SIEMENS

Edition 11/2023

FUNCTION MANUAL

SIMATIC

S7-1500, ET 200MP, ET 200SP, ET 200AL, ET 200pro, ET 200eco PN

PROFINET with STEP 7

support.industry.siemens.com

SIEMENS

SIMATIC

S7-1500, ET 200MP, ET 200SP, ET 200AL, ET 200pro, ET 200eco PN PROFINET with STEP 7

Function Manual

11/2023

A5E03444486-AN

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

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

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

Qualified Personnel

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

Proper use of Siemens products

Note the following:

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

Trademarks

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

Disclaimer of Liability

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

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E03444486-AN
Ⓟ 10/2023 Subject to change

Table of contents

Introduction

1

Purpose of the documentation

This Function Manual provides an overview of the PROFINET communications system together with SIMATIC STEP 7.

STEP 7 is integrated into the high-performance, graphical Totally Integrated Automation Portal (TIA Portal), the integration platform for all automation software tools.

This Function Manual supports you in planning a PROFINET system. The manual is structured into the following subject areas:

- PROFINET basics
- PROFINET diagnostics
- PROFINET functions

Basic knowledge required

The following knowledge is required in order to understand the manual:

- General knowledge of automation technology
- Knowledge of the industrial automation system SIMATIC
- Knowledge about the use of Windows-based computers
- Knowledge about how to use STEP 7 (TIA Portal)

Scope

This documentation is the basic documentation for all SIMATIC products from the PROFINET environment. The product documentation is based on this documentation.

The examples are based on the functionality of the S7-1500 automation system.

What's new in the PROFINET Function Manual, 11/2023 edition compared to 11/2022 edition

Function	What are the customer benefits?	Where can I find this information?
Implementation of PROFINET Security Class 1	You read which configuration options for protocols and processes in STEP 7 V19 have been adapted for PROFINET Security Class 1.	Section PROFINET Security Class 1 (Page 34)
Configuring SNMP and DCP	As of V19, STEP 7 offers extended configuration options for the SNMP and DCP protocols in order to meet the requirements for PROFINET Security Class 1.	Section Configuring SNMP and DCP in the PROFINET network (Page 298)
Project-internal shared device / shared I-device	As of STEP 7 V19, a shared device/shared I-device can be created together with a maximum of two IO controllers in a project. Previously, the 2nd IO controller required a separate project.	Section Shared device (Page 147)

Function	What are the customer benefits?	Where can I find this information?
Handling timeouts during data exchange	In the case of high network loads, timeouts may occur during data record communication in PROFINET IO devices. Previously, PROFINET IO communication was terminated by the CPU in this case. As of STEP 7 V19 and FW version V3.1, you can configure the behavior of the respective PROFINET interface.	Section Dealing with timeouts while exchanging data (Page 289)

What's new in the PROFINET Function Manual, 11/2022 edition compared to 05/2021 edition

Function	What are the customer benefits?	Where can I find this information?
Project-internal shared device	As of STEP 7 V18, it is also possible to configure project-internal shared devices under certain conditions.	Section Shared device (Page 147)
Maintenance of the communication relation (AR)	In the event of high network load, the communication relation is terminated by the CPU. As of FW version V3.0, the communication relation can be maintained by writing a data record.	Section Dealing with timeouts while exchanging data (Page 289)
Support PROFINET system redundancy R1	As of FW version V3.0, the S7-1500H supports PROFINET system redundancy R1. R1 devices are equipped with two interface modules compared to S2 devices. If one interface module fails, the R1 device can still be reached by the H-CPUs via the second interface module. Thus, R1 devices have a higher availability than S2 devices.	Section PROFINET with redundant S7-1500R/H system (Page 304)
MRP interconnection	As of FW version V3.0, the CPUs of the S7-1500H redundant system may be located in 1 or 2 rings during MRP interconnection.	Section MRP Interconnection with the redundant S7-1500R/H system (Page 329)

What's new in the PROFINET Function Manual, Edition 05/2021 compared to Edition 11/2018

Function	What are the customer benefits?	Where can I find this information?
MRP interconnection	The MRP interconnection procedure is an extension of MRP. MRP interconnection allows the redundant coupling of 2 or more rings with MRP in PROFINET networks. MRP interconnection offers the following advantages: There is no limit to the maximum number of 50 devices in a ring when setting up redundant network topologies. Monitoring of larger topologies with ring redundancy.	Section MRP interconnection (Page 193) Section MRP interconnection with the redundant system S7-1500R/H (Page 329)
Enable/disable the I-device	You can use the "D_ACT_DP" instruction to locally disable or enable the I-device function in the user program of the I-device CPU.	Section Enabling/disabling I-device in the user program of the I-device CPU (Page 142)

What's new in the PROFINET Function Manual, Edition 11/2019 compared to Edition 10/2018

Function	What are the customer benefits?	Where can I find this information?
Direct data exchange	In the case of direct data exchange, an S7-1500 CPU provides cyclic net data from the I/O area to one or more partners. The "Direct data exchange" function enables deterministic, isochronous IO communication between multiple S7-1500 CPUs.	Section Direct data exchange (Page 232)
Switched S1 device	The "Switched S1 device" function of the CPU enables operation of standard IO devices in the S7-1500R/H redundant system.	Section Switched S1 device (Page 311)

What's new in the PROFINET Function Manual, Edition 10/2018 compared to Edition 12/2017

This manual (version 10/2018) includes the following new functions compared to the previous version (version 12/2017):

Function	Applications	Your benefits
PROFINET IO with the redundant S7-1500R/H system	In a PROFINET IO system with the redundant S7-1500R/H system, the IO communication continues even when one CPU fails.	The redundant S7-1500R/H system offers a high degree of reliability and system availability. A redundant configuration of the most important automation components reduces the probability of production downtimes and the consequences of component failures.

What's new in the PROFINET Function Manual, version 12/2017 compared to version 09/2016

This manual (version 12/2017) encompasses the following new functions compared to the previous version (version 09/2016):

Function	Applications	Your benefits
Specifying the router for a PROFINET IO device	You can specify the IP address of a router for each IO device. You reach the IO device from outside the IP subnet through the router.	In the past, it was only possible to specify a router for a PROFINET IO interface at the IO controller. The IO devices inherited the setting of the IO controller interface. Now you can set the router address independently of the IO controller setting. This allows, for example, a router address at the IO device although you have not set a router address or have set a different address at the IO controller.
Configuring an IO device through hardware detection	You can detect an existing IO device and enter it in your project.	STEP 7 inserts the IO device with all the modules and submodules into the project. Article numbers and firmware versions between real and configured IO devices match. You reduce the configuration effort required.
Asset management	You can centrally manage non-PROFINET components (assets) of a PROFINET device. The PROFINET device makes the identification data of the assets available for evaluation via a standardized data record.	The new standardized PROFINET service makes it possible to manage all the hardware and firmware components of PROFINET devices centrally. The possibilities available for filtering device data, for example, depend on the range of performance of the evaluating application.

Function	Applications	Your benefits
Asset management data record for I-devices	Special application of asset management: From the point of view of a higher-level IO controller, the modules plugged into the I-device represent assets. The user program in the I-device compiles the asset management data record. The IO controller can read identification data of the I-device modules through this data record.	See Asset management.

What's new in the PROFINET Function Manual, Version 09/2016 compared to Version 12/2014

This manual (version 09/2016) encompasses the following new functions compared to the previous version (version 12/2014):

Function	Applications	Your benefits
PROFINET IO on the 2nd PROFINET interface	You can operate another PROFINET IO system on the CPU or connect additional IO devices.	You use a fieldbus type in the plant. The CPU can perform fast and deterministic data exchange as an I-device with a higher-level controller (PROFINET/Ethernet) through the second line.
IRT with very short data cycle times down to 125 μs	You realize high-end applications with IO communication which place very high performance demands on the IO processing.	You make PROFINET IO communication and standard communication possible via one cable even with a send clock of 125 μs.
MRPD: Media Redundancy with Planned Duplication of frames	PROFINET IO IRT enables you to realize applications that place particularly high demands on the reliability and accuracy (isochronous mode).	By sending the cyclic IO data in both directions in the ring, the communication to the IO devices is maintained even when the ring is interrupted and does not result in device failure even with fast update times. You achieve higher reliability than with MRP.
PROFINET performance upgrade	You can implement applications with high speed and send clock requirements. This is of interest to applications with high demands on performance.	Better utilization of the bandwidth results in short reaction times.
Limitation of the data infeed into the network	You limit the network load for standard Ethernet communication to a maximum value.	You flatten peaks in the data feed. You share the remaining bandwidth based on demand.

Conventions

STEP 7: We refer to "STEP 7" in this documentation as a synonym for the configuration and programming software "STEP 7 as of V12 (TIA Portal)" and subsequent versions.

This documentation contains pictures of the devices described. The figures may differ slightly from the device supplied.

You should also pay particular attention to notes such as the one shown below:

NOTE

A note contains important information on the product, on handling of the product and on the section of the documentation to which you should pay particular attention.

Industry Mall

The Industry Mall is the catalog and order system of Siemens AG for automation and drive solutions on the basis of Totally Integrated Automation (TIA) and Totally Integrated Power (TIP).
You can find catalogs for all automation and drive products on the Internet (https://mall.industry.siemens.com).

1.1 Function manuals documentation guide

1.1.1 Information classes Function Manuals

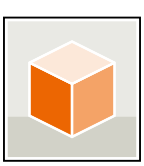
The documentation for the SIMATIC S7-1500 automation system, for the 1513/1516pro-2 PN, SIMATIC Drive Controller CPUs based on SIMATIC S7-1500 and the SIMATIC ET 200MP, ET 200SP, ET 200AL and ET 200eco PN distributed I/O systems is arranged into three areas.
This arrangement enables you to access the specific content you require.
You can download the documentation free of charge from the Internet (https://support.industry.siemens.com/cs/ww/en/view/109742705).

Basic information

The system manuals and Getting Started describe in detail the configuration, installation, wiring and commissioning of the SIMATIC S7-1500, SIMATIC Drive Controller, ET 200MP, ET 200SP, ET 200AL and ET 200eco PN systems. Use the corresponding operating instructions for 1513/1516pro-2 PN CPUs.
The STEP 7 online help supports you in the configuration and programming.
Examples:

- Getting Started S7-1500
- System manuals
- Operating instructions ET 200pro and 1516pro-2 PN CPU
- Online help TIA Portal

Device information



Equipment manuals contain a compact description of the module-specific information, such as properties, wiring diagrams, characteristics and technical specifications.
Examples:

- Equipment manuals for CPUs
- Equipment manuals for interface modules
- Equipment manuals for digital modules
- Equipment manuals for analog modules
- Equipment manuals for communication modules
- Equipment manuals for technology modules
- Equipment manuals for power supply modules
- Equipment manuals for BaseUnits

General information

The function manuals contain detailed descriptions on general topics relating to the SIMATIC Drive Controller and the S7-1500 automation system.

