU”) sends one GD packet only to one other S7-300 CPU (”receiver CPU”) only and this receiver CPU sends only one GD packet back to the sender CPU, only **one** GD circle is used.

This property reflects the technical data ”max. number of receive GD packets per GD circle = 1.”

In the example below you will see from the GD identifier (GD packet number) that only one GD circle is required.

Example (GD Table After Compiling)

GD Identifier	**CPU 300 (1)**	**CPU 300 (2)**
GD 1.1.1	> MW100	IW2
GD 1.2.1	IW4:3	>MW10:3

For S7-400:

If a maximum of three CPUs exchange GD packets and each of the three CPUs only sends one GD packet to the other two, only **one** GD circle is required here too.

This property reflects the technical data ”max. number of receive GD packets per GD circle = 2.”

In the example below you will see from the GD identifier (GD packet number) that only one GD circle is required.

Example (GD Table After Compiling)

GD Identifier	**CPU 400 (1)**	**CPU 400 (2)**	**CPU 400 (3)**
GD 1.1.1	> MW0	IW0	IW0
GD 1.2.1	IW2	IW2	> MW0
GD 1.3.1	IW0	> MW0	IW2

12.3 Conditions for Sending and Receiving

With the aid of a scan rate you can set the following for every CPU participating in the exchange of a GD packet:

- After how many cycles the GD packet is sent (only for the CPU designated as sender)
- After how many cycles the GD packet is received

Special Case: scan rate "0" means that the sending of the GD packet is event-driven (not cyclic) (only possible in S7-400 with SFC60/SFC61).

Example:

A scan rate of 20 entered for a GD packet at the sender CPU means that the CPU sends the GD packet at the scan cycle checkpoint after every 20 cycles.

A scan rate of 8 entered for a GD packet at the receiver CPU means that the CPU receives the GD packet at the scan cycle checkpoint after every 8 cycles (or more exactly, enters the received GD packet into the address area).

Sender Scan Rate

The following conditions should be maintained, however, to keep the communication load on the CPU at a low level:

S7-300 CPUs: scan rate × scan cycle time >= 60 milliseconds

S7-400 CPUs: scan rate × scan cycle time >= 10 milliseconds

Receiver Scan Rate

To prevent the loss of GD packets, they must be received more often than they are sent.

To guarantee this, the following must apply:

Scan rate (receiver) × scan cycle time (receiver) × scan rate (sender) × scan cycle time (sender).

12.3.1 Relationship between the Scan Rate and Scan Cycle Time

Valid Scan Rates

Valid scan rates for senders as well as for receivers are 0 and values between 1 and 255. Note that scan rates which are too small overload the CPU.

Recommendation: Maintain the default scan rate or make sure that the product of scan cycle time x scan rate is greater than 0.5 s. For higher communication requirements, you should use different communication mechanisms, such as connection by means of PROFIBUS-DP.

The scan rate 0 identifies event-driven data transmission using system functions (SFCs) in a user program (not possible with all CPUs).

If you do not enter any scan rates, the default settings are used.

Example for Scan Rates

The product of scan cycle time x scan rate determines the time interval of global data exchange.

Assumption: The user program in a CPU 412 has a scan cycle time of approximately 50 ms. The default scan rate is 22.

50 ms x 22 = 1100 ms

Global data are thus sent or received by this CPU every 1.1 s.

If the scan cycle time is increased by the user program to 80 ms, for example, global data will be sent or received every 80 ms x 22 = 1760 s.

If the value 1100 s is reached again, you must recalculate the scan rate.

Scan rate (new) = 1100 ms/80 ms = 13.75

That is, because of the extension of the scan cycle, you have to set the scan rate to 14 in order to maintain the time interval.

12.4 Response Time

The response time for two stations that exchange GD packets via an MPI subnet can be calculated approximately.

Requirement for Calculating the Response Time

- Transmission rate 187.5 kbit/s
- No other communication by means of MPI (for example, by connected programming devices or operator panels)

Calculation

response time = approx. scan rate (sender) x scan cycle time (sender) + scan rate (receiver) x scan cycle time (receiver) + number (of MPI nodes) x 10 ms

For higher transmission rates, the factor "x 10 ms" is smaller; however, this factor does not diminish linearly with an increase in transmission speed.

Tip

Because the response time is very dependent on the scan cycle times and on the other communication load over the MPI, you must determine the response time in your system empirically and, where appropriate, adapt the scan rate.

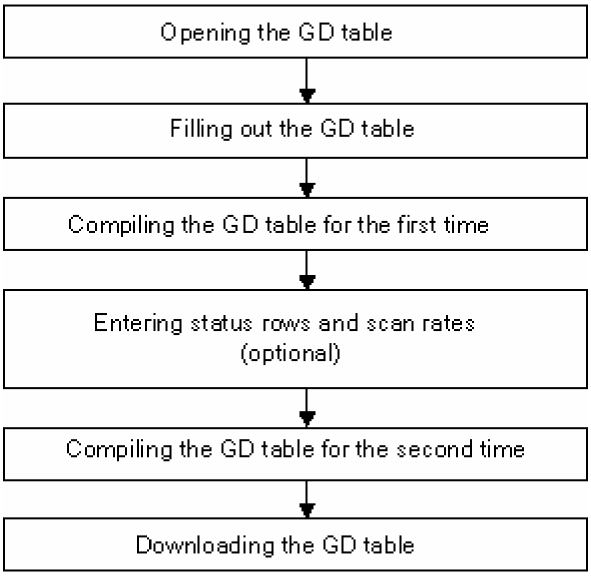
12.5 How to Configure and Save Global Data Communication

12.5.1 Procedure for Configuring Global Data Communication

Requirement

You must have configured an MPI subnet with all the required stations.

Overview Procedure


Opening the GD table
Filling out the GD table
Compiling the GD table for the first time
Entering status rows and scan rates (optional)
Compiling the GD table for the second time
Downloading the GD table

12.5.2 Opening the GD Table

There are two ways of opening a global data table:

- Open the entire GD table for a subnet
- Open the GD table for a CPU

Opening the GD Table for a Subnet (Recommended)

- In the SIMATIC Manager select an MPI subnet and select the menu command **Options > Define Global Data**.
  Result: The GD table for the selected subnet is opened.

Opening the GD Table for a CPU

The following option for opening a global data table is recommended if you want to retrieve the GD table from the CPU during servicing and when searching for faults.

1. Select the menu command **GD Table > Open > Global Data for CPU**. The dialog box "Open" appears, in which you can select the project and station for the required CPU.
2. Select one of the following options:
   - "Online" to read the data directly from the CPU, or
   - "Offline" to obtain the system data for the CPU from the offline project
3. Select the project and double-click the station in which the required CPU is located to open it.
4. Select the "Blocks" object of the CPU whose GD table you wish to display.
5. Confirm your selection by clicking "OK."
   Result: A global data table appears in which all CPUs participating in global data communication are entered in the column headers. However, the table only contains the values for the selected CPU.

To complete the empty cells of the GD table, you must update the table.

- Select the menu command **View > Update**. The GD table is now displayed with all entries.

12.5.3 Tips for Working with GD Tables

Inserting Global Data Rows

- Select the menu command **Insert > GD Row**

Deleting Global Data Rows

- Select a row and select the menu command **Edit > Delete**.

Inserting and Deleting CPU Columns

- Select the menu command **Insert > CPU Column**.

Deleting CPU Columns

- Select a column and select the menu command **Edit > Delete**.

Note on Deleting CPU Columns

If you delete a CPU column from a global data table, the corresponding system data of the CPU (offline) are deleted. Bear in mind that you must also delete the online data in the CPU for the deleted CPU column.

Requirement: The programming device must be connected to the CPU via the multipoint interface and the CPU must be in STOP mode.

Select the menu command **GD Table > Delete Global Data for CPU** and in the dialog box that follows, select the option "Online." Select the "Blocks" folder and confirm by clicking "OK."

Modifying the Column Width

The width of each CPU column can be modified. You can also make a column so narrow that it cannot be seen.

- Position the cursor on the right-hand side of the CPU cell in the table header, press and hold down the left mouse button, and drag the border horizontally to resize the column.

12.5.4 Filling Out the GD Table

Requirement

The scan rates and the GD status must be toggled off in the GD table.

Entering CPUs in the Table Header

1. In the global data table, click on a column in the table header. The column is then selected.
2. Select the menu command **Edit > Assign CPU**. The dialog box "Open" will then appear. Alternatively, you can open this box by double-clicking the column header.
3. Select the current project and double-click to open the station for the required CPU.
4. Select the CPU and confirm your selection by clicking "OK."
   Result: The name of the selected CPU is displayed in the table header.

Entering Data in GD Rows

Requirement: You must have entered the relevant CPU in the table header (see above).

1. Position the cursor in a table cell and enter the required address. You can only enter absolute addresses (for example, IW0); symbolic entries are not possible.
   Tip: A set of continuous addresses of the same data type requires only one entry in the global data table. In this case, enter a colon after the address and then the repetition factor. The repetition factor defines the size of the data area.

   Example: IW4: 3 means: 3 words from IW4.
2. Switch from overwrite to insert mode by pressing the F2 key.
3. Edit the table as you would normally. You can also use the menu commands **Edit > Cut**, **Edit > Copy**, or **Edit >Paste**.
4. Finish your entries with RETURN.

Note

In a GD circle, use only either the communication bus (within an S7-400 station) or the MPI subnet (outside the stations). Mixed operation is not possible.

Defining Sender and Receiver Cells

Each global data row contains only one sender and one or more receivers. ">" indicates a sender. All cells in the global data table are preset as receivers.

- To define a data cell as a sender, highlight the cell and select the menu command **Edit > Sender**.
- To define a data cell as a receiver, highlight the cell and select the menu command **Edit > Receiver**.

Note

Cells that contain timers and counters can be used only as senders.

12.5.5 Saving and Compiling the GD Table for the First Time

Saving

When saving, you store the data which you have entered in your global data table in a source file.

- Select the menu command **GD Table > Save**

  or

1. Select the menu command **GD Table > Save As**
2. Navigate to the project in which you want to save the global data table.
3. Confirm with "OK."

Note

In order for the changes you made in the global data table to be saved in the system data as well, you must compile the global data table.

The data will automatically be saved in the system data of the relevant CPUs directly after compiling.

The data consistency between the source file and the system data can only be guaranteed if each change to the GD table is saved both in the source file (save) and in the system data (compile).

Compiling

A compiler run is necessary in order to compile the data you enter in the global data table into a language that the CPUs can understand.

The tabular data in the global data table are compiled into system data that the CPUs can process.

For each CPU column, the compiler creates the exact system data necessary for communicating with that particular CPU. For this reason, each CPU has its own GD configuration.

- Click the appropriate button in the toolbar or select the menu command **GD Table > Compile**. The global data table will be compiled to phase 1.
  Result: STEP 7 checks:
  - The validity of the CPU specified in the CPU column headers
  - The syntax of the addresses you entered in the table cells.
  - The size of the data areas for the sender and receiver (the data areas for the sender and receiver must be of equal size)
  - That the global data in a row are exchanged either via the communication bus only or via the MPI subnet only. Mixed operation is not possible.

After the first compilation has been completed successfully, the GD table is in phase 1. In phase 1 you can edit the status rows and scan rates in the GD table.

12.5.6 Entering Scan Rates

Introduction

Global data exchange occurs as follows:

- The sender CPU sends the global data at the end of a cycle.
- The receiver CPU reads the data at the start of a cycle.

With the aid of the scan rate, you can determine after how many cycles the data will be sent or received.

Procedure

1. Compile the global data table if it is not already in phase 1 (you can view this information in the status bar at the bottom edge of the screen).
2. If no scan rates are displayed in the GD table, select the menu command **View > Scan Rates**.
3. Enter the required scan rates. You can enter data only in the columns in which the GD packet in question has entries.
   Note: When you view status rows and/or scan rate rows, you can edit these rows only.
4. Compile the global data table again (phase 2).

12.5.7 Entering Status Rows

Introduction

For each global data (GD) packet you can specify a status double word for each participating CPU. Status double words have the ID "GDS" in the table. If you assign the status double word (GDS) to a CPU address of the same format, you can evaluate the status in the user program or in the status row.

Group Status

STEP 7 creates a group status (GST) for all global data packets.

The group status, which is also a double word with an identical structure to the status double word (GDS), is formed by linking all the status double words with an OR logic operation.

Procedure

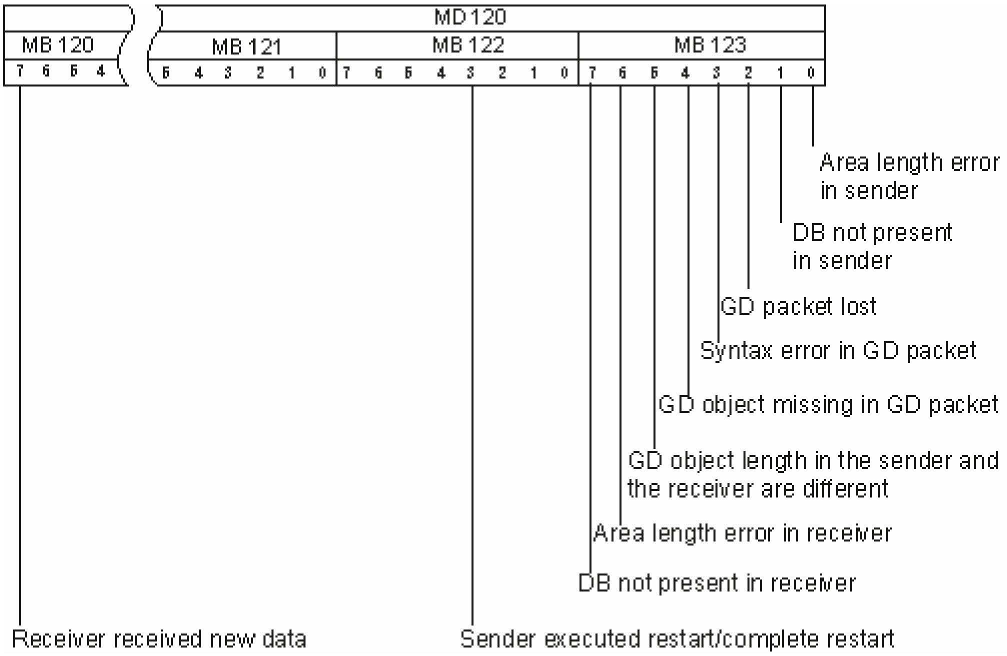
1. Compile the global data table if it is not already in phase 1 (you can view this information in the status bar at the bottom edge of the screen).
2. If no GD status rows are displayed in the GD table, select the menu command **View > GD Status**.
3. Enter the required status double words. You can enter data only in the columns in which the GD packet in question has entries. Enter addresses according to the syntax of the STEP 7 programming languages.
   Note: When you view status rows and/or scan rate rows, you can edit these rows only.
4. Compile the global data table again (phase 2).