Examples:

- Function Manual Diagnostics
- Function Manual Communication
- Function Manuals Motion Control
- Function Manual Web Server
- Function Manual Cycle and Response Times
- PROFINET Function Manual
- PROFIBUS Function Manual

Product Information

Changes and supplements to the manuals are documented in a Product Information. The Product Information takes precedence over the device and system manuals.

You will find the latest Product Information on the Internet:

- S7-1500/ET 200MP (https://support.industry.siemens.com/cs/de/en/view/68052815)
- SIMATIC Drive Controller (https://support.industry.siemens.com/cs/de/en/view/109772684/en)
- Motion Control (https://support.industry.siemens.com/cs/de/en/view/109794046/en)
- ET 200SP (https://support.industry.siemens.com/cs/de/en/view/73021864)
- ET 200eco PN (https://support.industry.siemens.com/cs/ww/en/view/109765611)

Manual Collections

The Manual Collections contain the complete documentation of the systems put together in one file.

You will find the Manual Collections on the Internet:

- S7-1500/ET 200MP/SIMATIC Drive Controller (https://support.industry.siemens.com/cs/ww/en/view/86140384)
- ET 200SP (https://support.industry.siemens.com/cs/ww/en/view/84133942)
- ET 200AL (https://support.industry.siemens.com/cs/ww/en/view/95242965)
- ET 200eco PN (https://support.industry.siemens.com/cs/ww/en/view/109781058)

1.1.2 Basic tools

Tools

The tools described below support you in all steps: from planning, over commissioning, all the way to analysis of your system.

TIA Selection Tool

The TIA Selection Tool tool supports you in the selection, configuration, and ordering of devices for Totally Integrated Automation (TIA).

As successor of the SIMATIC Selection Tools , the TIA Selection Tool assembles the already known configurators for automation technology into a single tool.

With the TIA Selection Tool , you can generate a complete order list from your product selection or product configuration.

You can find the TIA Selection Tool on the Internet.
(https://support.industry.siemens.com/cs/ww/en/view/109767888)

SIMATIC Automation Tool

You can use the SIMATIC Automation Tool to perform commissioning and maintenance activities on various SIMATIC S7 stations as bulk operations independent of TIA Portal.

The SIMATIC Automation Tool offers a wide range of functions:

- Scanning of a PROFINET/Ethernet system network and identification of all connected CPUs
- Assignment of addresses (IP, subnet, Gateway) and device name (PROFINET device) to a CPU
- Transfer of the date and the programming device/PC time converted to UTC time to the module
- Program download to CPU
- RUN/STOP mode switchover
- CPU localization through LED flashing
- Reading out of CPU error information
- Reading the CPU diagnostic buffer
- Reset to factory settings
- Firmware update of the CPU and connected modules

You can find the SIMATIC Automation Tool on the Internet.
(https://support.industry.siemens.com/cs/ww/en/view/98161300)

PRONETA

SIEMENS PRONETA (PROFINET network analysis) is a commissioning and diagnostic tool for PROFINET networks. PRONETA Basic has two core functions:

- In the network analysis, you get an overview of the PROFINET topology. Compare a real configuration with a reference installation or make simple parameter changes, e.g. to the names and IP addresses of the devices.
- The "IO test" is a simple and rapid test of the wiring and the module configuration of a plant, including documentation of the test results.

You can find SIEMENS PRONETA Basic on the Internet:
(https://support.industry.siemens.com/cs/ww/en/view/67460624)

SIEMENS PRONETA Professional is a licensed product that offers you additional functions. It offers you simple asset management in PROFINET networks and supports operators of automation systems in automatic data collection/acquisition of the components used through various functions:

- The user interface (API) offers an access point to the automation cell to automate the scan functions using MQTT or a command line.
- With PROFIenergy diagnostics, you can quickly detect the current pause mode or the readiness for operation of devices that support PROFIenergy and change these as needed.
- The data record wizard supports PROFINET developers in reading and writing acyclic PROFINET data records quickly and easily without PLC and engineering.

You can find SIEMENS PRONETA Professional on the Internet.
(https://www.siemens.com/proneta-professional)

SINETPLAN

SINETPLAN, the Siemens Network Planner, supports you in planning automation systems and networks based on PROFINET. The tool facilitates professional and predictive dimensioning of your PROFINET installation as early as in the planning stage. In addition, SINETPLAN supports you during network optimization and helps you to exploit network resources optimally and to plan reserves. This helps to prevent problems in commissioning or failures during productive operation even in advance of a planned operation. This increases the availability of the production plant and helps improve operational safety.

The advantages at a glance

- Network optimization thanks to port-specific calculation of the network load
- Increased production availability thanks to online scan and verification of existing systems
- Transparency before commissioning through importing and simulation of existing STEP 7 projects
- Efficiency through securing existing investments in the long term and the optimal use of resources

You can find SINETPLAN on the Internet
(https://new.siemens.com/global/en/products/automation/industrial-communication/profinet/sinetplan.html).

1.1.3 S7 Port Configuration Tool (S7-PCT)

SIMATIC S7-PCT

The Port Configuration Tool (PCT) is a PC-based software for the parameter assignment of Siemens IO-Link Master modules and IO-Link devices from any manufacturer.

You integrate IO-Link-devices using the standardized device description "IODD", which you get from the respective device manufacturer. S7-PCT supports version 1.0 and V1.1 of the IODD.

S7-PCT is called via the hardware configuration of the IO-Link Master from STEP 7. When STEP 7 is not used or the IO-Link Master is not operated on a SIMATIC controller, "standalone"-operation is also possible.

You can find additional information on IO-Link on the Internet (https://new.siemens.com/global/en/products/automation/industrial-communication/io-link.html).

1.1.4 S7 Failsafe Configuration Tool (S7-FCT)

SIMATIC S7-FCT

Failsafe Configuration Tool (FCT) enables you to GSD configure the following devices in third-party engineering systems:

- Selected, functionally fail-safe SIMATIC I/O devices
- Functionally fail-safe SIRIUS ACT PROFINET interfaces

The engineering system must meet the following requirements for this:

- Support of the CPD system integration acc. to "PROFIsafe - Profile for Safety Technology on PROFIBUS DP and PROFINET IO"
- TCI implementation to Conformance Class C3

Additional information on S7-FCT can be found on the Internet (https://support.industry.siemens.com/cs/ww/en/view/109762827).

1.1.5 MultiFieldbus Configuration Tool (MFCT)

MultiFieldbus Configuration Tool

MultiFieldbus Configuration Tool (MFCT) is a PC-based software and supports the configuration of MultiFieldbus- and DALI-devices. In addition, the MFCT offers convenient options for mass firmware updates of ET 200 devices with MultiFieldbus- support and reading service data for many other Siemens devices.

Functional scope of the MFCT

- MultiFieldbus configuration:
  Engineering, configuration and diagnostics of MultiFieldbus-devices, provision of the required project files (project, UDT-, CSV- and EDS-file), transfer/export of the files to device and/or data memory.
- DALI configuration:
  Device selection and online configuration of DALI devices.
- TM FAST:
  Generation and download of FPGA-UPD- and FPGA-DB-files.

- Maintenance:
  Topology scan of a Ethernet network, reading of service data, parameter assignment and firmware update.
- Settings:
  Language switching German / English, network scanner speed, setting of the network adapter, installation of GSDML-and EDS-files.

System/installation requirements for MFCT

The MFCT runs under Microsoft Windows and does not require installation or administrator rights.

For MFCT you must also install the following software:

- Microsoft .NET Framework 4.8 (You can find an Offline Installer on the Internet. (https://support.microsoft.com/en-us/topic/microsoft-net-framework-4-8-offline-installer-for-windows-9d23f658-3b97-68ab-d013-aa3c3e7495e0))
- NPcap from directory "Misc"
- PG/PC interface from directory "Misc"
- Microsoft C++ Redistributable for x86-systems (you can find the installation data for download on the Internet. (https://aka.ms/vs/15/release/vc_redist.x86.exe))

The download of the tool and further information as well as documentation on the individual functions of the MFCT can be found on the Internet. (https://support.industry.siemens.com/cs/de/en/view/109773881)

1.1.6 SIMATIC Technical Documentation

Additional SIMATIC documents will complete your information. You can find these documents and their use at the following links and QR codes.

The Industry Online Support gives you the option to get information on all topics. Application examples support you in solving your automation tasks.

Overview of the SIMATIC Technical Documentation

Here you will find an overview of the SIMATIC documentation available in Siemens Industry Online Support:

Industry Online Support International (https://support.industry.siemens.com/cs/ww/en/view/109742705)

Watch this short video to find out where you can find the overview directly in Siemens Industry Online Support and how to use Siemens Industry Online Support on your mobile device:

Quick introduction to the technical documentation of automation products per video (https://support.industry.siemens.com/cs/us/en/view/109780491)

YouTube video: Siemens Automation Products - Technical Documentation at a Glance (https://youtu.be/TwLSxxRQQsA)

Retention of the documentation

Retain the documentation for later use.

For documentation provided in digital form:

1. Download the associated documentation after receiving your product and before initial installation/commissioning. Use the following download options:
   - Industry Online Support International: (https://support.industry.siemens.com)
     The article number is used to assign the documentation to the product. The article number is specified on the product and on the packaging label. Products with new, non-compatible functions are provided with a new article number and documentation.
   - ID link:
     Your product may have an ID link. The ID link is a QR code with a frame and a black frame corner at the bottom right. The ID link takes you to the digital nameplate of your product. Scan the QR code on the product or on the packaging label with a smartphone camera, barcode scanner, or reader app. Call up the ID link.
2. Retain this version of the documentation.

Updating the documentation

The documentation of the product is updated in digital form. In particular in the case of function extensions, the new performance features are provided in an updated version.

1. Download the current version as described above via the Industry Online Support or the ID link.
2. Also retain this version of the documentation.

mySupport

With "mySupport" you can get the most out of your Industry Online Support.