Structure of the Status Double Word

The following figure shows the structure of the status double word and an explanation of the bits set.

A bit remains set until it is reset by the user program or via a programming device operation.

Any bits not listed are reserved and have no significance at present. The global data status occupies a double word; to make it clearer, MD120 has been used in the figure.


MD120
MB 120
MB 121
MB 122
MB 123
7 6 5 4
5 4 3 2 1 0
7 6 5 4 3 2 1 0
7 6 5 4 3 2 1 0
Area length error in sender
DB not present in sender
GD packet lost
Syntax error in GD packet
GD object missing in GD packet
GD object length in the sender and the receiver are different
Area length error in receiver
DB not present in receiver
Receiver received new data
Sender executed restart/complete restart

12.5.8 Compiling the GD Table for a Second Time

After editing the status rows and/or scan rate rows, recompile the global data table to ensure that the new information is incorporated in the system data.

The system data generated in phase 1 are sufficient for running global data communication. You can download this data to the CPUs from the programming device. Phase 2 is necessary only if you want to change default values for the scan rates or make entries in the status rows.

12.6 Global Data Transmission Using System Functions

Event-driven data transmission is supported for S7-400 CPUs. A command from the system functions (SFC) in the user program determines when the data exchange occurs. SFC60 "GD_SND" (send global data) is available to send global data and SFC61 "GD_RCV" (receive global data) is available to receive global data. If the CPU supports event-driven data transmission, you must enter the scan rate "0" in the global data table. If a value greater than "0" is entered, global data transmission is cyclic and event-driven.

13 Downloading

13.1 Downloading a Configuration to a Programmable Controller

13.1.1 Requirements for Downloading

Requirements for Downloading to the Programmable Controller

- There must be a connection between your programming device and the CPU in the programmable controller (for example, via the multipoint interface).
- Access to the programmable controller must be possible.
- For the download of blocks to the PLC, the entry "STEP 7" must have been selected for "Use" in the object properties dialog of the project.
- The program you are downloading has been compiled without errors.
- The CPU must be in an operating mode in which downloading is permitted (STOP or RUN-P). Note that in RUN-P mode the program will be downloaded a block at a time. If you overwrite an old CPU program doing this, conflicts may arise, for example, if block parameters have changed. The CPU then goes into STOP mode while processing the cycle. We therefore recommend that you switch the CPU to STOP mode before downloading.
- If you opened the block offline and want to download it, the CPU must be linked to an online user program in the SIMATIC Manager.
- Before you download your user program, you should reset the CPU to ensure that no "old" blocks are on the CPU.

STOP Mode

Set the operating mode from RUN to STOP before you do the following:

- Download the complete user program or parts of it to the CPU
- Execute a memory reset on the CPU
- Compress the user memory

Restart (Warm Restart (Transition to RUN Mode)

If you execute a restart (warm restart) in the "STOP" mode, the program is restarted and first processes the startup program (in the block OB100) in STARTUP mode. If the startup is successful, the CPU changes to RUN mode. A restart (warm restart) is required after the following:

- Resetting the CPU
- Downloading the user program in STOP mode

13.1.2 Differences Between Saving and Downloading Blocks

You should always distinguish between saving and downloading blocks.

	Saving	Downloading
Menu commands	**File > Save** **File > Save As**	**PLC > Download**
Function	The current status of the block in the editor is saved on the hard disk of the programming device.	The current status of the block in the editor is only downloaded to the CPU.
Syntax check	A syntax check is run. Any errors are reported in dialog boxes. The causes of the errors and the error locations are also shown. You must correct these errors before you save or download the block. If no errors are found in the syntax, the block is compiled into machine code and either saved or downloaded.	A syntax check is run. Any errors are reported in dialog boxes. The causes of the errors and the error locations are also shown. You must correct these errors before you save or download the block. If no errors are found in the syntax, the block is compiled into machine code and either saved or downloaded.

The table applies independent of whether you have opened the block online or offline.

Tip for Block Changes - Save First Then Download

To enter newly created blocks or changes in the code section of logic blocks, in declaration tables or to enter new or changed data values in data blocks, you must save the respective block. Any changes you make in the editor and transfer to the CPU using the menu command **PLC > Download**, -for example, for testing small changes-, must also be saved on the hard disk of the programming device in every case before you exit the editor. Otherwise, you will have different versions of your user program in the CPU and on the programming device. It is generally recommended that you save all changes first and then download them.

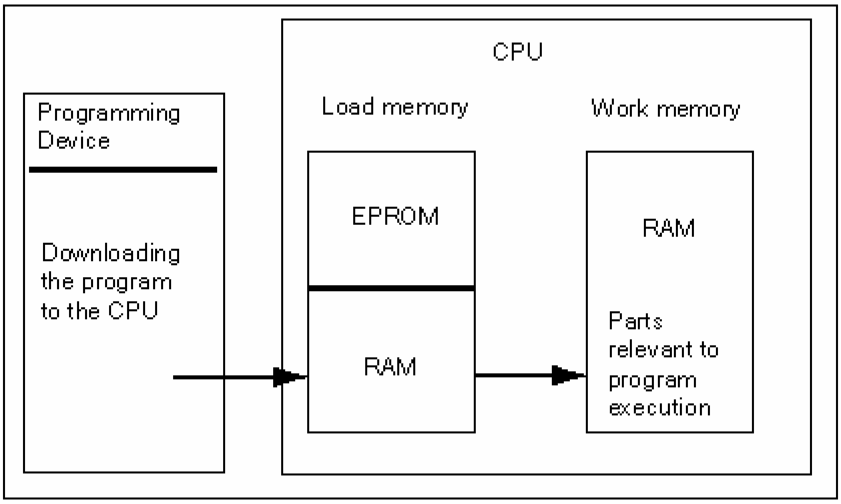
13.1.3 Load Memory and Work Memory in the CPU

After completing the configuration, parameter assignment, and program creation and establishing the online connection, you can download complete user programs or individual blocks to a programmable controller. To test individual blocks, you must download at least one organization block (OB) and the function blocks (FB) and functions (FC) called in the OB and the data blocks (DB) used. To download the system data created when the hardware was configured, the networks configured, and the connection table created to the programmable controller, you download the object "System Data".

You download user programs to a programmable controller using the SIMATIC Manager, for example, during the end phase of the program testing or to run the finished user program.

Relationship - Load Memory and Work Memory

The complete user program is downloaded to the load memory; the parts relevant to program execution are also loaded into the work memory.


CPU
Programming Device
Downloading the program to the CPU
Load memory
EPROM
RAM
Work memory
RAM
Parts relevant to program execution

CPU Load Memory

- The load memory is used to store the user program without the symbol table and the comments (these remain in the memory of the programming device).
- Blocks that are not marked as required for startup will be stored only in the load memory.
- The load memory can either be RAM, ROM, or EPROM memory, depending on the programmable controller.
- The load memory can also have an integrated EEPROM part as well as an integrated RAM part (for example, CPU 312 IFM and CPU 314 IFM).
- In S7-400, it is imperative that you use a memory card (RAM or EEPROM) to extend the load memory.

CPU Work Memory

The work memory (integrated RAM) is used to store the parts of the user program required for program processing.

Possible Downloading/Uploading Procedures

- You use the download function to download the user program or loadable objects (for example, blocks) to the programmable controller. If a block already exists in the RAM of the CPU, you will be prompted to confirm whether or not the block should be overwritten.
- You can select the loadable objects in the project window and download them from the SIMATIC Manager (menu command: PLC > Download).
- When programming blocks and when configuring hardware and networks you can directly download the object you were currently editing using the menu in the main window of the application you are working with (menu command: PLC > Download).
- Another possibility is to open an online window with a view of the programmable controller (for example, using **View > Online** or **PLC > Display Accessible Nodes**) and copy the object you want to download to the online window.

Alternatively you can upload the current contents of blocks from the RAM load memory of the CPU to your programming device via the load function.

13.1.4 Download Methods Dependent on the Load Memory

The division of the load memory of a CPU into RAM and EEPROM areas determines the methods available for downloading your user program or the blocks in your user program. The following methods are possible for downloading data to the CPU:

Load Memory	Method of Loading	Type of Communication between PG and PLC
RAM	Downloading and deleting individual blocks	Online PG - PLC connection
	Downloading and deleting a complete user program	Online PG - PLC connection
	Reloading individual blocks	Online PG - PLC connection
Integrated (S7-300 only) or plugin EPROM	Downloading complete user programs	Online PG - PLC connection
Plugin EPROM	Downloading complete user programs	External loading of the EPROM and inserting the memory card or via online connection on the PLC where the EPROM is inserted.

Downloading to the RAM via Online Connection

In the programmable controller the data are lost if there is a power failure and the RAM is not backed up. The data in the RAM will then be lost in this case.

Saving to EPROM Memory Card

Blocks or the user program are saved on an EPROM memory card which is then inserted in a slot on the CPU.

Memory cards are portable data media. They are written by the programming device and then inserted in the appropriate slot on the CPU.

The data stored on them are retained following power down and when the CPU is reset. The contents of the EPROM are copied to the RAM area of the CPU memory again when power returns following a memory reset of the CPU and power down if the RAM is not backed up.

Saving in the Integrated EPROM

For the CPU 312, you can also save the contents of the RAM to the integrated EPROM. The data in the integrated EPROM are retained during power down. The contents of the integrated EPROM are copied to the RAM area of the CPU memory again when power returns following power down and a memory reset of the CPU if the RAM is not backed up.

13.1.5 Updating Firmware in Modules and Submodules Online

As of STEP 7 V5.1 Service Pack 3, you can update modules or submodules on a station in a standardized way online. To do so, proceed as described below:

Concept

To update the firmware on a module (CPU, IM, etc.) or a submodule (DI, DO, etc.), you can download the files (*.UPD) containing the latest firmware on the Internet ("http://www.siemens.com/automation/support").

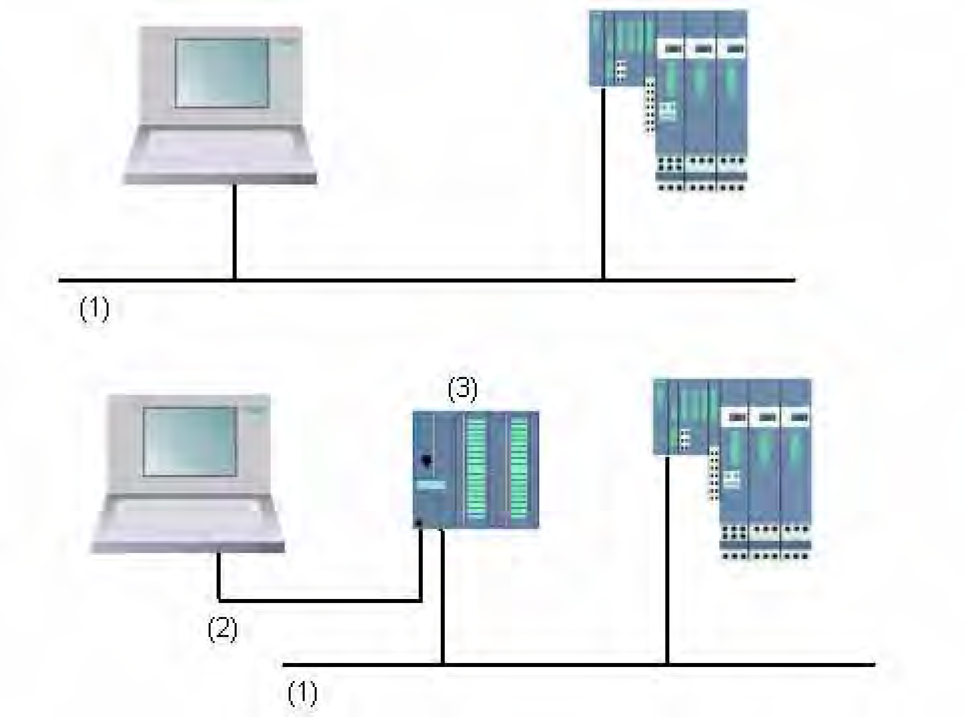
Select one of these files and download it to the module (PLC Menu).

Prerequisites

The module in the station or module whose firmware is to be updated must be available online. This means that the programming device (PG) is connected to the same MPI PROFIBUS or Ethernet as the module whose firmware is to be updated. The firmware can also be updated when the programming device (PG) is connected to the MPI interface of the DP master CPU and the module whose firmware is to be updated is connected at the PROFIBUS of the DP interface or the Ethernet of the PN interface. The CPU must support S7 routing between the MPI interface and the DP interface or between the MPI interface and the PN interface.

The module or submodule itself must support Firmware updates.

The files containing the latest firmware versions must be available in the file system on your PG/PC. Only files for **one** firmware version must be in one folder.



(1) PROFIBUS or Ethernet subnet

(2) MPI subnet

(3) CPU with MPI interface and DP interface or PN interface (with S7 routing)

Procedure in HW Config

1. Open the station containing the module to be updated.
2. Select the module
   For PROFIBUS DP interface modules such as an IM 151, select the icon for the DP slave. In this case, it is the one that stands for ET 200S.
3. Follow the same procedure for PROFINET IO devices.
   If you want to update the firmware of modules of a DP slave or IO device, click the "Change Slot" button and select the slot of the module you want to update in the "Change Slot" dialog.
4. Select the menu command PLC > Update Firmware.
   You can only activate the menu command if the selected module / DP slave / IO device or the selected submodule supports the "Update firmware" function.
5. In the "Update firmware" dialog that is displayed, click the "Browse" button and select the path to the firmware update files (*.UPD).
6. After you have selected a file, the lower fields of the "Update firmware" dialog will contain information telling you for which modules the file is suitable and as of which firmware version.
7. Click the "Run" button.
   STEP 7 checks whether the selected file can be interpreted by the module. If the check result is positive, the file is downloaded to the module.
   If the operating mode of the CPU needs to be changed, dialogs will prompt you to carry out these steps.
   The module then carries out the firmware update independently.
   Note: For a firmware update, such as to a CPU 317-2 PN/DP, a separate connection is usually established to the CPU. In such case, the process can be interrupted. If no resources are available for another connection, the existing connection is automatically used instead. In this case, the connection cannot be interrupted. The "Cancel" button in the transfer dialog is grayed out and unavailable.
8. In STEP 7, check (read out the CPU diagnostic buffer) whether the module was able to start up with the new firmware.

Procedure in SIMATIC Manager

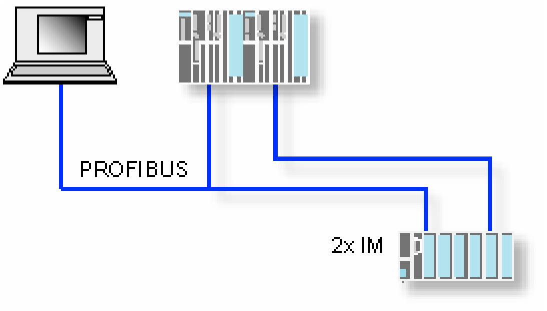
The procedure corresponds to that in HW Config. The menu command also is **PLC > Firmware update**. STEP 7 however only checks if the module supports the function at the time of the execution.

Firmware Update for Modules in Redundant Mode

As of STEP 7 V5.4, a firmware updates for a modules during redundant mode is supported, such as for the IM 153-2BA00 with an active backplane bus at an H station. You can carry out a firmware update for the redundant IMs in one process; the redundant IM will automatically be provided with the latest firmware version.

Requirement: The programming device (PG) must be connected to the same PROFIBUS as one of the IMs and you perform the update by means of "Accessible Nodes" in the SIMATIC Manager.

Principle



Consequences of updating the firmware during operation

You can decide to activate the new firmware immediately after updating via an option in the update dialog.

If you select this option, the station performs a restart like after POWER OFF/POWER ON. As result, it may happen that a CPU remains in STOP mode or the processing of the user program is adversely affected. You will need to take appropriate precautions in the operation of your plant to anticipate and accommodate these conditions.

For example, during a restart all modules of the station will fail, including the existing F I/O.

The F I/O outputs a communication error to the interface during POWER OFF and switches off safely - it is passivated. This passivation is not cleared by restarting the interface. You must depassivate the modules individually. However, the safety-related applications will not run as a result of this.

13.1.6 Downloading a Program to the S7 CPU

13.1.6.1 Downloading with Project Management

1. In the project window, select the user program or the blocks you want to download.
2. Download the selected objects to the programmable logic controller by selecting the menu command **PLC > Download**.

Alternative Procedure (Drag & Drop)

1. Open an offline window and an online window of your project.
2. Select the objects you want to download in the offline window and drag them to the online window.

13.1.6.2 Downloading without Project Management

1. Open the "Accessible Nodes" window using the menu command **PLC > Display Accessible Nodes** or by clicking the corresponding button in the toolbar.
2. Double-click in the "Accessible Nodes" window on the required node ("MPI=...") to display the "Blocks" folder.
3. Open the library or the project from which you want to download the user program or blocks to the programmable logic controller. Use the menu command **File > Open** for this.
4. In the window which opens for the project or the library, select the objects you want to download.
5. Download the objects to the programmable logic controller by copying them to the "Blocks" folder in the "Accessible Nodes" window using drag & drop.

13.1.6.3 Reloading Blocks in the Programmable Controller

You can overwrite blocks which already exist in the load memory (RAM) or work memory of the CPU in the S7 programmable logic controller with a new version (reload them). The existing version is then overwritten.

The procedure for reloading S7 blocks is the same as for downloading. A prompt simply appears, querying whether you want to overwrite the existing block.

A block stored in the EPROM cannot be deleted but is declared invalid once it is reloaded. The replacement block is loaded in the RAM. This creates gaps in the load memory or the work memory. If these gaps eventually mean that no new blocks can be downloaded, you should compress the memory.

Note

If the power goes down and then returns and the RAM does not have a battery backup, or following a memory reset of the CPU the "old" blocks become valid again.

13.1.6.4 Saving Downloaded Blocks on Integrated EPROM

For CPUs that have an integrated EPROM (such as CPU 312), you can save blocks from the RAM to the integrated EPROM so as not to lose the data following power off or memory reset.

1. Use the menu command **View > Online** to display a window containing the online view of an open project or open the "Accessible Nodes" window by clicking the "Accessible Nodes" button in the toolbar or selecting the menu command **PLC > Display Accessible Nodes**.
2. Select the S7 or M7 program in the online window of the project or the node in the "Accessible Nodes" window.
3. Select the "Blocks" folder on the CPU which you want to save using one of the following methods:
   - In the online window of the project if you are working with project management
   - In the "Accessible Nodes" window if you are working without project management
4. Select the menu command **PLC > Save RAM to ROM**.

13.1.6.5 Downloading via EPROM Memory Cards

Requirements

For access to EPROM memory cards in the programming device which are intended for an S7 programmable logic controller, you will require the appropriate EPROM drivers. For access to EPROM memory cards which are intended for an M7 programmable control system, the Flash File System must be installed (only possible on the PG 720, PG 740, and PG 760). EPROM drivers and the Flash File System are offered as options when you install the STEP 7 Standard package. If you are using a PC, an external prommer will be required to save to EPROM memory cards.

You can also install the drivers at a later date. To do this, call up the corresponding dialog box via **Start > Simatic > STEP 7 > Memory Card Parameter Assignment** or via the Control Panel (double-click the "Memory Card Parameter Assignment" icon).

Saving on the Memory Card

To save blocks or user programs to a memory card, proceed as follows:

1. Insert the memory card in the slot of your programming device.
2. Open the "Memory Card" window by:
   - Clicking the button for "Memory Card" in the toolbar. If necessary, activate the toolbar using the menu command **View > Toolbar**.
   - Alternatively, select the menu command **File > S7 Memory Card > Open**.
3. Open or activate one of the following windows displaying the blocks you want to save: The following windows are possible:
   - Project window, "ONLINE" view
   - Project window, "offline" view
   - Library window
   - "Accessible Nodes" window
4. Select the "Blocks" folder or individual blocks and copy them to the "S7 Memory Card" window.
5. If a block already exists on the memory card, an error message is displayed. In this case, erase the contents of the memory card and repeat the steps from 2.

13.2 Compiling and Downloading Several Objects from the PG

13.2.1 Requirements for and Notes on Downloading

Downloading Block Folders

For block folders, only logic blocks can be downloaded. Other objects in the block folder, such as system data (SDBs), etc. cannot be downloaded here. SDBs are downloaded through the "Hardware" object.

Note

For PCS 7 projects, blocks cannot be downloaded using the dialog "Compile and Download Objects"- just as they cannot be downloaded from the SIMATIC Manager. For PCS 7 projects, the following applies: PLCs must only be downloaded to by means of CFCs in order to ensure correct sequencing during the download. This must be done to prevent the CPU from going into STOP mode.

To determine whether the given project is a PCS 7 project, check the project properties.

Downloading the F-Shares of Failsafe Controllers

For security reasons, a password must be entered before modified F-shares can be downloaded. For this reason, with the "Compile and Download Objects" function, the download procedure will be aborted with an error message. In this case, load the appropriate parts of the program along with the optional package to the PLC.

Downloading the Hardware Configuration

Downloading the hardware configuration (i.e. downloading the offline SDBs) by means of the "Compile and Download Objects" function will only run without interruption for all selected objects if no error messages or prompts are triggered. The following section provides information on how to avoid such messages or prompts.

Requirements for Downloading the Hardware Configuration

- CPUs must be in STOP mode.
- It must be possible to establish online connections to the CPUs. In the case of the selected CPU or the selected block folder, password-projected CPUs require an authorized connection or entry of a password ("Edit" button) before the "Compile and Download Objects" function can be run.
- The interface of the target system that is being used for downloading must not be reconfigured to any substantial extent:
    - The interface address must not be changed.
    - If you change the network settings, this may mean that not all the modules will be able to be accessed.
    - In the case of H-CPUs, you can select the CPU to receive the download (H-CPU 0 or H-CPU 1) before running the "Compile and Download Objects" function (Select the "CPU" object and then click the "Edit" button).
- The following CPU parameters must not be changed:
    - The maximum size for local data and communications resources on the CPU ("Memory" tab)
    - The password protection for the F-CPU ("Protection" tab)
- For each configured module, the following conditions must be fulfilled:
    - The order number for the configured module must be identical with the order number of the module that is actually inserted.
    - The firmware version of the configured module must not be higher than the firmware version of the module that is actually inserted.
    - The station name, the name of the module and the plant designation must not have changed since the last download. However, you can assign a new plant designation.

Tips on the Download Procedure

- All offline SDBs will be downloaded (that is, in addition to the hardware configuration, also the connection SDBs and SDBs that were created through global data configurations).
- Downloading is only carried out if no errors occurred during the previous compilation process.
- During the download, any error feedback messages are suppressed. For example, if a CPU memory bottleneck occurs, the data will be compressed automatically without the user being informed.
- After the download is complete, the downloaded modules will be in STOP mode (except for those modules that are automatically stopped and restarted without the user being informed).

Tip

If, after the download is completed, a message appears stating that the download of the object was completed with warnings, then be sure to view the contents of the log. It may be that the object was either not downloaded or was not downloaded completely.

13.2.2 Compiling and Downloading Objects

In the "Compile and download objects" dialog you prepare the objects that can be selected in your project or multiproject for transfer to the PLC and their subsequent download (if desired). This dialog can be used for objects in a station, a project or a multiproject.

Depending on the object selected, certain information may not be displayed. In addition, not all the functions described below may be available for these objects. In particular, these restrictions may apply to objects that were created with optional software packages.

For blocks in a block folder "compile" means that the consistency of the blocks is checked. In the following, for simplicity, the consistency check for blocks will be referred to as compilation.

Procedure:

1. In SIMATIC Manager, select the object that you want to compile, or compile and download. The following objects can be selected in the SIMATIC Manager:
   - Multiproject
   - Project
   - Station
   - S7 program without station assignment
2. In the SIMATIC Manager, select menu command **PLC > Compile And Download Objects**.
3. Select "Only compile" if you want to perform a check of the blocks without downloading them to the PLC. Select this option if you do not want to download any of these objects to the PLC.
   Note: Stations with missing HSPs are not compiled and loaded (the check boxes are not visisble).
4. To prevent incomplete downloads to stations due to compilation errors, select the check box "No download on compilation error". If this check box is selected, nothing will be downloaded. If the check box is not selected, then all objects compiled without error are downloaded. Objects that caused an error during compilation are not downloaded.
   If you want to compile and download connections, select the corresponding check box for the "Connections" object.
5. A multiproject is particularly suited for use as a starting point, since all connection partners for cross-project connections can also be downloaded from this object.
6. In the "Compile" and "Download" columns, select the objects that you want to compile or download. You selections will be indicated by checkmarks. If you selected "Compile only" in Step 3, the "Download" column will be grayed out and unavailable.
7. Click on "Start" to begin the compilation.
8. Follow the instructions on the screen.

After the compilation or download is complete, a full log is displayed. You can open the full log or single-object log at any time:

- Click on the "All" button to view the full log of the complete action.
- Click on "Single object" button to view only the log of the object you have selected in the object table.

Special considerations when compiling and downloading connections

If, in a module, you select the "Connections" object as the one to be **compiled**, then STEP 7 automatically selects the corresponding "Connections" object in the connection partner. By doing this, STEP 7 always creates consistent configuration data (system data blocks). The automatically selected objects cannot be directly deselected manually. However, the selection is automatically removed if the originally selected "Connections" object is also deselected.

If, in a module, you select the "Connections" object as the one to be **downloaded**, then STEP 7 automatically selects the "Compile" check box. In addition, STEP 7 also selects the "Compile" and "Download" check boxes for all connection partners. If only objects of the "Connections" type were selected, you can also download the connections when the CPU is in RUN-P operating mode.

You can use NetPro to download individual connections.

Compiling and downloading hardware: effects on connections

If you select the "Hardware" object as the one to be compiled or downloaded, all "Connections" objects below the hardware selected are also automatically selected to be compiled or downloaded. In this case, however, the connection objects at the connections partners are **not** automatically selected as well!

13.3 Uploading from the Programmable Controller to the PG/PC

This function supports you when carrying out the following actions:

- Saving information from the programmable controller (for example, for servicing purposes)
- Fast configuring and editing of a station, if the hardware components are available before you start configuring.

Saving Information from the Programmable Controller

This measure may be necessary if, for example, the offline project data of the version running on the CPU are not, or only partially, available. In this case, you can at least retrieve the project data that are available online and upload them to your programming device.

Fast Configuring

Entering the station configuration is easier if you upload the configuration data from the programmable controller to your programming device after you have configured the hardware and restarted (warm restart) the station. This provides you with the station configuration and the types of the individual modules. Then all you have to do is specify these modules in more detail (order number) and assign them parameters.

The following information is uploaded to the programming device:

- S7-300: Configuration for the central rack and any expansion racks
- S7-400: Configuration of the central rack with a CPU and signal modules without expansion racks
- Configuration data for the distributed I/O cannot be uploaded to the programming device.

This information is uploaded if there is no configuration information on the programmable controller; for example, if a memory reset has been carried out on the system. Otherwise, the **Upload** function provides much better results.

For S7-300 systems without distributed I/O, all you have to do is specify these modules in more detail (order number) and assign them parameters.

Note

When you upload data (if you do not already have an offline configuration), STEP 7 cannot determine all the order numbers of the components.

You can enter the "incomplete" order numbers when you configure the hardware using the menu command **Options > Specify Module**. In this way, you can assign parameters to modules that STEP 7 does not recognize (that is, modules that do not appear in the "Hardware Catalog" window); however, STEP 7 will not then check whether you are keeping to the parameter rules.

Restrictions when Uploading from the Programmable Controller

The following restrictions apply to the data uploaded from the programmable controller to the programming device:

- Blocks do not contain any symbolic names for parameters, variables, and labels
- Blocks do not contain any comments
- The entire program is uploaded with all the system data, whereby the system can only continue to process the system data belonging to the "Configuring Hardware" application
- The data for global data communication (GD) and configuring symbol-related messages cannot be processed further
- Force jobs are not uploaded to the programming device with the other data. They must be saved separately as a variable table (VAT)
- Comments in the module dialog boxes are not uploaded
- The names of the modules are only displayed if this option has been selected during configuration (HW Config: the option "Save object names in the programmable logic controller" in the dialog box under **Options > Customize**).