Registration	You must register once to use the full functionality of "mySupport". After registration, you can create filters, favorites and tabs in your personal workspace.
Support requests	Your data is already filled out in support requests, and you can get an overview of your current requests at any time.
Documentation	In the Documentation area you can build your personal library.
Favorites	You can use the "Add to mySupport favorites" to flag especially interesting or frequently needed content. Under "Favorites", you will find a list of your flagged entries.
Recently viewed articles	The most recently viewed pages in mySupport are available under "Recently viewed articles".
CAx data	The CAx data area gives you access to the latest product data for your CAx or CAe system. You configure your own download package with a few clicks: • Product images, 2D dimension drawings, 3D models, internal circuit diagrams, EPLAN macro files • Manuals, characteristics, operating manuals, certificates • Product master data

You can find "mySupport" on the Internet. (https://support.industry.siemens.com/My/ww/en)

Application examples

The application examples support you with various tools and examples for solving your automation tasks. Solutions are shown in interplay with multiple components in the system - separated from the focus on individual products.
You can find the application examples on the Internet. (https://support.industry.siemens.com/cs/ww/en/ps/ae)

Safety instructions 2

2.1 Cybersecurity information

Siemens provides products and solutions with industrial cybersecurity functions that support the secure operation of plants, systems, machines, and networks.

In order to protect plants, systems, machines, and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial cybersecurity concept. Siemens’ products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For more information on protective industrial cybersecurity measures for implementation, please visit (https://www.siemens.com/global/en/products/automation/topic-areas/industrial-cybersecurity.html).

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customers' exposure to cyber threats.

To stay informed about product updates at all times, subscribe to the Siemens Industrial Cybersecurity RSS Feed under (https://new.siemens.com/global/en/products/services/cert.html).

Description 3

3.1 Introduction to PROFINET

What is PROFINET IO?

Within the framework of Totally Integrated Automation (TIA), PROFINET IO is the logical further development of:

- PROFIBUS DP, the established fieldbus and
- Industrial Ethernet

PROFINET IO is based on 20 years of experience with the successful PROFIBUS DP and combines the normal user operations with the simultaneous use of innovative concepts of Ethernet technology. This ensures the integration of PROFIBUS DP into the PROFINET world.

PROFINET IO as the Ethernet-based automation standard of PROFIBUS/PROFINET International defines a cross-vendor communication, automation, and engineering model.

Objectives of PROFINET

The objectives of PROFINET:

- Industrial networking, based on Industrial Ethernet (open Ethernet standard)
- Compatibility of Industrial Ethernet and standard Ethernet components
- High robustness due to Industrial Ethernet devices. Industrial Ethernet devices are suited to the industrial environment (temperature, noise immunity, etc.).
- Use of IT standards such as TCP/IP, http.
- Real-time capability
- Seamless integration of other fieldbus systems

Implementation of PROFINET in SIMATIC

PROFINET is implemented in SIMATIC as follows:

- We have implemented communication between field devices in SIMATIC with **PROFINET IO**.
- Installation technology and network components are available as SIMATIC NET products.
- Ethernet standard protocol and procedures (e.g., SNMP = Simple Network Management Protocol for network parameter assignment and diagnostics) are used for remote maintenance and network diagnostics.

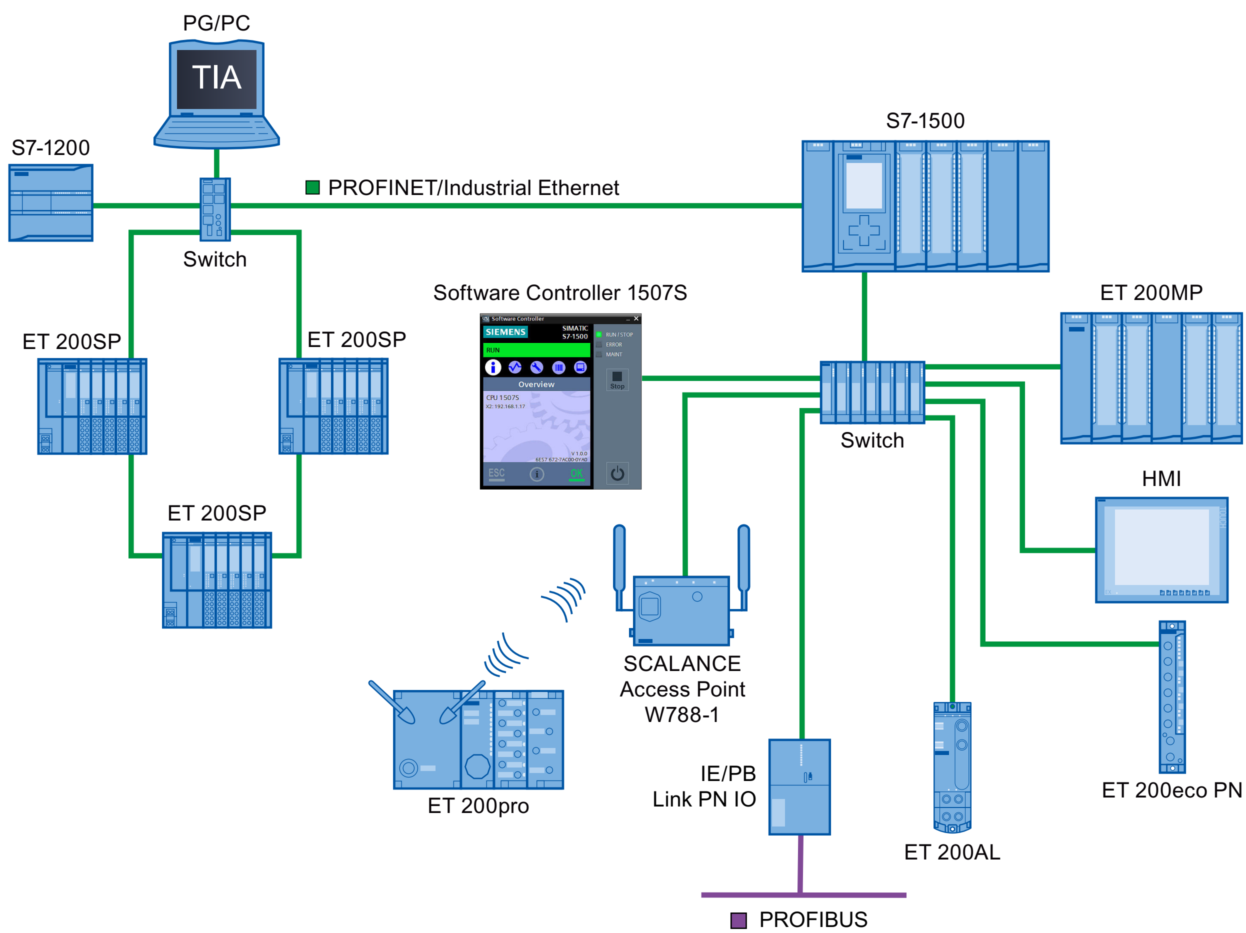


Figure 3-1 PROFINET overview configuration

STEP 7

The STEP 7 engineering tool supports you in setting up and configuring an automation solution. STEP 7 provides a uniform application view over all bus systems.

Documentation from PROFIBUS & PROFINET International on the Internet

You will find numerous documents on the topic of PROFINET at the Internet address (http://www.profibus.com) of the "PROFIBUS & PROFINET International" PROFIBUS user organization, which is also responsible for PROFINET.

Additional information can be found on the Internet (http://www.siemens.com/profinet).

Overview of the most important documents and links

A compilation of the most important PROFINET application examples, FAQs and other contributions in the Industry Online Support is available in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/108165711).

3.1.1 PROFINET terms

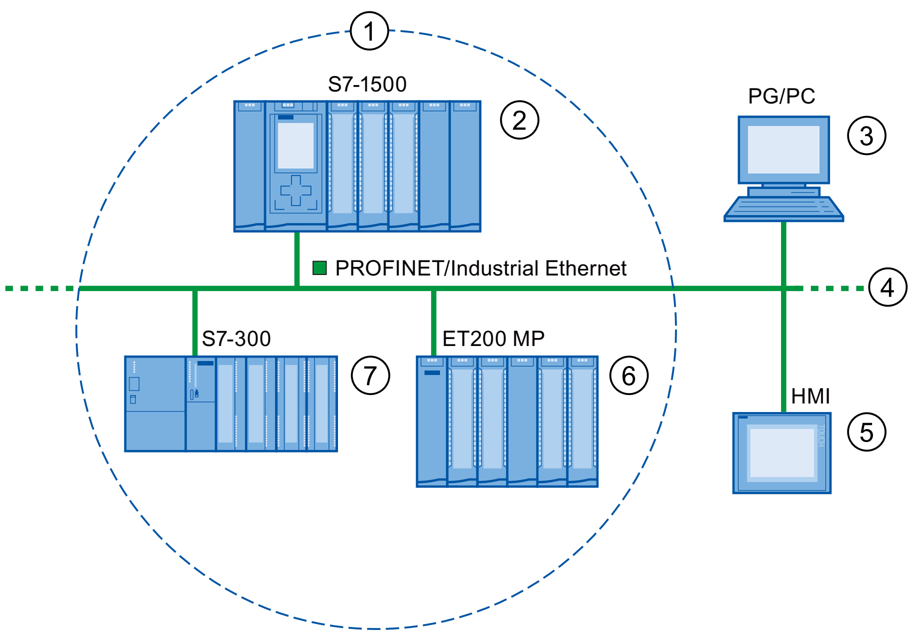
Definition: Devices in the PROFINET environment

In the PROFINET environment, "device" is the generic term for:

- Automation systems (PLC, PC, for example)
- Distributed I/O systems
- Field devices (for example, hydraulic devices, pneumatic devices)
- Active network components (for example, switches, routers)
- Gateways to PROFIBUS, AS interface or other fieldbus systems

PROFINET IO devices

The following graphic shows the general names used for the most important devices in PROFINET. In the table below the graphic you can find the names of the individual components in the PROFINET IO context.