13.3.1 Uploading a Station

Using the menu command **PLC > Upload Station** you can upload the current configuration and all blocks from the programmable controller of your choice to the programming device.

To do this, STEP 7 creates a new station in the current project under which the configuration will be saved. You can change the preset name of the new station (for example, "SIMATIC 300-Station(1)"). The inserted station is displayed both in the online view and in the offline view.

The menu command can be selected when a project is open. Selecting an object in the project window or the view (online or offline) has no effect on the menu command.

You can use this function to make configuring easier.

- For S7-300 programmable controllers, the configuration for the actual hardware configuration is uploaded including the expansion racks, but without the distributed I/O (DP).
- For S7-400 programmable controllers, the rack configuration is uploaded without the expansion racks and without the distributed I/O.

With S7-300 systems without distributed I/O, all you have to do is specify the modules in more detail (order number) and assign them parameters.

Restrictions when Uploading Stations

The following restrictions apply to the data uploaded to the programming device:

- Blocks do not contain any symbolic names for parameters, variables, and labels
- Block do not contain any comments
- The entire program is uploaded with all the system data, whereby not all the data can be processed further
- The data for global data communication (GD), configuring symbol-related messages, and configuring networks cannot be processed further
- Force jobs cannot be uploaded to the programming device and then loaded back to the programmable controller.

13.3.2 Uploading Blocks from an S7 CPU

You can upload S7 blocks from the CPU to the hard disk of the programming device using the SIMATIC Manager. Uploading blocks to the programming device is useful in the following situations:

- Making a backup copy of the current user program loaded in the CPU. This backup can then be downloaded again, for example, following servicing or following a memory reset of the CPU by maintenance personnel.
- You can upload the user program from the CPU to the programming device and edit it there, for example, for troubleshooting purposes. In this case you do not have access to symbols or comments for program documentation. Therefore we recommend that this procedure is used only for service purposes.

13.3.3 Editing Uploaded Blocks in the PG/PC

13.3.3.1 Editing Uploaded Blocks in the PG/PC

Being able to upload blocks from the CPU to the programming device has the following uses:

- During the test phase, you can correct a block directly on the CPU and document the result.
- You can upload the current contents of blocks from the RAM load memory of the CPU to your programming device via the load function.

Note

Time stamp Conflicts when Working Online and Offline

The following procedures lead to time stamp conflicts and should therefore be avoided.

Time stamp conflicts result when you open a block online if:

- Changes made online were not saved in the offline S7 user program
- Changes made offline were not downloaded to the CPU

Time stamp conflicts result when you open a block offline if:

- An online block with a time stamp conflict is copied to the S7 user program offline and the block is then opened offline.

Two Distinct Cases

When uploading blocks from the CPU to the programming device, remember that there are two distinct situations:

1. The user program to which the blocks belong is located on the programming device.
2. The user program to which the blocks belong is not on the programming device.

This means that the program sections listed below, that cannot be downloaded to the CPU, are not available. These components are:

- The symbol table with the symbolic names of the addresses and the comments
- Network comments of a Ladder Logic or Function Block Diagram program
- Line comments of a Statement List program
- Userdefined data types

13.3.3.2 Editing Uploaded Blocks if the User Program is on the PG/PC

To edit blocks from the CPU, proceed as follows:

1. Open the online window of the project in the SIMATIC Manager.
2. Select a "Blocks" folder in the online window. The list of loaded blocks is displayed.
3. Now select the blocks, open and edit them.
4. Select the menu command **File > Save** to save the change offline on the programming device.
5. Select the menu command **PLC > Download** to download the changed blocks to the programmable controller.

13.3.3.3 Editing Uploaded Blocks if the User Program is Not on the PG/PC

To edit blocks from the CPU, proceed as follows:

1. In the SIMATIC Manager, click the "Accessible Nodes" toolbar button or select the menu command **PLC > Display Accessible Nodes**.
2. Select the node ("MPI=..." object) from the list displayed and open the "Blocks" folder to display the blocks.
3. You can now open blocks and edit, monitor, or copy them as required.
4. Select the menu command **File > Save As** and enter the path for the programming device where you want to store the blocks in the dialog box.
5. Select the menu command **PLC > Download** to download the changed blocks to the programmable controller.

13.4 Deleting on the Programmable Controller

13.4.1 Erasing the Load/Work Memory and Resetting the CPU

Before downloading your user program to the S7 programmable controller, you should perform a memory reset on the CPU to ensure that no "old" blocks are still on the CPU.

Requirement for Memory Reset

The CPU must be in STOP mode to perform a memory reset (mode selector set to STOP, or to RUN-P and change the mode to STOP using the menu command **PLC > Diagnostics/Settings > Operating Mode**).

Performing a Memory Reset on an S7 CPU

When a memory reset is performed on an S7 CPU, the following happens:

- The CPU is reset.
- All user data are deleted (blocks and system data blocks (SDB) with the exception of the MPI parameters).
- The CPU interrupts all existing connections.
- If data are present on an EPROM (memory card or integrated EPROM), the CPU copies the EPROM contents back to the RAM area of the memory following the memory reset.

The contents of the diagnostic buffer and the MPI parameters are retained.

Performing a Memory Reset on M7 CPUs/FMs

When a memory reset is performed on an M7 CPU/FM, the following happens:

- The original state is restored.
- The system data blocks (SDB) with the exception of the MPI parameters are deleted.
- The CPU/FM breaks off all existing connections. User programs are retained and will continue running after you switch the CPU from STOP to RUN.

With the "memory reset" function you can restore the original state of the M7 CPU or FM following serious errors by deleting the current system data blocks (SDB) from the work memory and reloading the SDBs in the read-only memory. In some cases, a warm restart of the operating system will be required. To do this, you clear the M7 using the mode selector (switch to MRES position). A reset using the mode selector on SIMATIC M7 CPUs or FMs is only possible if the RMOS32 operating system is used on the CPU/FM.

13.4.2 Deleting S7 Blocks on the Programmable Controller

Deleting individual blocks on the CPU may be necessary during the test phase of the CPU program. Blocks are stored in the user memory of the CPU either in the EPROM or RAM (depending on the CPU and the load procedure).

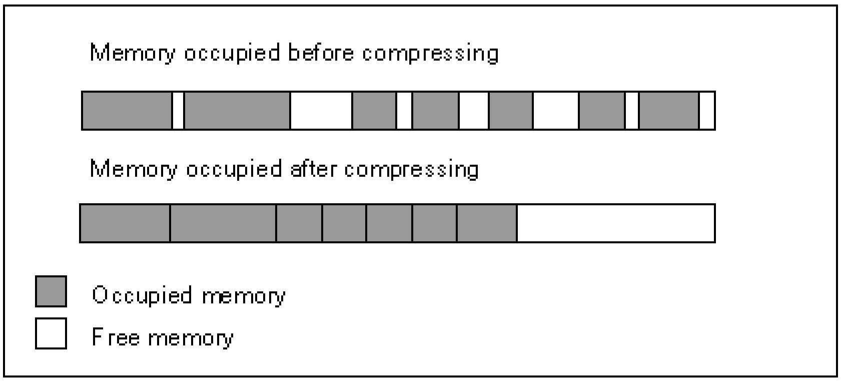
- Blocks in the RAM can be deleted directly. The occupied space in the load or work memory becomes free and can be used again.
- Blocks in the integrated EPROM are always copied to the RAM area following a memory reset of the CPU. The copies in the RAM can be deleted directly. The deleted blocks are then marked in the EPROM as invalid until the next memory reset or power down without RAM backup. Following a memory reset or power down without RAM backup, the "deleted" blocks are copied from the EPROM to the RAM and become active. Blocks in the integrated EPROM (for example, in the CPU 312) are deleted by overwriting them with the new RAM contents.
- EPROM memory cards must be erased in the programming device.

13.5 Compressing the User Memory (RAM)

13.5.1 Gaps in the User Memory (RAM)

After deleting and reloading blocks, gaps can occur in the user memory (load and work memory) and reduce the usable memory area. With the compress function, the existing blocks are rearranged in the user memory without gaps, and a continuous free memory is created.

The following figure shows a diagram of how occupied blocks of memory are shifted together by the compress function.


Memory occupied before compressing
Memory occupied after compressing
Occupied memory
Free memory

Always Try to Compress the Memory in STOP Mode

Only if you compress the memory in "STOP" mode are all the gaps closed up. In the RUNP mode (mode selector setting), the blocks currently being processed cannot be shifted since they are open. The compress function does not work in the RUN mode (mode selector setting) (write protection!).

13.5.2 Compressing the Memory Contents of an S7 CPU

Ways of Compressing the Memory

There are two methods of compressing the user memory, as follows:

- If there is insufficient memory available when you are downloading to the programmable controller, a dialog box appears informing you of the error. You can compress the memory by clicking the corresponding button in the dialog box.
- As a preventative measure, you can display the memory utilization (menu command **PLC > Diagnostics/Setting > Module Information**, "Memory" tab) and start the compressing function if required.

Procedure

1. Select the S7 program in the "Accessible Nodes" window or the online view of the project.
2. Select the menu command **PLC > Diagnostics/Setting > Module Information**.
3. In the dialog box which then appears, select the "Memory" tab. In this tabbed page there is a button for compressing the memory if the CPU supports this function.

14 More than One User Editing Projects

14.1 Multi-user Configuration in a Network

Overview

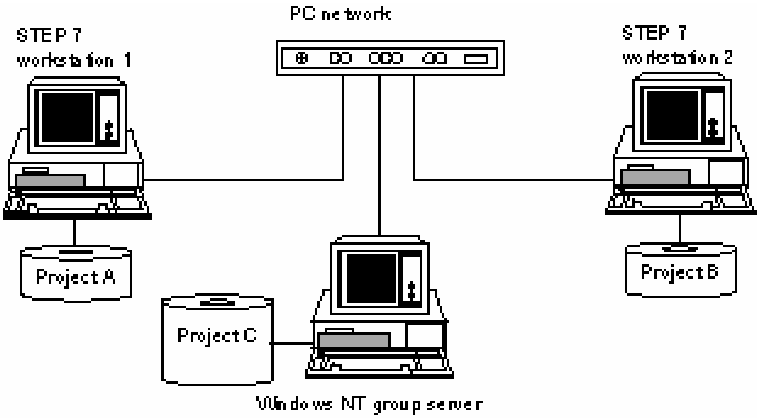
With the operating systems released for STEP 7, you can work in a multi-user configuration. You can either work with the multiproject or consider one of the following configurations:

- The project is on a local drive and is also used from another workstation.

  Example: Workstations 1 and 2 access project A on workstation 1.
- The project is on a network server.

  Example: Workstations 1 and 2 access project C on the network server.
- The projects are distributed among the local drives and one or more network servers.

  Example: Workstations 1 and 2 access projects A, B, and C.



Guidelines for Storing Projects on Network Servers

- As of STEP 7 V5.2, when projects stored on other computers in the network are opened, they will always be included in the local project management with their UNC path (= Universal Naming Convention, in other words, \\<Servername>\<Share>\...). This means that while it remains possible to open a project on a connected network drive, the storage location shown in STEP 7 will always be the UNC path (for example in the SIMATIC Manager with **File > Open/Manage**). In particular, this means that projects can be opened in the network directly via their UNC path and not as previously only via connected network drives.
  Local projects, on the other hand, are always included in project management with their physical path (in other words, <local drive letter>:\...) even if they are located on a local share.
  If you install STEP 7 V 5.4 incl. SP4 over a previous version, the project management will be updated by the setup.
- When you store your projects on network servers or on enabled drives of other network users, the Windows operating system can only be shut down on those servers or network nodes when all STEP 7 applications that access the projects are closed.
- If you store your projects on network servers or on shared drives of other network stations, ensure that STEP 7 is installed on the network server or the computer of the network station.

Guidelines for Multi-User Editing of S7 Programs

You should note the following:

- Before more than one user can work on one S7 program, you must set the workstation configuration (menu command **Start > Simatic > STEP 7 > Configure SIMATIC Workspace**). Note the corresponding Help on the dialog box.
- Blocks and STL source files:
  Each user should program a different block or source file. If two users attempt to edit a block or source file at the same time, a message is displayed, and access is denied for the second user.
- Symbol table:
  Several users can open the symbol table at the same time but only one user can edit it. If two users attempt to edit the symbol table at the same time, a message is displayed, and access is denied for the second user.
- Variable tables:
  Several users can open the variable table at the same time but only one user can edit it. If two users attempt to edit the variable table at the same time, a message is displayed, and access is denied for the second user. There can be a number of variable tables in an S7 program. These can, of course, be edited separately and independently of each other.

Guidelines for Multi-User Editing of Stations

You should note the following:

- The hardware configuration and the network configuration of a station should only be edited centrally by one user.

14.1.1 Setting the Workstation Configuration

To work on a project from different STEP 7 workstations, you must make the following settings at each workstation.

1. In the Windows start bar, select the menu command **Start > Simatic > STEP 7 > Configure SIMATIC Workstation**.
2. Select the option "Multi-terminal system" and the network protocol you want to use.

14.2 Single-Node Configuration for Non-Networked Workstations

14.2.1 Merging Several S7 Programs into One

STEP 7 does not offer support with merging S7 programs on non-networked workstations. The only way to merge S7 programs in this case is to copy individual blocks or source files. Shared data for the project such as the symbol table or the variable table must be edited manually after copying.

1. Copy blocks and source files to their respective folders in an S7 program.
2. Export the symbol tables of the individual S7 programs to ASCII format and import them into the symbol table of the merged S7 program.
3. Check whether symbols have been used more than once.

   Tip: You can also integrate short symbol tables using the clipboard (copy and paste).
4. Copy the variable tables you want to use or integrate the various variable tables using the clipboard (copy and paste) into a new variable table.

14.2.2 Copying S7 Programs with Message Attributes

If you have assigned message attributes to blocks, note the following restrictions when you copy S7 programs:

Assigning Project-Oriented Message Numbers

Message numbers could overlap. To avoid conflicts:

- Use the menu command **Edit > Special Object Properties > Message Numbers** to allocate a fixed message number range to each S7 program.
- When you copy S7 programs, make sure they do not overwrite other S7 programs.
- Note that only message types (FBs) can be programmed separately from the S7 program.

Assigning CPU-Oriented Message Numbers

- Programs can be copied within a project to interprojects without a change of message number.
- The message number will change when you copy single blocks. Here you need to recompile the block in order to implement the changed message number in the program.