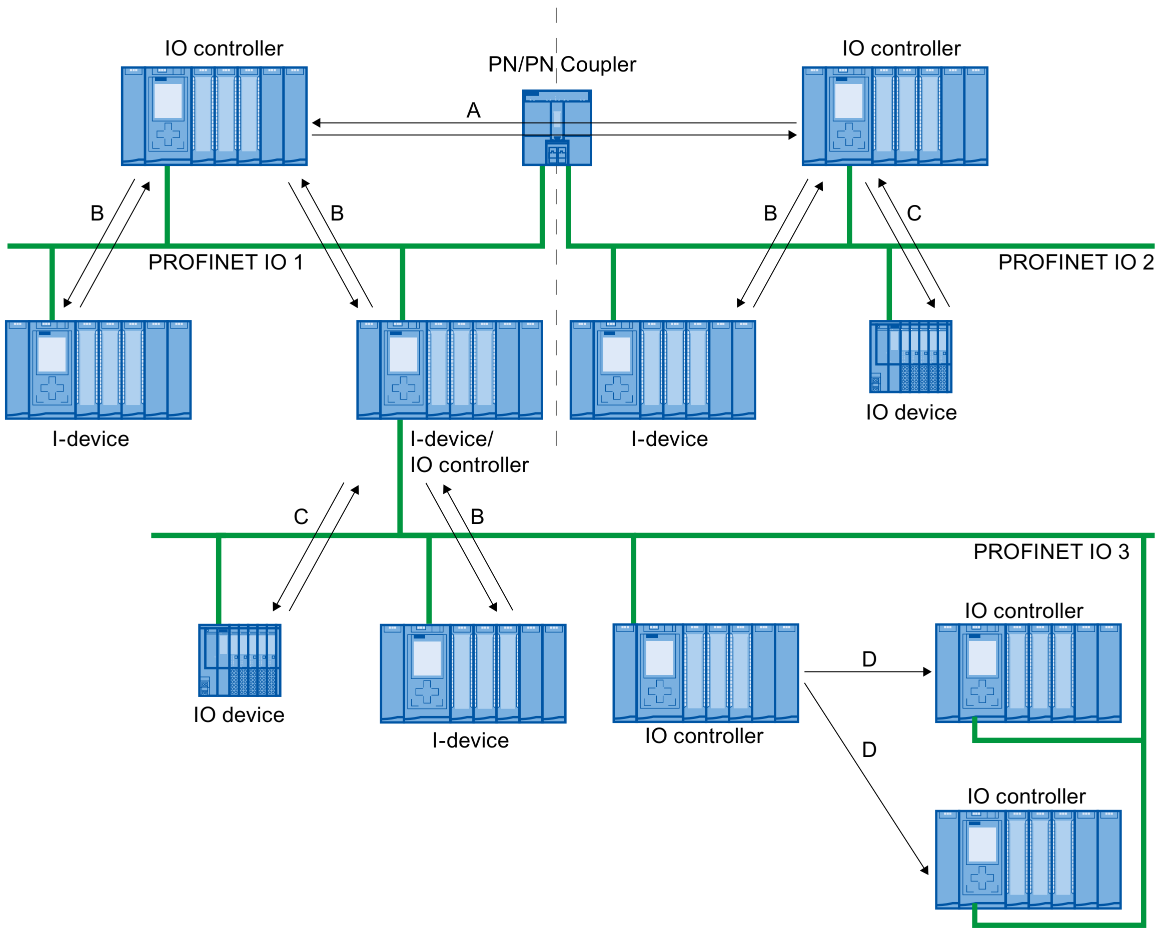
Number	PROFINET	Explanation
①	PROFINET IO System	
②	IO controller	Device used to address the connected IO devices. This means that: The IO controller exchanges input and output signals with field devices.
③	Programming device / PC (PROFINET IO supervisor)	PG/PC/HMI device used for commissioning and for diagnostics
④	PROFINET/Industrial Ethernet	Network infrastructure

⑤	HMI (Human Machine Interface)	Device for operating and monitoring functions.
⑥	IO device	A distributed field device that is assigned to one of the IO controllers (e.g., Distributed IO, valve terminals, frequency converters, switches with integrated PROFINET IO functionality)
⑦	I-device	Intelligent IO device

Figure 3-2 PROFINET devices

IO communication via PROFINET IO

The inputs and outputs of distributed I/O devices are read and written by means of PROFINET IO using what is referred to as IO communication. The following figure provides an overview of IO communication by means of PROFINET IO.



A IO controller - IO controller communication via PN/PN coupler

B IO controller - I-device communication

C IO controller - IO-device communication

D Direct data exchange between S7-1500-CPUs

Figure 3-3 IO communication via PROFINET IO

IO communication via PROFINET IO

Table 3-1 IO communication via PROFINET IO

Communication between ...	Explanation
IO controllers and IO devices	The IO controller sends data cyclically to the IO devices of its PROFINET IO system and receives data from these devices.
IO controller and I-device	A fixed quantity of data is transferred cyclically between the user programs in CPUs of IO controllers and I-devices. The IO controller does not access the I/O module of the I-device, but instead accesses configured address ranges, i.e. transfer ranges, which may be located inside our outside the process image of the CPU of the I-device. If parts of the process image are used as transfer ranges, it is not permitted to use these for real I/O modules. Data transfer takes place using load- and transfer operations via the process image or via direct access.
IO controller and IO controller (PN/PN coupler)	A fixed quantity of data is cyclically transferred between the user programs in CPUs of IO controllers. A PN/PN coupler is required as additional hardware. The IO controllers mutually access configured address ranges, i.e. transfer ranges, which may be located inside or outside the process image of the CPU. If parts of the process image are used as transfer ranges, it is not permitted to use these for real I/O modules. Data transfer takes place using load- and transfer operations via the process image or via direct access. I/O communication with PN/PN coupler is possible between two PROFINET IO systems.
S7-1500-CPU and S7-1500-CPU (direct data exchange)	In the case of direct data exchange, an S7-1500 CPU provides cyclic user data from the I/O area to one or more partners. The direct data exchange is based on PROFINET with IRT and isochronous mode. The data exchange takes place via transfer areas.

See also

Communication (http://support.automation.siemens.com/WW/view/en/59192925)

Network security (Page 44)

Functions (Page 117)

3.1.2 Basic terminology of communication

PROFINET communication

PROFINET communication takes place via Industrial Ethernet. The following transmission types are supported:

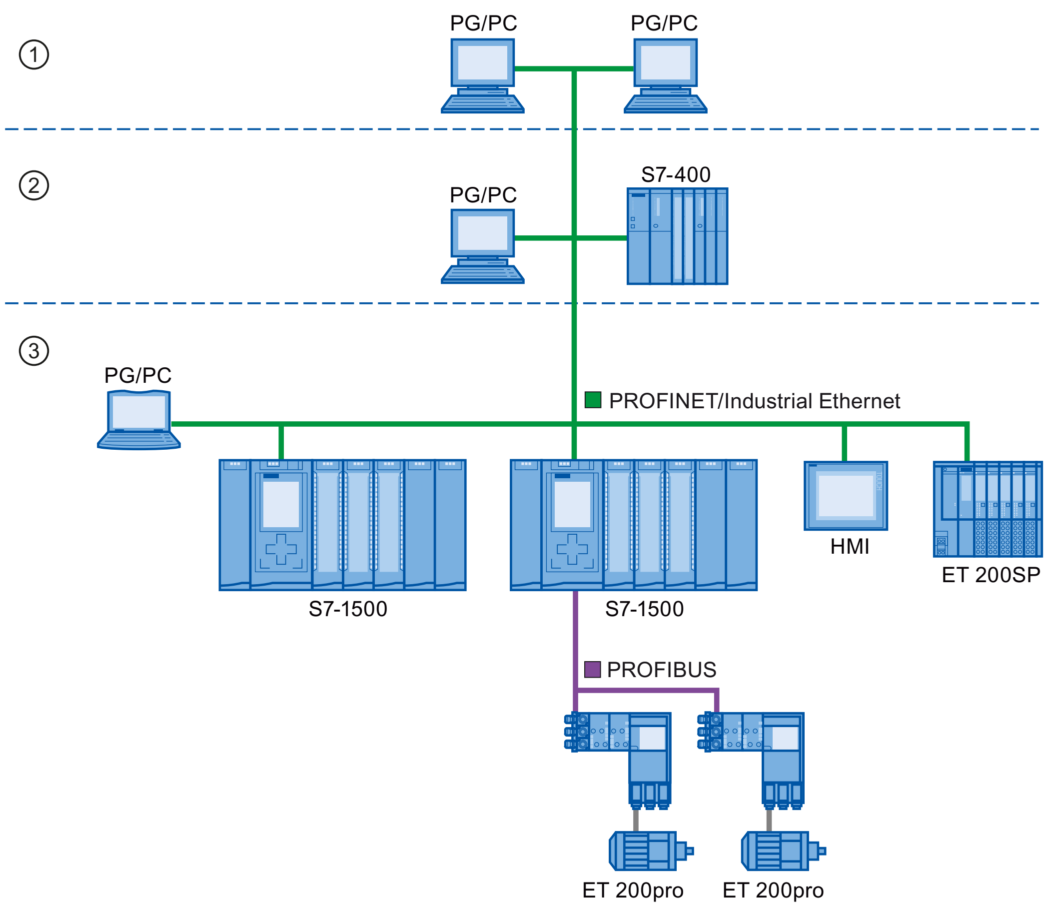
- Acyclic transmission of engineering and diagnostics data and interrupts
- Cyclic transmission of user data

The PROFINET-IO communication takes place in real-time.

For additional information on the real-time communication, refer to chapter Real-time communication (Page 196).

Transparent data access

Access to process data from different levels of the factory is supported by PROFINET communication. By using Industrial Ethernet, standard mechanisms of communication and information technology such as OPC/XML can now be used along with standard protocols such as UDP/TCP/IP and HTTP in automation engineering. This allows transparent access from company management level directly to the data from the automation systems at the control level and production level.



① Management level

② Control level

③ Production level

Figure 3-4 Access to process data

Update time

The update time is a time interval. IO controller and IO device/I-device exchange IO data cyclically in the IO system within this time interval. The update time can be configured separately for each IO device and determines the interval at which output data is sent from the IO controller to the IO device (output module/submodule) as well as input data from the IO device to the IO controller (input module/submodule).

STEP 7 calculates the update time automatically in the default setting for each IO device of the PROFINET IO system, taking into account the volume of data to be exchanged as well as the set send clock.

For additional information on the update time, refer to section Real-time communication (Page 196).

Watchdog time

The watchdog time is the time interval that an IO controller or IO device permits, without receiving IO data. If the IO device is not supplied by the IO controller with data within the watchdog time, the IO device detects the missing frames and outputs substitute values. This is reported in the IO controller as a station failure.

In STEP 7, the watchdog time is made up from an integral multiple of the update time and can be set by the user.

Send clock

The period of time between two consecutive communication cycles. The send clock is the shortest possible interval in data exchange and thus also the smallest value that can be set for the update time.

Relationship between the update time and send clock

The calculated update times are reduction ratios (1, 2, 4, 8, ..., 512) of the send clock. The minimum possible update time thus depends on the minimum send clock of the IO controller that can be set and the efficiency of the IO controller and IO device. Depending on the send clock, it can be that only some of the reduction ratios are available (STEP 7 guarantees this through a pre-selection).

The following tables illustrate the dependency of the update time that can be set on the send clock, using an example of the CPU 1516-3 PN/DP. The update times satisfy the requirements of the PROFINET standard IEC 61158.

Table 3-2 With real-time communication the following applies:

Send clock	Update time	Reduction ratios
250 µs	250 µs to 128 ms	1,2, ..., 512
500 µs	500 µs to 256 ms	1,2, ..., 512
1 ms	1 ms to 512 ms	1,2, ..., 512
2 ms	2 ms to 512 ms	1,2, ..., 256
4 ms	4 ms to 512 ms	1,2, ..., 128

Additional information

For information on real-time communication, refer to the section Real-Time Communication (RT) (Page 198).

3.1.3 PROFINET interface

Overview

PROFINET devices of the SIMATIC product family have one or more PROFINET interfaces (Ethernet controller/interface). The PROFINET interfaces have one or more ports (physical connection options).

In the case of PROFINET interfaces with multiple ports, the devices have an integrated switch.

PROFINET devices with two ports on one interface allow you to configure the system in a line or ring topology. PROFINET devices with three or more ports on one interface are also ideal for setting up tree topologies.

Properties and rules for naming the PROFINET interface and its representation in STEP 7 are explained in the following.

Properties

Every PROFINET device on the network is uniquely identified via its PROFINET interface. For this purpose, each PROFINET interface has:

- A MAC address (factory default)
- An IP address
- A PROFINET device name

Identification and numbering of the interfaces and ports

Interfaces and ports for all modules and devices in the PROFINET system are identified with the following characters:

Table 3-3 Identification for interfaces and ports of PROFINET devices

Element	Symbol	Interface number
Interface	X	In ascending order starting from number 1
Port	P	In ascending order starting from number 1 (for each interface)
Ring port	R	

Examples of identification

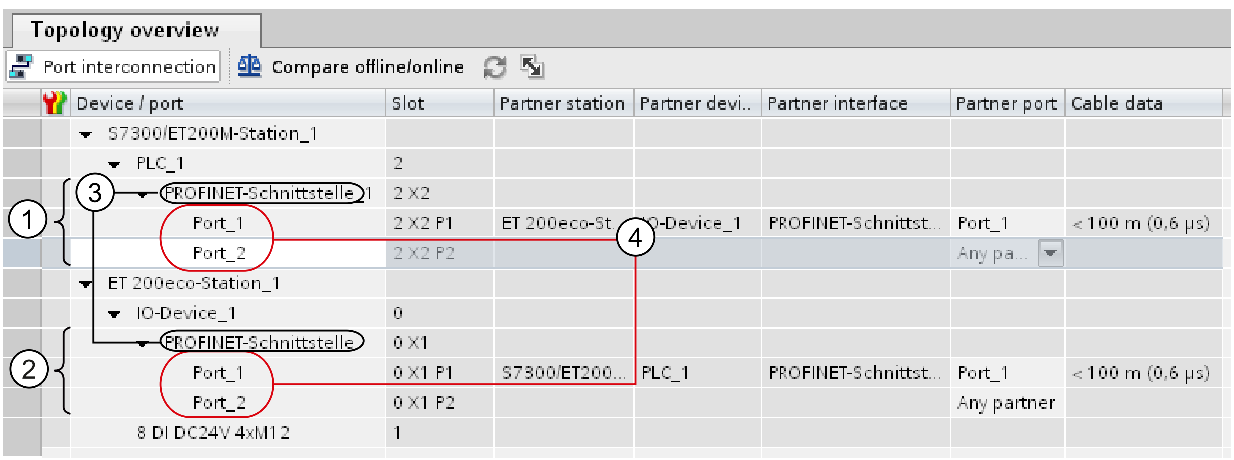
Three examples illustrate the rules for identifying PROFINET interfaces:

Table 3-4 Examples for identifying PROFINET interfaces

Sample labeling	Interface number	Port number
X2 P1	2	1
X1 P2	1	2
X1 P1 R	1	1 (ring port)

Representation of PROFINET Interfaces in the Topology Overview in STEP 7

You can find the PROFINET interface in the topology overview in STEP 7. The PROFINET interface for an IO controller and an IO device is represented as follows in STEP 7:



Number	Description
①	PROFINET interface of an IO controller in STEP 7
②	PROFINET interface of an IO device in STEP 7
③	These lines represent the PROFINET interface.
④	These lines represent the "ports" of a PROFINET interface.

Figure 3-5 Representation of the PROFINET interfaces in STEP 7

Schematic Representation of a PROFINET Interface with Integrated Switch

The following schematic diagram shows the PROFINET interface with integrated switch and its ports for all PROFINETdevices.

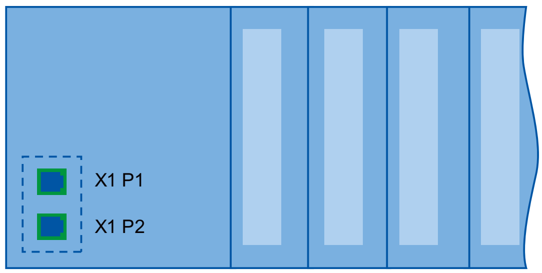


Figure 3-6 PROFINET interface with integrated switch

Functional differences of the PROFINET interfaces

PROFINET interfaces can provide different functions. PROFINET interface functions include identification, configuration, diagnostics and communication services (e.g., open communication). PROFINET interfaces that provide PROFINET IO functions and network security functions are also available.

The following table illustrates the differences using the example of the CPU 1516-3 PN/DP (as of firmware version V2.0), which features two PROFINET interfaces with different functionality.

Table 3-5 Differences between the PROFINET interfaces of the CPU 1516-3 PN/DP (as of firmware version V2.0)

PROFINET interface (X1)	PROFINET interface (X2)
2 ports with PROFINET IO functionality:	**1 port with PROFINET IO functionality:**
Identification, configuration and diagnostics	
PG communication	
HMI communication	
S7 communication	
Time-of-day synchronization	
Web server	
Open communication	
OPC UA server	
IO controller	
I-device	
RT	
IRT	-
Isochronous mode	-
Media redundancy	-
Prioritized startup	-

Additional Information on the Functionality of PROFINET interfaces

You can find information on the number and functionality of the interfaces of a PROFINET device in the documentation for the specificPROFINET device.

PROFINET communication services are described in the Communication function manual (http://support.automation.siemens.com/WW/view/en/59192925).

In the Network security (Page 44) section you can find components that are used to protect networks against hazards.

The Functions (Page 117) section describes the PROFINET IO functions.

3.1.4 Implementation of the PROFINET device model in SIMATIC

Slots and modules

A PROFINET device can have a modular and compact structure. A modular PROFINET device consists of slots into which the modules are inserted. The modules have channels which are used to read and output process signals. A compact device has the same design and can include modules, however, it cannot be physically expanded, which means that no modules can be inserted.

This is illustrated by the following graphic.

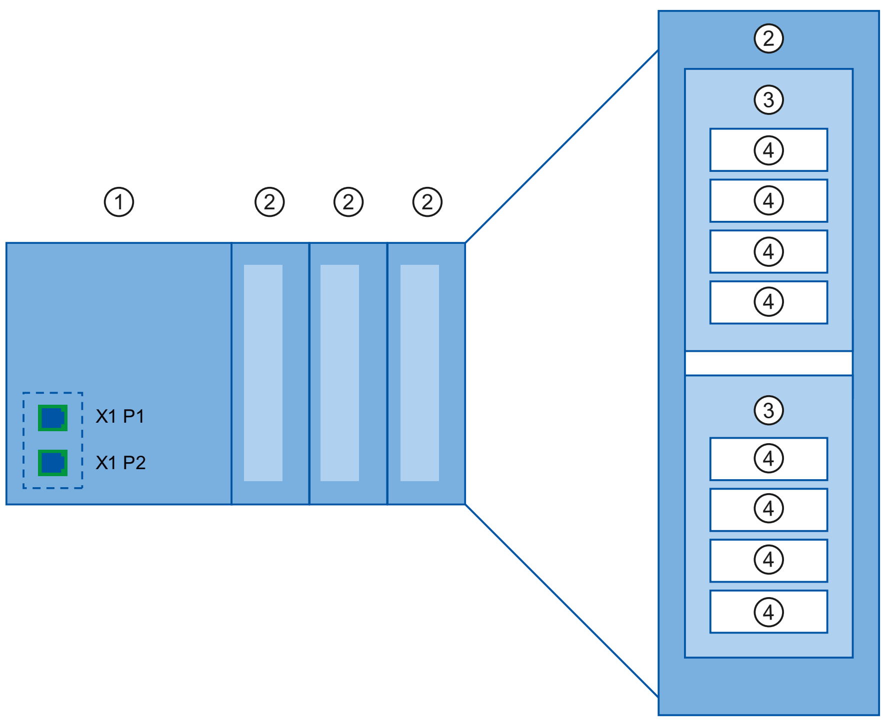


Figure 3-7 Configuration of a PROFINET device

Number	Description
①	Slot with bus interface
②	Slot with module
③	Subslot with submodule
④	Channel

A module can contain multiple submodules.

Representation of PROFINET Device Model in the Device View of STEP 7

The following figure shows the representation of the PROFINET device model in the device view of STEP 7, based on the example of a distributed I/O system ET 200MP:

Device overview

Module	Rack	Slot	I address	Q address	Type
IO-Device_1	0	1			IM 155-5 PN ST
PROFINET-Schnittstelle	0	1 X1			PROFINET interface
Port_1	0	1 X1 P1			Port
Port_2	0	1 X1 P2			Port
DI 32x24VDC HF_1	0	2	16...19		DI32 x 24VDC HF
DQ 32x24VDC/0.5A ST_1	0	3		0...3	DQ32 x 24VDC / 0.5A ST

Figure 3-8 PROFINET device model in the device view of STEP 7

3.2 PROFINET Security Class 1

Introduction

With the introduction of PROFINET Security Class 1, additional security settings have been integrated into the PROFINET communication. They provide for protection of the PROFINET network at the protocol level, for example. Consequently, the individual components are also protected from unauthorized or inadvertent access.

Like in other areas of IT security, PROFINET security is based on the Defense-in-Depth concept. Multiple independent methods provide multi-layered protection against attacks or inadvertent changes.