Copying a Program with Project-Oriented Message Number Assignment into a Project with CPU-Oriented Message Number Assignment

- To copy a program whose message numbers were assigned project oriented to another project that contains a program whose message numbers were assigned CPU oriented, select the required program, call menu command **File > Save As...** and enable the "Reorganize" check box in the pop-up dialog box.
- Message attributes are assigned by default when copying. Inconsistent assignments are displayed in a pop-up dialog box. In this box you can choose your assignment.

Copying a Program with CPU-Oriented Message Number Assignment into a Project with Project-Oriented Message Number Assignment

You can only copy single FBs with messages.

Caution

The assignment of message numbers in the program of a project must be uniform!

When you copy a message-type block that contains a cross-reference to a text library to another program, you must include the corresponding text libraries or create another text library of the same name, or modify the cross-reference in the message text.

15 Working With Projects in the Multiproject

15.1 What You Should Know About Multiprojects

What is a multiproject?

The term multiproject refers to the folder containing all projects and libraries of an automation solution with one or several STEP 7 projects as well as the (optional) libraries. The projects of a multiproject can contain objects linked to other objects (e.g. S7 cross-project connections).

What are the benefits of a multiproject?

Projects as part of a multiproject can be split into smaller objects that are much easier to survey.

You can also use the multiproject, for example, to create projects for distributed authoring and assign a station to each author for offline editing of the project.

Functions linked to other projects (cross-project functions) allow you to handle a multiproject almost in the same way as a single project.

Examples of cross-project functions

STEP 7 cross-project functions (standard software package):

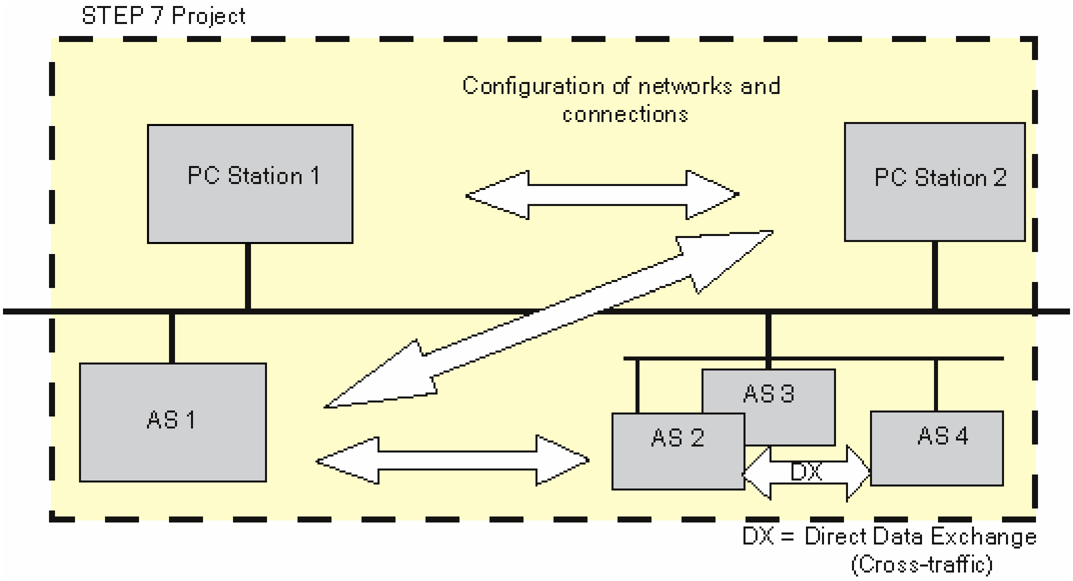
- Save As (storing a multiproject with all projects at other locations)
- Archiving (multiproject with all projects)
- Adjustment of the cross-project network (e.g. merging subnets)
- Update of views (all projects of a multiproject)

Further cross-project functions (with PCS 7, BATCH flexible)

- Transfer of AS-OS connection data
- Exporting BATCH flexible system data

Project structure without multiproject

In a usual project structure without multiproject all objects must be available in a central project, for example, to allow the transfer of AS-OS connection data.

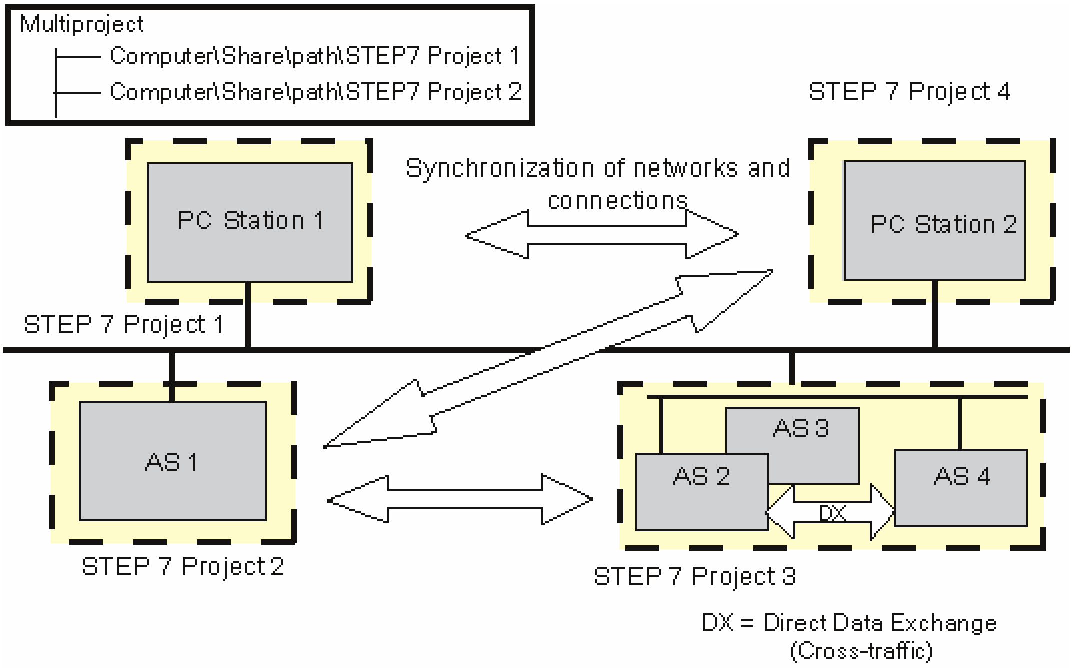

STEP 7 Project
Configuration of networks and connections
PC Station 1
PC Station 2
AS 1
AS 3
AS 2
AS 4
DX
DX = Direct Data Exchange (Cross-traffic)

Project structure with multiproject

STEP 7 projects are edited at distributed locations and can therefore be shared by several authors.

Cross-project functions allow the user to split the projects of a multiproject into smaller objects that are much easier to survey.

You can configure one or several automation systems for a project.


Multiproject
Computer\Share\path\STEP7 Project 1
Computer\Share\path\STEP7 Project 2
STEP 7 Project 4
Synchronization of networks and connections
PC Station 1
PC Station 2
STEP 7 Project 1
AS 1
AS 3
AS 2
AS 4
DX
STEP 7 Project 2
STEP 7 Project 3
DX = Direct Data Exchange (Cross-traffic)

Access protection

- To use the "Adjust in Multiproject" function, you must be authenticated as project administrator in the SIMATIC Logon Admin Tool.
- When working on a multiproject, you can only open those projects to which you have been allowed access as project editor or project administrator. With all other projects, the SIMATIC Manager displays the icon and text for the project in light gray.
- To open an access-protected project or library, you must be authenticated as a project editor or administrator in the SIMATIC Logon administration tool or know the project password.
- In multiprojects with access protection, the "Compile and Download Objects" function is possible only when the logged on user is entered with user or administrator rights in all projects.
- The "Adjust in Multiproject" function is effective only in projects in which no access protection has yet to be enabled and in projects in which the user is authenticated as project administrator.
- If one or more projects in a multiproject have access protection, "Save As" is only available to users entered with administrator rights in all access-protected projects.

15.2 Multiproject - Requirements and Recommendations

Different methods of working with the multiproject

The multiproject lets you configure a flexible system. The main feature is time-saving, parallel creation of projects that can be merged by the system.

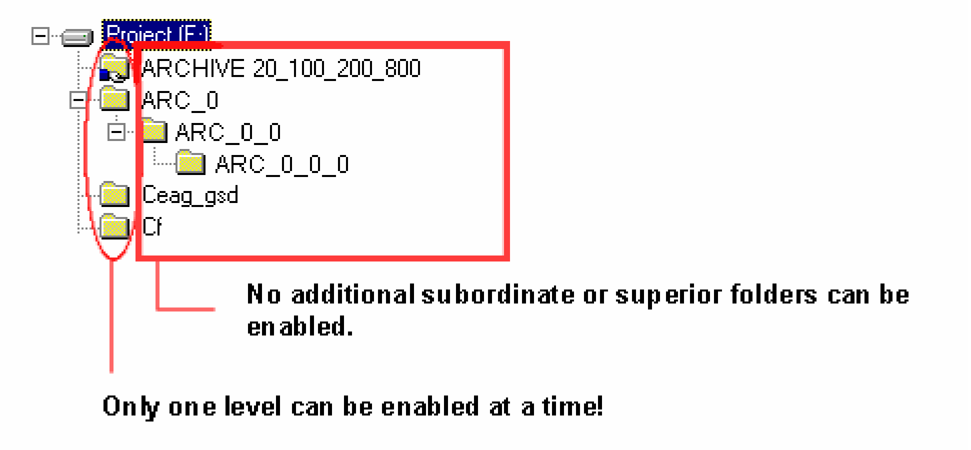
Basically we can differentiate between two working methods:

- Several members of a user group in a network environment share a multiproject. The projects of the multiproject are stored in different network folders.
  - In this case, all communication partners are available online for the configuration of connections, for example.
  - Only the execution of cross-project functions on the networks must be agreed upon between the partners, because read/write access to the projects is not permitted while these functions are active.
- The project administrator manages the multiproject at the master database, defines the structure of projects (locally, if required), controls distributed transactions for external authors, then transfers the incoming projects to the multiproject, adjusts the cross-project data via the system functions (file update synchronization) and executes necessary cross-project functions.
  - In this case, for example, you must declare connection names, because during the adjustment of projects the S7 connections could easily be merged due to identical names.
  - The user can also combine those two working methods. STEP 7 supports the distribution of projects across the network as well as the simple insertion and removal of projects of the multiproject, regardless of the method used.

Basic requirements

The following is required for the distribution of projects to different network folders:

- The projects are stored in folders with read/write attribute. This implies in particular:
  - Read/write access to drives storing the multiproject or projects must have been granted prior to the creation of the multiproject.
  - Release names must be unique within the network. We recommend you choose a release name that consists of the computer name and the drive name (e.g. PC52_D).
  - Access rights and release names must not be modified for any resources (folders) participating in the multiproject. The reason is: When a project is inserted into the multiproject, STEP 7 will generate a reference to the location of this project. This reference is determined by access rights and the release name of the corresponding resources.
  - A project can only be found under the release name it was assigned when it was implemented in the multiproject.
  - Access to the complete drive hierarchy must be denied. Access rights to a folder may only be granted at one hierarchy level.



No additional subordinate or superior folders can be enabled.

Only one level can be enabled at a time!

- STEP 7 or PCS 7 must be installed on computers that contain the project folders. The reason being: STEP 7 or PCS 7 provides the required database functions on the server for access to the projects.
- When you implement projects for which you have configured messages into a multiproject, note the following:
  - When assigning message numbers for the project (project-oriented message numbers), the message number ranges for the CPUs must not overlap. If you insert several subprojects with project-oriented message numbers into a multiproject, no automatic redundancy check of message numbers is carried out. You yourself have to check to make sure that message numbers only occur once.

Requirements for cross-project functions

When initiating cross-project functions within such a constellation, you must ensure that:

- All computers containing projects or the multiproject must be available online on the network as long as any projects are being processed.
- Processing of projects is not allowed while these cross-project functions are being executed.

If this cannot be ensured, we recommend that you transfer all projects to a PG/PC and then execute the cross-project functions locally.

Multiple users editing to the same object:

A station should always be edited only by one author at a given time (not only in the multiproject).

When you split the projects into projects containing only one station, you can be sure that only one author is accessing a project at a given time.

Recommendations and rules for project size

Generally speaking, there is really no way to say how many stations a project should have. Nonetheless, here are a few points that may make it easier for you to make a decision:

- The more complex the station, the less stations the project should have.
- The number of team members available for configuring the system can also be a criterion for specifying your distributed transactions. Each author could work in parallel on a project of the multiproject, independently of others. You could then optimize the size of each project to meet all deadlines for the completion of all projects.
- Stations interconnected via Direct Data Exchange ("cross-traffic") **must be located in the same project**. Note that you cannot configure Direct Data Exchange across project boundaries.
- Stations that are operating in global data communication via an MPI subnet also **must be located in the same project**.
- Stations you want to include in your network visualization (NetPro) must also be located in the same project.
- Important for commissioning is: Download functions available under NetPro have no cross-project effect, but rather take effect only within the project. Affected are the functions:
  - **Download in the Current Project > Selected and Partner Stations**
  - **Download in the Current Project > Stations on the Subnet**
  - **Download in the Current Project > Selected Connections**
- Compile and save operations are also restricted to the currently active project.

If an S7 connection is a cross-project connection, for example, the network configuration of both participating projects must be compiled.

15.3 Handling Multiprojects

Creating a new multiproject

1. Select menu command **File > New**.
2. In the "Name" box of the "New" dialog box, assign a name to the multiproject and select a "Multiproject" type.
   Click on "Find" or edit the path to determine a storage location for the multiproject.

   Example:

   Storage location: \\S_S712\Plant_1

3. Confirm the dialog with "OK".

Notes on path definitions

Wherever possible you should use UNC conventions to specify your paths. This allows you flexible and future-proof handling of STEP 7 projects.

Example:

UNC path: **\\Computer\Share\Path**

\\	double backslash (<Alt> <9><2>)
Computer	Server name: max. string length = 15 characters Name of the computer that supplies the resource (File, directory, ..).
Share	Release name: Under Windows, access to resources of other Windows computers is only possible if corresponding rights have been granted.
Path	This is an optional setting.

Creating a new project for the multiproject

While you create a new project, you can also assign a new project to the current multiproject in the same step.

The multiproject to be inserted in the new project must already be open.