PROFINET Security Class 1 includes the following security mechanisms for protecting your PROFINET network:

- Configuration options for the SNMP and DCP protocols
- Digital signature of PROFINET GSD files

These introductory sections provide information on which configuration options have been modified for PROFINET Security Class 1.

Simple Network Management Protocol (SNMP)

SNMP is used for simple, network-based device management. For example, PROFINET devices support the following SNMP functions:

- Monitoring and diagnostics of the network topology through network management services
- Provision of network-specific information, such as details about the IP stack

Since STEP 7 V18, SNMP has been disabled in the basic settings of a CPU. You can enable or disable SNMP via the configuration or by writing a data record from the user program. You enter the community strings for read and write access via the configuration.
Starting from STEP 7 V19, you can configure SNMP both for CPUs and for IO devices for the following access types:

- Read and write access
- Read access only

To determine whether IO devices support the new configuration of SNMP, refer to the respective Equipment Manuals. IO devices that support the new configuration of SNMP behave as follows during configuration or initial loading of the hardware configuration:

- STEP 7 as of V19: SNMP is disabled by default. Changed SNMP settings do not take effect until the hardware configuration has been loaded.
- STEP 7 < V19: The IO device adopts the setting "SNMP enabled". When SNMP is enabled, the community strings are set to the following default values:
  - Read-only (read-only community string): public
  - Read and write access (read-write community string): private

Discovery Configuration Protocol (DCP)

The DCP protocol detects PROFINET devices and enables the basic settings, e.g. assignment of IP address or PROFINET device name.
Starting from STEP 7 V19, the "Activate DCP write protection" option is available for DCP. This "PROFINET Security Class 1" function prevents write access to PROFINET devices during IO data exchange and thus, for example, provides protection from the following dangers:

- Takeover of devices by attackers
- Undesired interruption of communication relations (ARs)

If the IO controller and IO device support the write protection function for DCP, this option is enabled by default for the corresponding communication relation. An enabled DCP write protection function takes effect in the following cases:

- On the IO controller side: An IO controller sets up the communication relation with a corresponding IO device (the IO device does not need to be accessible for this purpose).
- On the IO device side: There is an active communication relation between the IO controller and the IO device.

During an effective write protection function of DCP, for example, Set commands (DCP Set) are rejected.
When the write protection option of DCP is enabled, the ongoing operation is secured as follows:

- DCP only has read access to the device information of the IO device.
- Changes to the basic settings of IO devices are no longer possible.

Digital signatures of PROFINET GSD files

The manufacturer provides a general station description file (GSD file) for each PROFINET device. The GSD file describes the properties of the device in a machine readable XML format and contains all properties of a PROFINET-device that are needed for configuring.

With the introduction of PROFINET Security Class 1, GSD files together with other related files, such as images or text files, are provided in a GSDX file with a verified manufacturer signature. A GSDX file can also contain multiple GSD files, such as all GSD files of a device family.

The included signature confirms the integrity and authenticity of the contained files. This ensures that the files contained in the GSDX file originate from the expected manufacturer and have not been altered.

If your engineering system does not support the import of GSDX files, continue using GSD files as up to now.

More information

You can find detailed information on SNMP in the Communication (https://support.industry.siemens.com/cs/us/en/view/59192925) Function Manual.

For information on how to configure SNMP in your PROFINET network, see section Configuring the SNMP (Page 298).

For information on how to configure DCP in your PROFINET network, see section Configuring DCP (Page 301).

3.3 Setting up PROFINET

Contents of this chapter

The following chapter provides background information on building your communication network.

- Overview of the most important passive network components: These are network components that forward a signal without the possibility of actively influencing it, for example, cables, connectors, etc.
- Overview of the most important active network components: These are network components that actively affect a signal, for example switches, routers, etc.
- Overview of the most common network structures (topologies).

Physical connections of industrial networks

The networking of PROFINET devices in industrial systems is generally possible in 2 different physical ways:

- Connected line
  - By means of electrical pulses via copper cables
  - By means of optical pulses via fiber-optic cables
- Wireless via wireless network using electromagnetic waves

PROFINET devices and cabling technology in SIMATIC are suited for industrial use, as they are based on Fast Ethernet and Industrial Ethernet.

- **Fast Ethernet**
  With Fast Ethernet, you transfer data at a speed of 100 Mbps. This transfer technology uses the 100 Base-T standard for this.
- **Industrial Ethernet**
  Structure of Ethernet in industrial environment.
  The biggest difference from standard Ethernet is the mechanical current carrying capacity and noise immunity of the individual components.

3.3.1 Active Network Components

Introduction

The following active network components are available for PROFINET:

- Switch
- Router

Switched Ethernet

PROFINET IO is based on switched Ethernet with full-duplex operation and a bandwidth of 100 Mbps. In this way, the network can be used much more efficiently through the simultaneous data transfer of several devices. The PROFINET IO frames are processed with high priority.

Switches

Switches are network components used to connect several terminal devices or network segments in a local network (LAN).

For the communication of a device with several other devices on PROFINET, the device is connected to the port of a switch. Other communication devices (including switches) can then be connected to the other ports of the switch. The connection between a communication device and the switch is a point-to-point connection.

A switch has the task of receiving and distributing frames. The switch "learns" the Ethernet address(es) of a connected PROFINET device or additional switches and only forwards those frames that are intended for the connected PROFINET device or the connected switch.

Switch variants

Switches are available in two models:

- Integrated into a PROFINET device
  For PROFINET devices with multiple ports (two or more), we are dealing with devices with an integrated switch (for example, CPU 1516-3 PN/DP).
- As autonomous device (for example, switches of the SCALANCE product family)

Selection Guide for Switches

To use PROFINET with the RT class "RT", you can use any switch of "PROFINET Conformance Class A" or higher. All switches of the SCALANCE product family meet these requirements.

If you want to use PROFINET functions that provide an additional value, such as topology recognition, diagnostics, device exchange without exchangeable medium/programming device, you have to use a switch of the "PROFINET Conformance Class B" or higher.

To use PROFINET with the RT class "IRT", you must use a switch of "PROFINET Conformance Class C". With switches of the SCALANCE product family, watch out for the catalog feature "IRT PROFINET IO switch".

To select appropriate switches, we recommend the SIMATIC NET Selection Tool on the Internet (http://support.automation.siemens.com/WW/view/en/39134641).

Switches of the SCALANCE product family

Use the switches of the SCALANCE product family if you want to use the full scope of PROFINET. They are optimized for use in PROFINET IO.

In the SCALANCE X device family, you will find switches with electrical and optical ports and with a combination of both variants. SCALANCE X202-2IRT, for example, has two electrical ports and two optical ports and supports IRT communication.

Beginning with the SCALANCE X200, you can configure, diagnose and address switches of the SCALANCE X device series as PROFINET IO devices using STEP 7.

Router

A router connects separated network segments with each other (e.g. management level and control level). The volume of data volume must be coordinated with the services of the respective network segment. A router also separates two networks and acts as a mediator between both networks. It thus reduces the network load. Routing functionality is provided in the SCALANCE X device family, with SCALANCE X300 or higher.

Communication devices on different sides of a router can only communicate with one another if you have explicitly enabled communication between them via the router.

If you want to access manufacturing data directly from SAP, for example, use a router to connect your Industrial Ethernet in the factory with the Ethernet in your office.

NOTE

If devices need to communicate beyond the limits of a network, you must configure the router so that it allows this communication to take place.

Information on routing with STEP 7 is available in the function manual Communication (http://support.automation.siemens.com/WW/view/en/59192925).

3.3.2 Cabling technology

Cables for PROFINET

Electrical and optical cables are available for PROFINET. The type of cable depends on the data transfer requirements and on the ambient conditions.

Simple method for the prefabrication of twisted pair cables

When you set up your PROFINET system, you can cut the twisted-pair cable to the required length on site, strip it with the *stripping tool* (for Industrial Ethernet), and fit the *Industrial Ethernet Fast Connect RJ45 plugs* using the cut-and-clamp method. For more information on installation, refer to the installation instructions in the "SIMATIC NET Industrial Ethernet Network Manual" (http://support.automation.siemens.com/WW/view/en/8763736).

NOTE

A maximum of four plug-in pairs are allowed between two switches per Ethernet path.

Simple method for the prefabrication of fiber-optic cables

The FastConnect FO cabling system is available for the easy, fast and error-free prefabrication of fiber-optic cables. The glass-fiber optic cable consists of:

- FC FO Termination Kit for SC and BFOC plug (cleave tool, Kevlar scissors, buffer grip, fiber remains container)
- FC BFOC Plug
- FC SC Duplex plug
- FO FC Standard cable
- FO FC Trailing cable

Simple method for the prefabrication of POF and PCF cables

The following special tools provide an easy and safe way to prefabricate POF / PCF cables and fit the SC RJ POF plugs:

- POF cable
  Prefabrication case IE Termination Kit SC RJ POF plug
- PCF cable
  Prefabrication case IE Termination Kit SC RJ PCF plug

Overview of transmission media with PROFINET

The following table summarizes the technical specifications of a PROFINET interface with integrated switch or an external switch, and possible transmission media.

Table 3-6 Transmission media with PROFINET

Physical properties	Connection methods	Cable type / transmission medium standard	Transmission rate / mode	Max. segment length (between two devices)	Advantages
Electrical	RJ45 connector ISO 60603-7	100Base-TX 2x2 twisted, symmetrical and shielded copper cable, CAT 5 transmission requirement IEEE 802.3	100 Mbps, full duplex	100 m	Simple and cheap cable connection
Optical	SCRJ 45 ISO/IEC 61754-24	100Base-FX POF fiber-optic cable (Polymer Optical Fiber, POF) 980/1000 μm (core diameter / external diameter) ISO/IEC 60793-2	100 Mbps, full duplex	50 m	Use when there are large differences in potential Insensitive towards electromagnetic radiation Low line attenuation Considerably longer segments possible[1]
		Plastic-cladded glass fiber (Polymer Cladded Fiber, PCF) 200/230 μm (core diameter / external diameter) ISO/IEC 60793-2	100 Mbps, full duplex	100 m	
	BFOC (Bayonet Fiber Optic Connector) and SC (Subscriber Connector) ISO/IEC 60874	Monomode glass fiber optic cable 10/125 μm (core diameter / external diameter) ISO/IEC 60793-2	100 Mbps, full duplex	26 km	
		Multimode glass fiber optic cable 50/125 μm and 62.5/125 μm (core diameter / external diameter) ISO/IEC 9314-4	100 Mbps, full duplex	3000 m	
Electromagnetic waves	-	IEEE 802.11 x	Depends on the extension used (a, g, h, etc.)	100 m	Greater mobility Cost-effective networking to remote, difficult to access devices

[1] Applies for fiber-optic cables only

See also

PROFINET interface (Page 30)
Assembly Instructions for SIMATIC NET Industrial Ethernet (http://support.automation.siemens.com/WW/view/en/27069465)
PROFINET Installation Guideline (http://www.profibus.com/nc/download/installation-guide/downloads/profinet-installation-guide/display/)

3.3.3 Wireless design

3.3.3.1 Basics

What is Industrial Wireless LAN?

In addition to data communication in accordance with the IEEE 802.11 standard, the SIMATIC NET Industrial Wireless LAN provides a number of enhancements which offer significant benefits for industrial customers. IWLAN is particularly suitable for demanding industrial applications that require reliable wireless communication. This is supported by the following properties:

- Automatic roaming when the connection to Industrial Ethernet is interrupted (Forced Roaming)
- Cost savings generated by using a single wireless network for reliable operation of a process with both process-critical data (alarm message, for example) and non-critical communication (service and diagnostics, for example)
- Cost-effective connection to devices in remote environments that are difficult to access
- Predictable data traffic (deterministic) and defined response times
- Cyclical monitoring of the wireless link (link check)

Objectives and advantages of Industrial Wireless LAN

Wireless data transmission achieves the following objectives:

- Seamless integration of PROFINET devices into the existing bus system via the wireless interface
- Mobile use of PROFINET devices for different production-linked tasks
- Flexible configuration of the system components for fast development in accordance with customer requirements
- Maintenance costs are minimized by savings in cables

Application examples

- Communication with mobile subscribers (mobile controllers and devices, for example), conveyor lines, production belts, translation stages , and rotating machines
- Wireless coupling of communication segments for fast commissioning or cost-effective networking where routing of wires is extremely expensive (e.g. public streets, railroad lines)
- Stacker trucks, automated guided vehicle systems and suspended monorail systems

The following graphic illustrates the many possible applications and configurations for SIMATIC device family wireless networks.

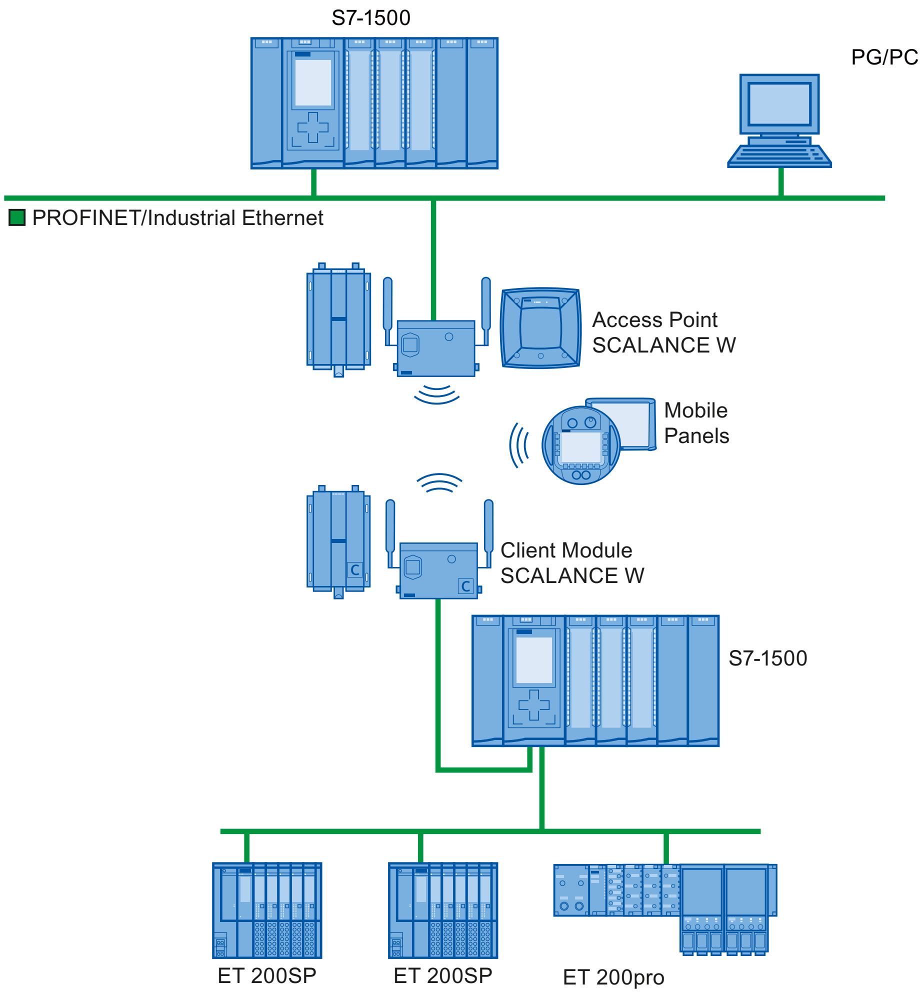


Figure 3-9 Application example for the use of Industrial Wireless LAN

Data transmission rate

In Industrial Wireless LAN, gross data transmission rates of 11 Mbps or 54 Mbps without full duplex are permitted.

Range

With SCALANCE W (access points), wireless networks can be set up indoors and outdoors. Multiple access points can be installed to create large wireless networks in which mobile subscribers are transferred seamlessly from one access point to another (roaming).

As an alternative to a wireless network, point-to-point connections of Industrial Ethernet segments can also be set up over large distances (several hundred meters). In this case, the range and characteristics of the RF field are determined by the antennas used.

NOTE

Range

The range can be considerably less, depending on spatial factors, the wireless standard used, the data rate, and the antennas on the send and receive sides.

3.3.3.2 Tips on assembly

Wireless networks, SCALANCE device family

With PROFINET, you can also set up wireless networks with Industrial Wireless Local Area Network (IWLAN) technology. We recommend implementing the SCALANCE W device line for this.

Update time in STEP 7

If you set up PROFINET with Industrial Wireless LAN, you may have to increase the update time for the wireless devices. The IWLAN interface provides lower performance than the wired data network: Several communication stations have to share the limited transmission bandwidth. For wired solutions, 100 Mbps is available for each communication device.

The Update time parameter can be found in the "Realtime settings" section in the Inspector window of IO devices in STEP 7.

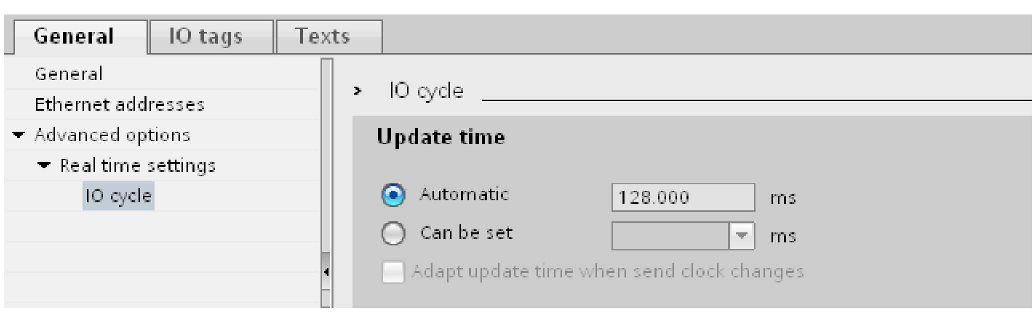


Figure 3-10 Update time in STEP 7

Additional information

More information about SCALANCE W Industrial Wireless LAN components can be found in the manual SIMATIC NET SCALANCE W-700 (http://support.automation.siemens.com/WW/view/en/42784493).

More information about wired data transmission can be found in the manual SIMATIC NET Twisted Pair and Fiber Optic Networks (http://support.automation.siemens.com/WW/view/en/8763736).

More information about wireless data transmission can be found in the manual Basics for configuring an industrial wireless LAN (http://support.automation.siemens.com/WW/view/en/9975764).

You should also read the PROFINET installation guideline of the PROFIBUS User Organization on the Internet (http://www.profibus.com/nc/download/installation-guide/downloads/profinet-installation-guide/display/). Various documents that assist with the setting up of your PROFINET automation solution are available here:

- PROFINET planning guideline
- PROFINET installation guideline
- PROFINET commissioning guideline
- Additional documents for setup of PROFINET

3.3.4 Network security

3.3.4.1 Basics

Introduction

The topic of data security and access protection (Security) has become increasingly important in the industrial environment. The increased networking of entire industrial systems, vertical integration and networking of levels within a company and new techniques such as remote maintenance all result in higher requirements for protecting the industrial plant.

Data security solutions for office environments cannot simply be transferred one-to-one to industrial applications to protect against manipulation in sensitive systems and production networks.

Requirements

Additional security requirements arise from the specific communication requirements in the industrial environment (real-time communication, for example):

- Protection against interaction between automated cells
- Protection of network segments
- Protection against faulty and unauthorized access
- Scalability of network security
- Must not influence the network structure

Definition of security

Generic term for all the measures taken to protect against:

- Loss of confidentiality due to unauthorized access to data
- Loss of integrity due to manipulation of data
- Loss of availability due to destruction of data, for example, through faulty configuration and denial-of-service attacks

Threats

Threats can arise from external and internal manipulation. The loss of data security is not always caused by intentional actions.
Internal threats can arise due to:

- Technical errors
- Operator errors
- Defective programs

Added to these internal threats there are also external ones. The external threats are not really any different to the known threats in the office environment:

- Software viruses and worms
- Trojans
- Man-in-the-middle attacks
- Password Phishing
- Denial of Service

Protective measures

The most important precautions to prevent manipulation and loss of data security in the industrial environment are:

- Physical access protection to the devices
- Filtering and control of data traffic by means of firewall
- A virtual private network (VPN) is used to exchange private data on a public network (Internet, for example).
  The most common VPN technology is IPsec. IPsec (Internet Protocol Security) is a collection of security protocols that are used as the basis for the IP protocol at the mediation level and allow a secured communication via potentially unsecure IP networks.
- Segmenting in protected automation cells
  This concept has the aim of protecting the lower-level network devices by means of security modules. A group of protected devices forms a protected automation cell.
- Authentication (identification) of the devices
  The security modules identify each other over a safe (encrypted) channel using authentication procedures. It is therefore impossible for unauthorized parties to access a protected segment.
- Encrypting the data traffic
  The confidentiality of data is ensured by encrypting the data traffic. Each security module is given a VPN certificate which includes the encryption key.