1. Select menu command **File > New**.
2. In the "Name" box of the "New" dialog box, assign a name to the project and select the type "Project".
   Activate the "Insert project in current Multiproject" check box.
   Click the "Browse" button to determine a storage location for the project. If you are operating in a computer network, you can store the project file on a network drive to provide access to other team members.
3. Confirm the dialog with "OK".

As an alternative, you can also use menu command **File > Multiproject > Create in Multiproject** to generate a multiproject in the current multiproject.

Separating a project from a multiproject or removing a project for editing

For distributed authoring (for example, by external authors) you can also remove projects from the multiproject and reinsert them after they have been edited.

When you remove the project, the dependencies to other projects will be maintained (e.g. cross-project connections) and the projects will still be available for compilation.

1. Select the project(s) you want to remove from the multiproject.
2. Select menu command **File > Multiproject > Remove From Multiproject** or the menu command **File > Multiproject > Remove for Editing**

You can now save the project to a storage medium with "Save As" and distribute it to external authors.

Tip

Another method of preparing projects for external editing:

Create a copy of the project for the external authors with **File > Save As** and keep this "unprocessed" copy in the multiproject, for example, so that you will still be able to debug the cross-project functions.

Then, after the external project has been completed, you can call menu command **File > Add to Multiproject** and replace the original with the edited project.

In this case, you must ensure that only the project copy is edited, rather than both the copy and the original project!

Inserting projects into a multiproject or returning projects to a multiproject after editing

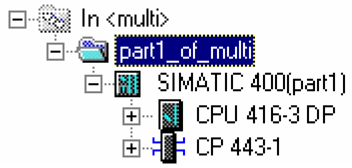
Any existing project can be newly inserted into or returned to a multiproject. For this procedure it is irrelevant whether the object in question had been moved previously or whether a "third-party" or new project is added.

1. Open a multiproject.
2. Select **File > Multiproject > Insert into Multiproject** menu command or the **File > Multiproject > Reapply After Editing** menu command.
3. In the next dialog, select the project you want to add.
4. Confirm the dialog with "OK".

Now you can call the "Adjusting Projects in a Multiproject" wizard to merge cross-project subnets and to adjust the connection configurations.

Note

SIMATIC Manager identifies an open project that is part of a multiprojects with a grayed out multiproject icon:



Adjusting projects in a multiproject

The "Adjusting Projects in a Multiproject" wizard merges the subnets and adjusts cross-project connection data.

Start this wizard in SIMATIC Manager under menu command **File > Multiproject > Adjust Projects**.

The adjustment is carried out automatically (as far as possible) and step by step. The wizard logs all inconsistencies. Open NetPro to debug errors that might have occurred.

Copying the multiproject ("Save As")

Regardless whether or not the projects of a multiprojects are distributed to network locations: Whenever you copy the multiproject (use menu command **File > Save As** to save the multiproject), all objects of the multiproject, that is, the actual multiproject and all its projects will be saved to this location. It is here not possible to distribute the projects to several different target folders!

Copying a project in a multiproject ("Save As")

Depending on the current arrangement in the SIMATIC Manager, you have different options available for copying a project that is part of a multiproject:

Case 1: The multiproject is open and a project in the multiproject is selected:

1. Select the menu command **File > Save As**.
2. In the dialog box that appears, select from the following options:
   - "Add to multiproject":
     By default, the copy is inserted in the current multiproject. You can also insert the project in another multiproject selected from the list.
     If this option is not selected, then the copy is saved as a "normal" project with no relationship to a multiproject. The original project remains in the multiproject.
   - "Replace current project":
     The copy is inserted in the current multiproject. The original project is removed from the multiproject and remains in STEP 7 database as a "normal" project with no relationship to a multiproject.
3. Exit the dialog with "OK".

Case 2: The project in a multiproject is open:

1. Select the menu command **File > Save As**.
2. In the dialog box that appears, select from the following options:
   - "Add to multiproject":
     By default, the copy is inserted in the first multiproject in the list of multiprojects. You can also insert the project in another multiproject selected from the list.
     If this option is not selected, then the copy is saved as a "normal" project with no relationship to a multiproject. The original project remains in the multiproject.
     The option "Replace current project" is not available.
3. Exit the dialog with "OK".

Copying projects of a multiproject to an MMC

You can archive the multiproject with all its projects (menu command **File > Archive**) and then save these data to a Micro Memory Card (MMC) that is equipped with sufficient memory space.

Multiproject data you have not archived cannot be written to a single Micro Memory Card (MMC).

However, you have the option of "spanning" all projects of the multiproject across multiple MMCs. When you save the data to MMC, the multiproject component is automatically included, so that the multiproject structure can be "restored" completely from these volumes at a later time.

We recommend you create a text file to log the multiproject components and their respective storage locations (e.g. CPU IDs), and to include this log file in your data transfer to the MMCs. When service tasks are to be carried out (PG without project information), this procedure saves you a tiresome search to find relevant CPUs that have saved the multiproject objects to the MMC.

How to move projects from one multiproject to another multiproject

You can move a project that is part of a multiproject to another multiproject. To do this, proceed as follows:

1. Open the multiproject to which you want to move the project
2. Select menu command **File > Multiproject > Add to Multiproject**
3. In the dialog box that appears, select the project to be moved.
4. A message is displayed asking you if the project should be added to the current multiproject Select "Yes" to move the project to the current multiproject.

Moving stations within a multiproject

You can move stations (e.g. S7-400 or SIMATIC PC stations) to other locations in the multiproject.

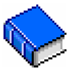
When a station is moved between projects of a multiproject (for example by dragging them with the mouse), any configured connections will be broken. You can re-establish these broken connections in NetPro (menu command **Edit > Merge connections...**).

Setting a Library as Master Data Library

You cannot only add projects to a multiproject, but also implement libraries, one of which you can identify as master data library. In this master data library you can store blocks used globally for all projects. This function is especially useful for PCS 7 engineering, for example.

The master data library may contain only one program of each type (S7, M7, etc.)

1. Open the library you want to identify as master data library.
2. Select menu command **File > Multiproject > Set as Master Data Library**.
   This is the icon for master data library:



15.4 Online Access to PLCs in a Multiproject

Cross-project access with an assigned PG/PC

The "Assign PG/PC" function for the objects "PG/PC" and "SIMATIC PC Station" are also available for the multiproject.

You can specify the target module for online access in any project of the multiproject. This procedure is the same as if you were working with one project only.

Requirements

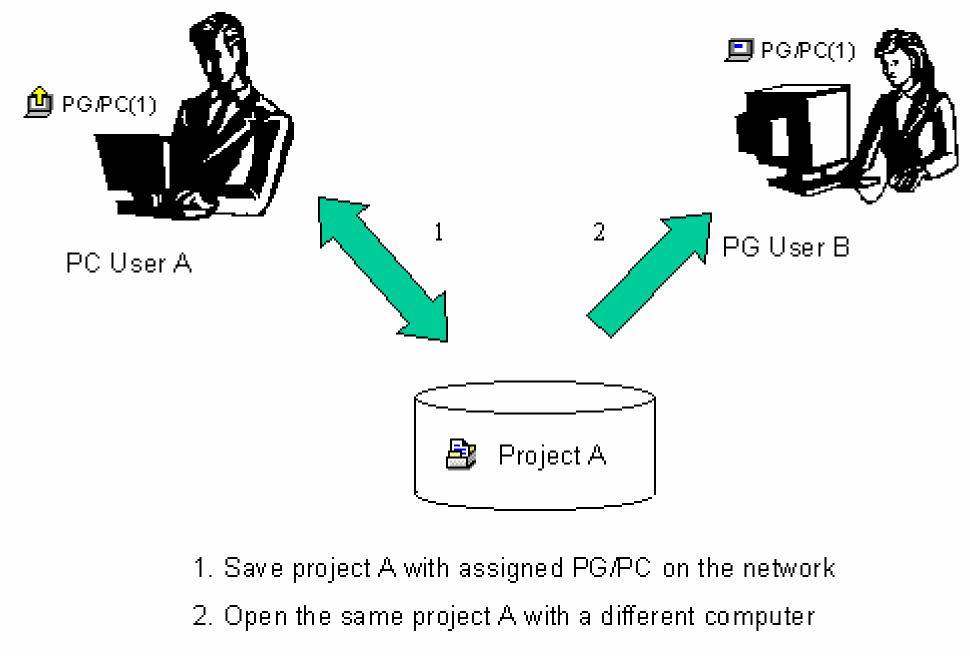
- PGs/PCs or PC stations you want to use for online access to PLCs must have been assigned in any one project of the multiproject.
  Note: The assigned PG/PC or PC station is highlighted in yellow color when the corresponding project is open.
  The PG/PC assignment is only visible if the PG that opens the project is properly assigned.
- The cross-project subnets are merged.
- All projects of the multiproject have been compiled and configuration data have been downloaded to the participating stations; for example, to provide routing information to all participating modules for establishing connections between the PG/PC and the target module.
- The target modules can be accessed across the networks.

Possible problems when working with distributed projects

The PG/PC assignment is not visible if the allocation of projects is changed and a project is opened on a PG/PC on which it was not created.

Nonetheless, the configured PG/PC object still maintains the "assigned" status - but with the "wrong" PG/PC.

In this case you must clear the existing assignment and then reassign the PG/PC object. Online access to modules available within the multiproject is then possible without any problem.


PG/PC(1)
PC User A
PG/PC(1)
PG User B
1
2
Project A
1. Save project A with assigned PG/PC on the network
2. Open the same project A with a different computer

Tip for working with distributed projects

If more than one of the team members wants to access the PLCs online on their PG, it would be useful to create one "PG/PC" or "SIMATIC PC station" object in the multiproject and then set up an assignment for each one of the PGs.

Depending on which PG has opened the project, SIMATIC Manager indicates only the object assigned to this PG with a yellow arrow.

15.5 Creating Cross-Project Subnets

Concept of cross-project subnets

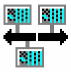
Under STEP 7 V5.2 you can configure cross-project subnets that you can use to configure connections.

Subnets "that run across several projects" (that is, are cross-project subnets) are not created in a single operation. Rather, already configured subnets in the different projects of the multiproject are **merged**!

Individual subnets of a merged subnet will be maintained as usual. When grouped, they are assigned to a logical "Subnet Group" that represents the properties that are common to all assigned subnets.

Merged subnets (that is, cross-project subnets) are of the same subnet type, with identical S7 subnet IDs. They are identified under NetPro by their name suffix "(cross-project)".

In the SIMATIC Manager merged subnets are represented by the following special icons.

Small icon	
Large icon	

Subnets that can be merged

Industrial Ethernet, PROFIBUS and MPI subnets can be merged.

Subnets that cannot be merged

- PtP subnets **cannot** be merged.
- Constant-cycle PROFIBUS subnets also cannot be merged. Reason: Constant bus cycle time can only be configured for mono-master systems, which means that cross-project communications connections do not make sense in this configuration.
- For this same reason, PROFIBUS subnets that have H stations connected to them as DP masters also cannot be merged.
- If a subnet with a configured sync domain is located in an Ethernet subnet group, no other Ethernet subnet with a sync domain may be added to this group.

Requirements

To merge subnets, write access must have been granted to the participating projects and their subnets.

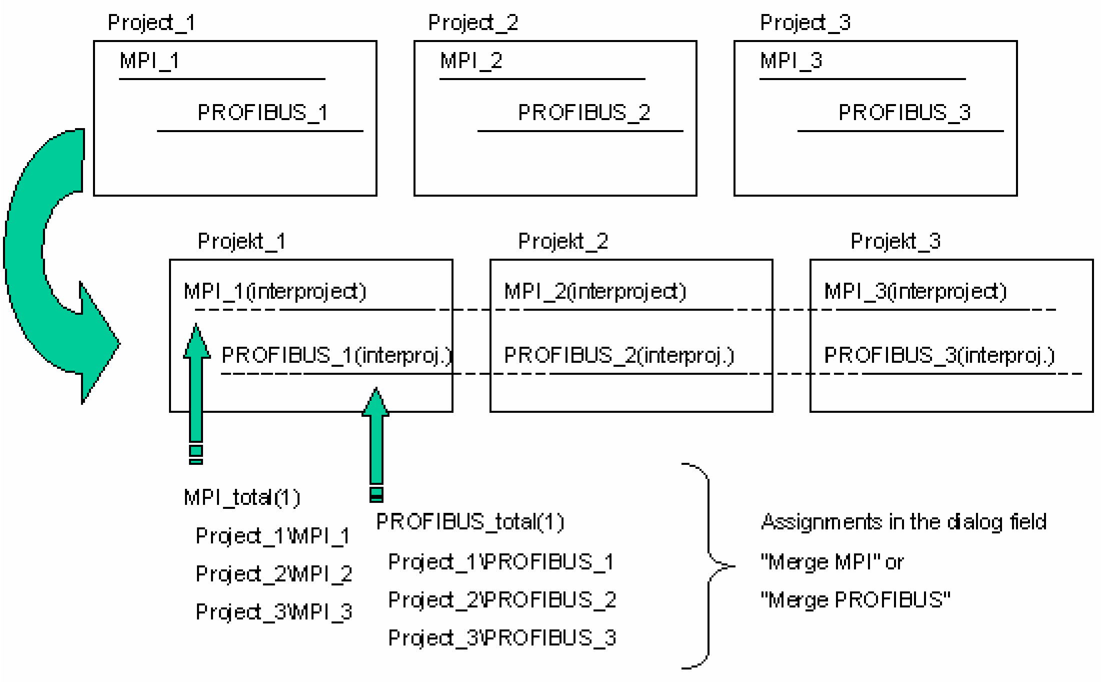
How to merge subnets in NetPro

To merge or unmerge subnets, proceed as follows:

1. Select the menu command **Edit > Merge / Unmerge Subnets > ...**
   Select the type of subnet in the last part of this menu command.
2. In the next dialog, assign the multiproject subnets to a merged subnet.
   - In the right-hand field ("Merged"), select a whole subnet.
   - In the left-hand field ("Subnets in the multiproject"), select subnet that is to be made part of the merged subnet.
   - Click the button with the arrow pointing to the right.

   Subnets are grouped so that you can maintain an overview of the subnets you are going to assign common properties (e.g. an identical S7 subnet ID).
   All subnets are merged to form a "Merged Subnet" (Default name, for example: PROFIBUS_merged(1)). You can change the default name of the merged subnet.
   The properties of the first subnet you add to the subnet group determine the properties of all subnets you subsequently add to the group. You can later specify another "leading" subnet, that is, the one that determines the properties of all others via the "Select" button. The icon for the leading subnet has green border to clearly identify it.
3. If you need more merged subnets, click the "New" button and assign subnets from the field ("Subnets in the multiproject") to this merged subnet, as described above.
4. If required, you can change the subnet properties via the "Properties" button. This may be necessary, for example, for changing the bus parameters of PROFIBUS subnets you are going to merge.
5. Confirm the settings with "OK" or "Apply".
   The transferable subnet properties of a leading subnet are then applied to all other subnets of the group.

The picture below highlights the context:



How to unmerge subnets

To unmerge subnets, proceed as follows:

1. Select the menu command **Edit > Merge / Unmerge Subnets > ...**
   Select the type of subnet in the last part of this menu command.
2. In the next dialog, select a subnet in the right-hand field ("Merged") that is to be removed from the merged subnet.
3. Click the button with the arrow pointing to the left.
   The selected subnet now appears in the left-hand field ("Subnets in the multiproject").

Properties of merged subnets

When you merge subnets, the transferable properties of the master subnet are transferred to the other subnets of the same group.

In addition to the S7 subnet ID, these properties are (for example, with PROFIBUS subnets) the profile, transmission speed, highest address, and the number of nodes to be taken into account.

The following subnet parameters will **not** be adapted - which means that the individual subnets of a group will maintain their own subnet properties:

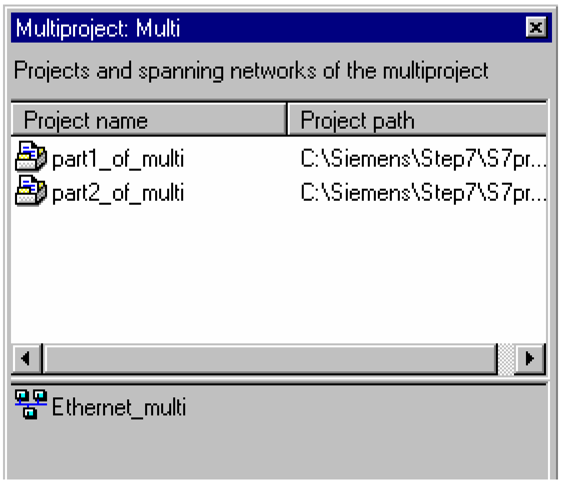
- Name
- Author
- Comment

15.6 Display Of Merged Subnets In The Network View

The "Multiproject" Window

NetPro displays the "Multiproject" window when you open a project that is part of the multiproject.

Use menu command **View > Multiproject** to hide or show this window. You can also control the view with an icon.



Multiproject: Multi

Projects and spanning networks of the multiproject

Project name	Project path
part1_of_multi	C:\Siemens\Step7\S7pr...
part2_of_multi	C:\Siemens\Step7\S7pr...

Ethernet_multi

The upper section of the window displays the projects of the multiproject. The lower section displays all cross-project subnets of the multiproject under their master subnet name.

After you have selected the projects in the upper section of the window (multiple selection is also possible), the lower section of this window displays exactly those cross-project subnets that are routed through all the selected projects.

Tip: You can quickly navigate 'through the projects' of a multiprojects by double-clicking on the corresponding project in the "Multiproject" window. STEP 7 will then open the network view of the corresponding project in a separate window.

Network View

The network view displays all cross-project subnets with the name suffix "...(cross-project)".

15.7 Multiproject-Oriented Network View

The multiproject-oriented network view provides an overview of all stations in a multiproject with the connections to their respective subnets.

Cross-project subnets are only shown in this view in context (i.e. like a subnet). The name of the displayed cross-project subnet is the name of the subnet group.

You can print out the multiproject-oriented network view, and you can download to all stations from this view. However, you cannot edit or save projects in this view.

Requirements

- This function can only be started from a project that is part of a multiproject.
- All changes made to projects in the multiproject must be saved. If necessary, you will be prompted to do so.
- All projects in the multiproject must be accessible (this is relevant for distributed projects).
- Projects in a multiproject must not be write-protected (read-only).
- No connection status must be activated in an opened project (from which multiproject-oriented network view is supposed to be started).

Procedure

1. In the network view, select a project with the **View > Multiproject-Oriented Network View** menu command.
   After this, the menu command is enabled with a check mark.
2. Position the objects as you like. You can also print out the multiproject-oriented network view.
3. From this view, you can download to all stations in the multiproject (command: **PLC > Download ...**)
4. To return to the "normal" network view of the project, select the **View > Multiproject-Oriented Network View** menu command once again.

When you close the project, all positions of the object are automatically saved as well.

Properties of the multiproject-oriented network view

- After this function is called for the first time from a project, all objects are rearranged. You can change the positions of the objects. If you deactivate the multiproject-oriented network view and then call it again from another project in the multiproject, the objects will again be shown in the same arrangement.
- Features of the multiproject-oriented network view such has the zoom factor, the subnet lengths (reduced or not reduced), with or without DP slaves/IO devices, etc. are all determined by the project from which the multiproject-oriented network view has been called.
- When the multiproject-oriented network view is active, you cannot make any changes to the projects. However, you can trigger a cross-project consistency test. For this reason, you **cannot** use the **Save** command (**Network > Save**) in the multiproject-oriented network view. If you do select the **Network > Save Compile** menu command, the network configuration **will only be compiled**.
- In the multiproject-oriented network view, a configuration can be downloaded to all stations in the multiproject.
- Only one multiproject-oriented network view can be open at one time. If another STEP 7 user attempts to start the multiproject-oriented network view from another project, this action will be canceled and an error message will be displayed.

15.8 Configuring Cross-Project Connections

Introduction

After you have configured cross-project subnets, STEP 7 also allows you to configure connections with such master subnets. The end points of these connections may be located in different projects.

STEP 7 offers support for the **creation** of cross-project connections within the multiproject as well as for the **adjustment** of connections configured with no multiproject context.

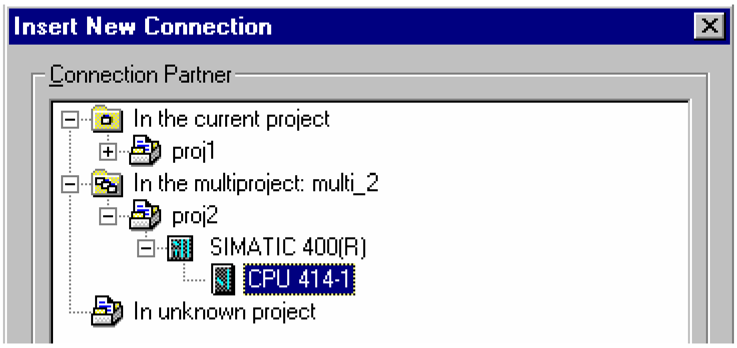
Connection types of cross-project connections

Under STEP 7 you can create cross-project connections for redundant and S7 connections.

Under SIMATIC NET you can also configure further cross-project connection types - with the exception of FMS connections. For further information, refer to the SIMATIC NET documentation.

Cross-project connections to a specified partner

Cross-project connections to a specified partner (e.g. a CPU) are created same as internal connections of a project (identical procedure). The dialog for selecting the connection partner has been expanded and now lets you select - apart from selecting the end point (e.g. a module) - a project in the multiproject in which the end point is located.



The condition is, that the projects are part of a multiproject and that the subnets have been merged (e.g. by using the "Adjusting Projects in a Multiproject" wizard of SIMATIC Manager).

Properties of cross-project connections

Consistency of cross-project connections is maintained when projects of a multiproject are handled. Cross-project connections within a multiproject stay fully operational a can be compiled, even if the project that contains the connection partner is removed from the multiproject.

The following applies to STEP 7 connections: STEP 7 prompts you to split the connection before the properties dialog pops up only when you call this dialog to view the connection properties. You can only modify the connection properties after you have confirmed this prompt with "Yes". You must adjust the connection properties manually when you change the properties, same as with S7 connections to an unspecified connection partner. Whenever you change connection properties, there is a risk of malfunctions during runtime.

Only the local ID for a connection can be changed without terminating the connection.

Fault-tolerant STEP 7 connections cannot be terminated.

Tip

STEP 7 uses the S7 subnet ID - an object property of a subnet - to adjust cross-project connections.

When you attempt to configure cross-project connections without having merged both of the corresponding subnets (which are to form the connection path), for example because other, similarly affected projects are temporarily unavailable, you can help yourself in the following way:

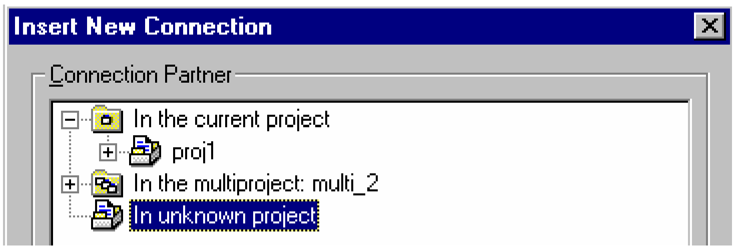
Adjust the S7 subnet ID at all subnets. Select the respective subnet and view its object properties via the context-sensitive menu. Enter **identical** S7 subnet IDs for both of the subnets.

You can then specify that the nodes on "manually adjusted" subnets of the other project form the connection end point. However, this process still does not confirm the uniqueness of individual subnets from the viewpoint of NetPro. The consistency test would report an error if the projects are connected through a common, cross-project subnet. In this case, the same subnet IDs in different projects will be detected and recognized as errors.

For this reason, merge the subnets in NetPro.

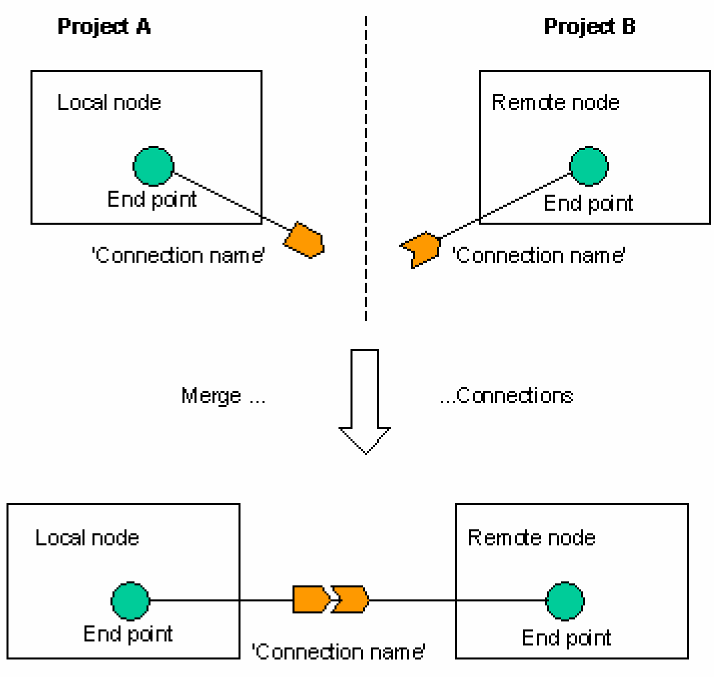
Cross-project connections to a partner in an "unknown" project

If you cannot "lay your hands" on the connection partner in the multiproject, simply because the corresponding project was created at another location or is currently being edited and therefore locked, you can select a connection with "Partner In Unknown Project". Repeat this selection at the partner project.



This procedure generates a standby connection in both projects. You can use this connection later to automatically adjust the partner project, after it has been incorporated into the multiproject.

In this case you need to configure an **identical connection name (reference)** in the connection properties of both projects. Using the connection name, you can assign a connection partner and also adjust the connection properties (Menu command **Edit > Merge Connections**).



Special considerations for downloads

After you have configured cross-project subnets and connections, you must download the network configuration to all participating modules. They represent the end points of the connection as well as the participating routers.

When you upload these data to the PG (Download to PG) the network configurations and the connections are merged automatically as soon as all conditions for this action have been met (e.g. both end points have been uploaded).

The download functions provided under NetPro do not affect cross-project data. rather, they only have an effect within a project. Functions affected are:

- **Download in the Current Project > Selected and Partner Stations**
- **Download in the Current Project > Stations on the Subnet**
- **Download in the Current Project > Selected Connections**

Save and compile functions are also restricted to the currently active project.
An S7 cross-project connection requires of you to compile both participating projects.

These restrictions apply to the "standard" network view of a project. You can also download throughout the entire project ("cross-project") in the cross-project (multiproject-oriented) network view.

15.9 Options for Merging Cross-Project Connections

You can merge cross-project connections such as STEP 7 connections.

- When adjusting projects within a multiproject in the SIMATIC Manager (menu command **File > Multiproject > Adjust Projects**)
- In NetPro with the menu command **Edit > Merge Connection**

However there are the following differences between both procedures:

In the SIMATIC Manager, only those connections are merged that were configured in the affected projects as "Connection partners in unknown project" with the **identical connection names** (reference). In NetPro, however, you can relate connections that have similar or different connection names to each other.

When merging in the SIMATIC Manager, you cannot know in advance which connection partner has which connection properties or which connection partner is the one that adapts its connection properties (i.e. is the active connection partner). When merging in NetPro, the partner adapts its connection properties to those of the local module. In addition, the properties of the connections can be changed in a dialog for merging the connections.

STEP 7 connections to an unspecified partner can only be merged to a cross-project S7 connection in NetPro. Such connections are not handled in the SIMATIC Manager.

15.10 S7 Connections to Unspecified Connection Partners

When you insert a project that contains unspecified S7 connections in a multiproject, you can use a simple method to merge these S7 connections with cross-project S7 connections:

1. Merge the subnets across which the unspecified S7 connection is routed.
2. Call menu command **Edit > Merge Connections**.
   STEP 7 will automatically merge the matching unspecified S7 connections.

15.11 Adjusting Projects in the Multiproject

After projects were inserted into a multiproject, the projects must be adjusted in the multiproject. You also call this function after you have modified subnets and connections in a multiproject. Always execute this function when you prepare your configuration for commissioning.

15.12 Archiving and Retrieving Multiprojects

Archiving multiproject data

You can add multiprojects, individual projects or libraries to a compressed (packed) archive file that can be stored on a hard disk drive or on a mobile storage medium (e.g. Zip disk).

If segments of the multiproject are stored on network drives, you can only use the following file compression tool to create an archive for multiproject data:

- PKZip V12.4 (ships with STEP 7)
- WinZip (tested with Version 14)

Requirement for archiving:

Since the archiving function affects all projects of the multiproject (that is, is a cross-project function), the relevant projects must not be accessed by any other processes.

Procedure

1. In SIMATIC Manager, select the multiproject
2. Select menu command **File > Archive**
3. Confirm the multiproject in the next dialog and click "OK".
4. In the next dialog, select the archive name and path as well as the archiving program (e.g. PKZip)
5. Confirm the dialog with "OK".

How to retrieve multiproject archives

1. In the SIMATIC Manager, select menu command **File > Retrieve**
2. In the next dialog, select the multiproject archive and click on "Open".
3. In the "Select destination directory" dialog that appears, select the target directory for unpacking the archive files.
   A new directory is generated in the selected directory. The complete project directory structure of the unpacked multiproject will then appear on the same level below this directory.

16 Commissioning and Service

Preparations for commissioning

For initial commissioning of networked stations you must initially download configuration data to each one of the PLCs separately via menu command **PLC > Download** (SIMATIC Manager). This supplies the PLCs with their required hardware configuration as well as their network node address. After that you can access the network with the menu command **PLC > Compile And Download Objects**.

What has to be done if modifications were made during a commissioning or service phase

If a multiproject is affected, we recommend that you adjust the projects before you download them to the PLC (In SIMATIC Manager, call menu command **File > Multiproject > Adjust projects**). After that you can call the central function "Compile and download objects" to download the changes to the PLC.

16.1 Commissioning PROFIBUS Nodes

As of STEP 7 V5.2 you can use your PG for online access to PROFIBUS nodes, even if only DP slaves and are connected to the PROFIBUS apart from the PG.

At these nodes you can perform diagnostics, monitor their inputs and modify the outputs. You can also reconfigure the nodes and save the currently displayed configuration as the new configuration (select the menu command **PLC > PROFIBUS > Save PROFIBUS Online Configuration as**).

You can use the configuration saved in this manner as a basis for configuring new station (in HW Config, select the menu command **Station > Open**).

How to start online access

In SIMATIC Manager, start this function with menu command **PLC > PROFIBUS > Operator Control/Monitoring And Diagnosis Of Nodes**, or, when you configure the hardware for an open offline station, with menu command **PLC > Operator Control/Monitoring And Diagnosis Of Nodes**.

You will be displayed a view similar to that of HW Config (**Station > Open Online**). The difference is here that you can specify nodes (e.g. via the context-sensitive menu **Specify Module**), modify parameters and save the new configuration as new station.

Offline and online views

After you open the stored configuration ("Offline view") and then start the "Online view" under menu command **PLC > Operator Control/Monitoring And Diagnosis Of Nodes**, the following relationship is applied:

- The Online view displays all nodes found, irrespective of the number of nodes in the offline configuration.
- STEP 7 uses the manufacturer ID and the configuration data as base for an attempt to find a corresponding node in the offline view for each node in the online view.
  If successful, (that is, all nodes are identical), the offline view parameters are applied to the online view.

This method ensures that changes made and saved during the commissioning phase are immediately available again when the online is opened.

Operator control and monitoring

In the online view you can read inputs and control outputs. Here, the programming device (PG) functions as the DP master.

1. Select the required node.
2. Call menu command PLC > Operator Control/Monitoring.

The operator control/monitoring dialog box pops up. The path definition indicates that online access was performed directly via PROFIBUS services.

The difference between calling this function during the hardware configuration (online via assigned CPU services) and direct access via PROFIBUS services is that triggering conditions cannot be applied to DP slaves.

The dialog displays all inputs/outputs of a module in one line.

Using a saved configuration as a basis for configuring a new station

The PROFIBUS nodes determined online can be saved as a station configuration (menu command **PLC > PROFIBUS > Save PROFIBUS Online Configuration As...**). Any changes made (in "Specify Module" or "Change Properties") are also saved.

This newly saved configuration does not, however, contain any specific configuration data for the DP master connected to the PROFIBUS. For this reason, the DP masters are only shown symbolically, in a way similar to DP slaves.

You have the option of configuring DP masters at a later time and then replacing a DP master determined online (and represented symbolically) with the newly configured one.

Procedure:

1. Open the saved PROFIBUS online configuration.
2. Configure the same number of DP masters as the number of DP master systems that were determined online
   (the complete rack with CPU and PROFIBUS CP or CPUs with integral DP interfaces).
3. Select one of the DP masters determined online that you want to replace with a newly configured DP master.
4. Select the menu command PLC > PROFIBUS > Assign Master.
5. In the dialog that is displayed, assign a configured master to a master that was determined online.
6. Repeat steps 3 to 5 until all of the DP masters determined online are replaced by configured ones.

16.2 Identification & Maintenance (I&M)

I&M functions define information functions that you can use to call up information about devices, such as the manufacturer, version, ordering information etc. With I&M functions, you can find out information about a device during the various phases of configuration, commissioning, parameter assignment, diagnostics and repair.

What devices support I&M data?

PROFIBUS DPV1 slaves generally support I&M data. No I&M data is defined for PROFIBUS DPV0 slaves.

PROFINET IO devices also support I&M data.

New Features of I&M Functions

Information and maintenance functions are an integral part of S7 components, known by the name "Module Identification".

What is new about I&M functions is that accessing this information is now standardized by PROFIBUS AND PROFINET International. Since STEP 7 supports this access, you can also process I&M data from non-S7 components.

I Data

I data is information on the module. Some of this information may also be printed on the module housing.

You can read I data during module diagnostics in STEP 7 (in "Module Information" go to the "General" tab and the "Identification" tab).

M Data

M data is plant-dependent information such as the plant designation, location designation, date installed and any comments.

M data can be written to the module online.

Note

If present, I&M data can only be transferred when the CPU is in STOP.

16.3 Configuring the CPU for Web Access

Web pages can be made available on an S7 CPU. You can download language-specific information to CPUs, to allow information to be read in the required language in a Web browser.

To allow this, the CPU must provide "Web server" functionality.

Information that can be read using a Web browser

The following information can be provided by a CPU with Web server functionality:

- Start page
- Identification of the CPU
- Diagnostic buffer entries
- Module information list
- Messages (for example message blocks AlarmS/SQ, AlarmD/DQ)
- Information on communication
- Topology display with diagnostic information
- Variable status
- Variable tables (assuming that configuration data was downloaded to the CPU for Web access)
- User-defined Web pages

For details of the full scope of information that the Web server can provide for the CPU, refer to the CPU description.

Requirements for Web access

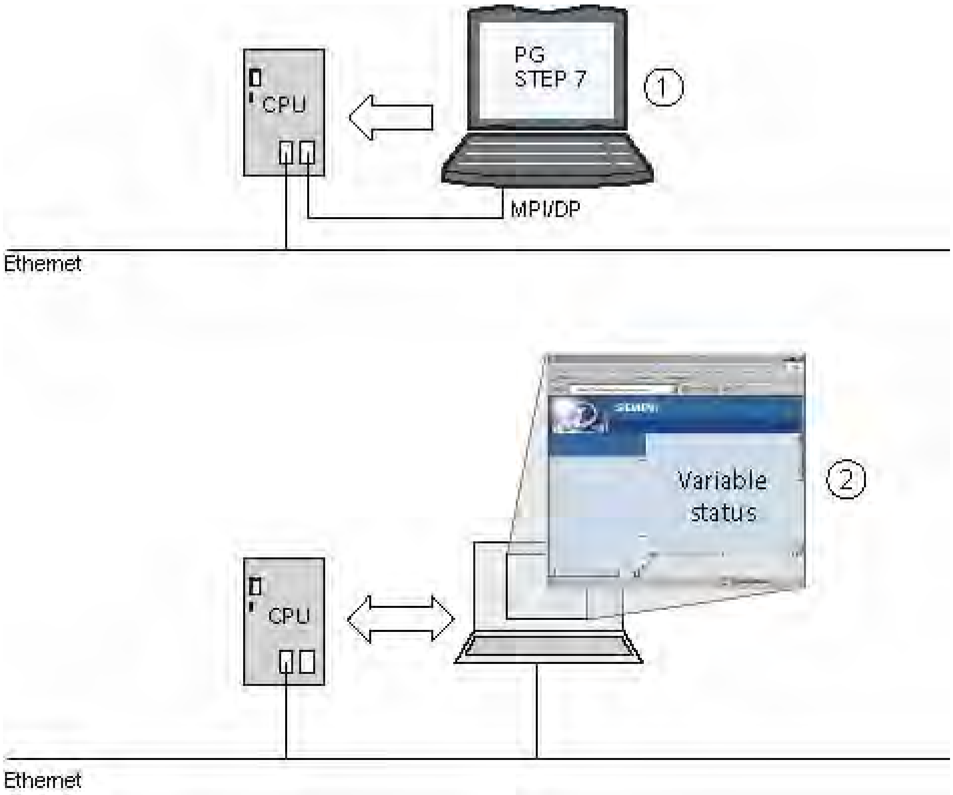
The Web server of the CPU must already be started up.

The Web server starts only when you have enabled the "Enable Web server on this module" check box in the "Web" tab of the parameter assignment dialog and have downloaded this parameter assignment to the CPU.

Notes

- As default, the Web pages are transferred via an unsecured connection and are not safe from attacks by third parties. If you want the Web pages to be transferred encrypted to the browser, enable the check box "Allow access only via HTTPS" in the "Web" tab. Note that the URL of the CPU begins with https:// in this case.
- Access via HTTPS only works if the time was set on the CPU. You set the time in the SIMATIC Manager with the menu command: **PLC > Diagnostics > Set Time of day...**
- To achieve greater access protection, you can create users in the "Web" tab and assign rights and passwords to them.

The following schematic illustrates the basics of how this works.



(1) Download configuration information for Web access to the CPU

(2) Read information from the CPU using the Web browser

Web access without configuration information

Without configuration, only basic information is available on the Web pages. This information is available in five languages (German, English, Spanish, Italian, French):

- Without configuration information, diagnostic buffer entries are displayed not as plain language but in hexadecimal code.
- If the Web server of the CPU supports variable tables:
  The variable table page cannot display variables without prior configuration. You can always query a variable status over a Web browser. You specify the addresses whose status you want to query on this page.

Web access with configuration information

The configuration information includes texts in the selected languages:

- Texts for diagnostic buffer entries as short and long text
- Texts for messages; for example from the message blocks AlarmS/SQ, AlarmD/DQ
- If variable tables are supported: Variable tables with the symbols and comments they contain

The texts for diagnostic buffer entries and messages are displayed during operation in the language you set in the Web browser.

Points to remember about a variable table: The texts you have entered as symbols and comments for the variable table are displayed unchanged in the Web browser regardless of the language you selected for Web access in the configuration. Pure comment rows within the variable table as found in the variable editor are ignored.

Web access to user-defined Web pages

You can run user-defined Web pages on your CPU. User-defined Web pages are programmed in HTML and are expanded by AWP (Advanced Web Programming) commands. The CPU parses (analyzes and codes) these AWP commands so that it can represent variables from the process image as part of a freely programmed Web page.

To configure user-defined Web pages, you require the following:

- Good knowledge of HTML and / or a Web design tool.
- An installation of the STEP 7 add-on "Web2PLC".

You will find further information on the procedures for creating user-defined Web pages in the online help of "Web2PLC".

The installation software for the Web2PLC tool is on the STEP 7 product DVD (CD_2/Optional Components).

Configuring the CPU

The selection of configuration information is restricted to the languages to be downloaded.

Since the storage capacity of the CPU is limited, there are restrictions depending on the CPU you are using:

- The number of selectable languages is limited; for example to two languages.
- If the Web server of the CPU supports variable tables:
  Identify the variable tables you want to work with over a Web browser as follows:
  - Select the "Web server" check box.

    or
  - Write "VATtoWEB" in the "Family" box of this variable table ("General - Part 2" tab). If a variable table has this family name, STEP 7 generates configuration data only for variable tables.

Requirements

All texts for the Web server must be located in the project when you compile the hardware configuration with the settings for the Web server.

Procedure

1. Open the object properties of the CPU with the Web server in HW Config.
2. Click the "Web" tab.
   - If you want the Web server to be enabled during CPU startup, select the "Enable Web server on this module" check box and select the languages for operation.
   - You can select the languages that were installed with STEP 7 (menu command **Options > Language for Display Devices** in the SIMATIC Manager).
   - The number of selectable languages is CPU dependent.
3. Acknowledge the settings with "OK".
4. Download the configuration to the CPU.

Notes

- If you expand the user program and, for example, add or change message texts or you have generated diagnostics blocks with "Report System Error", you will need to download the hardware configuration again.
- In addition to the procedure described above, you also have the option of enabling only specific display classes. This allows you, for example, to suppress uninteresting messages or to restrict memory requirements.

Opening Web pages of the CPU with a Web browser

1. Connect the PG/PC over the Ethernet interface.
2. Open the Web browser (for example Internet Explorer).
3. In the "Address" box of the Web browser, enter the IP address of the CPU in the form http://a.b.c.d.
   The start page of the CPU opens. From here, you can then navigate to the remaining information.

Points to note when using PDAs

With PDAs (devices with Windows CE/Mobile), Web pages with an adapted design are available at the address http://a.b.c.d/Basic.

16.4 Changing M Data in the SIMATIC Manager

Requirements

- The module supports I&M data.
- There is an online connection to the module over PROFIBUS "Accessible nodes".

Procedure

If you want to edit M data (for example the plant designation), follow the steps below:

1. Select the module in the "Accessible Nodes" window and then select the menu command **PLC >** Change Module Identification.
2. Edit the M data in the "Change Module Identification" dialog. If you have redundant modules, you will need to edit the data for the left and right module.
3. For data to be transferred to the module, enable the "Include" check box.
4. Acknowledge your entries with "OK".

The M data is transferred to the module.

16.5 Entering or Changing M Data in HW Config

Requirements

- The module supports I&M data.
- There is an online connection over PROFIBUS or PROFINET to the module.

Procedure

If you want to edit M data (for example the plant designation), follow the steps below:

1. Double-click on the module and select the "Identification" tab.
2. Enter the M data and confirm with "OK".
3. Select the menu command PLC > Download Module Identification.
4. In the "Download Module Identification" dialog box, compare the offline data in the STEP 7 database with the data on the module ("ONLINE").
5. To overwrite the online data with the offline data, select the "Include" check box.
6. Acknowledge the settings with "OK".

The M data is transferred to the module.

Index