3.3.4.2 Network components and software

Protection against unauthorized access

The following solutions may be used to connect industrial networks to the intranet and Internet to protect against internal and external threats:

- Communication processors, such as the SIMATIC CP 1543-1
- SCALANCE X-300 and SCALANCE S - the data security components of the SIMATIC NET product family
- SOFTNET security client for use on PCs

Features

Both of these products have a wide variety of features, such as:

- Easy integration of existing networks without configuration, with integrated firewall.
- Segmenting in protected automation cells
- Authentication (identification) of the devices
- Encrypting the data traffic

3.3.4.3 Application example

Data security at the office and production levels

The following graphic contains an application example with protected areas at different levels of the company created using SCALANCE S and the security client. The protected areas are highlighted in light gray.

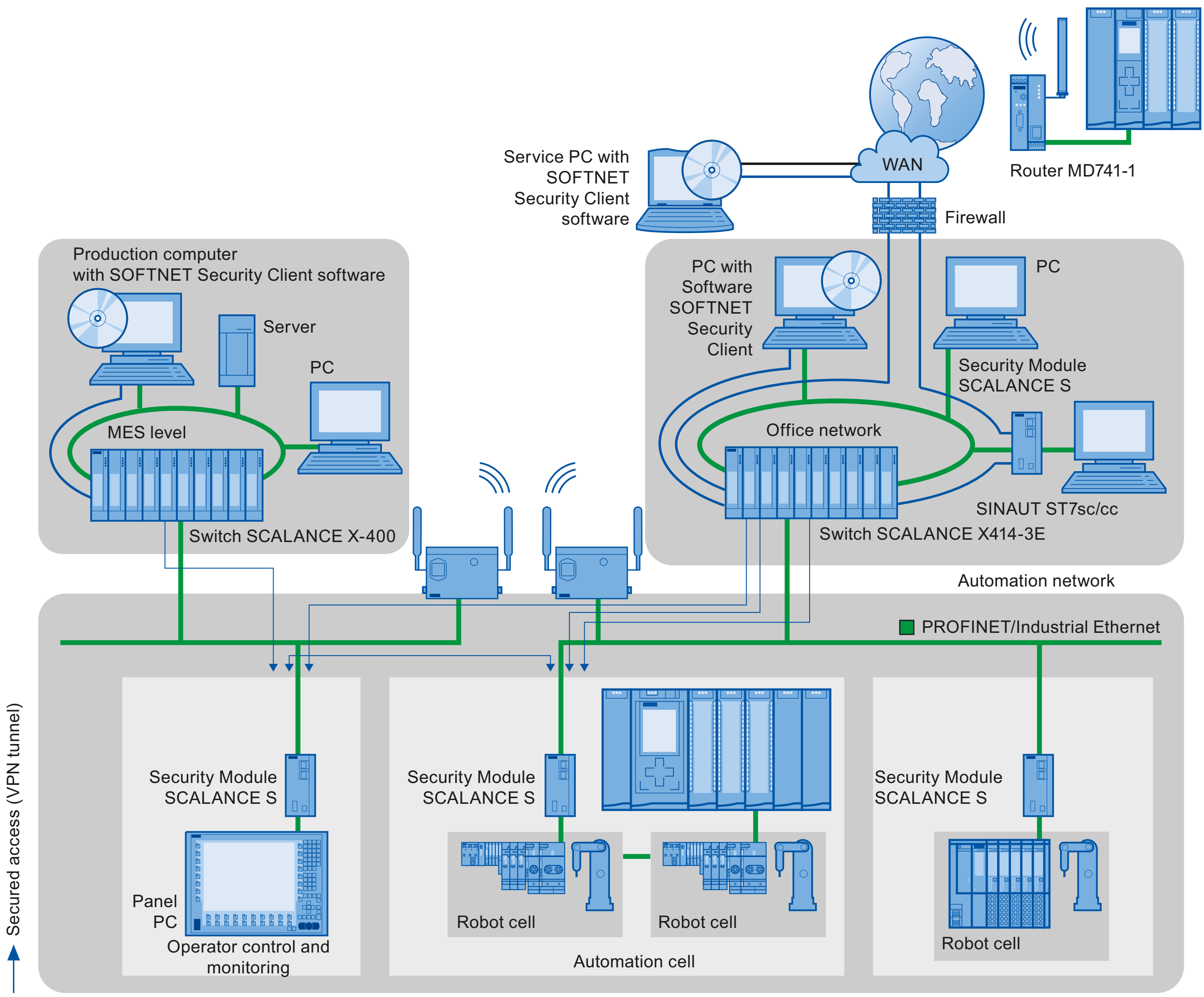


Figure 3-11 Network configuration with the SCALANCE S security module and the SOFTNET security client

Additional information

Additional information on the configuration of a security standard in PROFINET, is available:

- In the PROFINET security guideline. These guidelines can be found on the homepage of the PROFIBUS user organization on the Internet (http://www.profinet.com).
- In the Industrial Ethernet Security (http://support.automation.siemens.com/WW/view/en/56577508) manual
- In the SCALANCE S and SOFTNET Security Client (http://support.automation.siemens.com/WW/view/en/21718449) manual

You can find general information on industrial security concepts, functions and news on the Industrial Security website (http://www.siemens.com/industrialsecurity).

4 Parameter assignment/addressing

To set up an automation system, you will need to configure, assign parameters and interlink the individual hardware components. In STEP 7, the work needed for this is undertaken in the device, topology and network view.

Configuring

"Configuring" is understood to mean arranging, setting and networking devices and modules within the device, topology or network view.

An I/O address is automatically assigned to each module. The I/O addresses can be subsequently modified.

The CPU compares the configuration preset in STEP 7 with the actual current configuration of the system. In this way, potential errors can be detected and reported straight away.

The exact procedure for configuring devices is described in detail in the STEP 7 online help.

Assigning parameters

"Assigning parameters" is understood to mean setting the properties of the components used. The settings for the hardware components and for data communication are configured at the same time.

In STEP 7, you can "assign parameters" for the following settings PROFINET:

- Device names and IP address parameters
- Port interconnection and topology
- Module properties / parameters

The parameters are loaded into the CPU and transferred to the corresponding modules when the CPU starts up. Modules are easy to replace from spare parts, as the parameters assigned for the SIMATIC CPUs are automatically loaded into the new module at each startup.

Adjusting the hardware to the project requirements

You need to configure hardware if you want to set up, expand or change an automation project. To do this, add hardware components to your structure, link these with existing components, and adapt the hardware properties to the tasks.

The properties of the automation systems and modules are preset such that in many cases they do not have to be assigned parameters again.

Parameter assignment is however needed in the following cases:

- You want to change the default parameter settings of a module.
- You want to use special functions.
- You want to configure communication connections.

4.1 Assigning an IO device to an IO controller

PROFINET IO System

A PROFINET IO system is comprised of a PROFINET IO controller and its assigned PROFINET IO devices. After these devices have been placed in the network or topology view, STEP 7 assigns default values for them. Initially, you only have to worry about the assignment of IO devices to an IO controller.

Requirement

- You are in the network view of STEP 7.
- A CPU has been placed (e.g., CPU 1516-3 PN/DP).
- An IO device has been placed (e.g., IM 155-6 PN ST).

Procedure

To assign IO devices to an IO controller, proceed as follows:

1. Place the pointer of the mouse over the interface of the IO device.
2. Press and hold down the left mouse button.
3. Move the pointer.
   The pointer now uses the networking symbol to indicate "Networking" mode. At the same time, you see the lock symbol on the pointer. The lock symbol disappears only when the pointer is over a valid target position.
4. Now move the pointer onto the interface of the IO controller. You can keep the left mouse button pressed or release it when performing this action.
5. Now release the left mouse button or press it again (depending on your previous action).

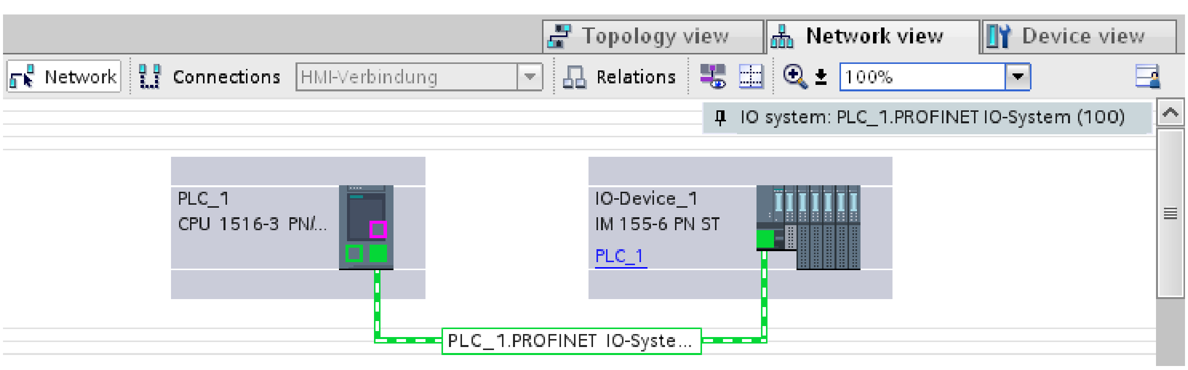


Figure 4-1 Assigning an IO device to an IO controller in the network view of STEP 7

Result

You have assigned an IO device to an IO controller.

Checking the assignment

You can find an overview of the communication relationships in the "IO communication" tab in the tabular area of the network view. This table is context-sensitive for selection in the graphic area:

- Selection of the interface shows the I/O communication of the respective interface.
- Selection of the CPU shows all I/O communication of the CPU (including PROFIBUS).
- Selection of the station (as in the above figure) interface shows the I/O communication of the complete station.

Information on S7-1500R/H

You can find the procedure for assigning an IO device with system redundancy S2 to the redundant S7-1500R/H system in section Assigning IO device to the redundant S7-1500R/H system (Page 324).

4.2 Device name and IP address

Introduction

In order that the PROFINET device can be addressed as node on PROFINET, the following are required:

- A unique PROFINET device name
- A unique IP address in the relevant IP subnet

STEP 7 assigns a device name during the arrangement of a PROFINET device in the hardware and network editor. The IP addresses are typically assigned automatically by STEP 7 and assigned to the devices based on the device name.

You can change the name and IP address manually.

In STEP 7

You can find the device name and the IP address under "Ethernet addresses" in the properties of the PROFINET interface in the Inspector window.

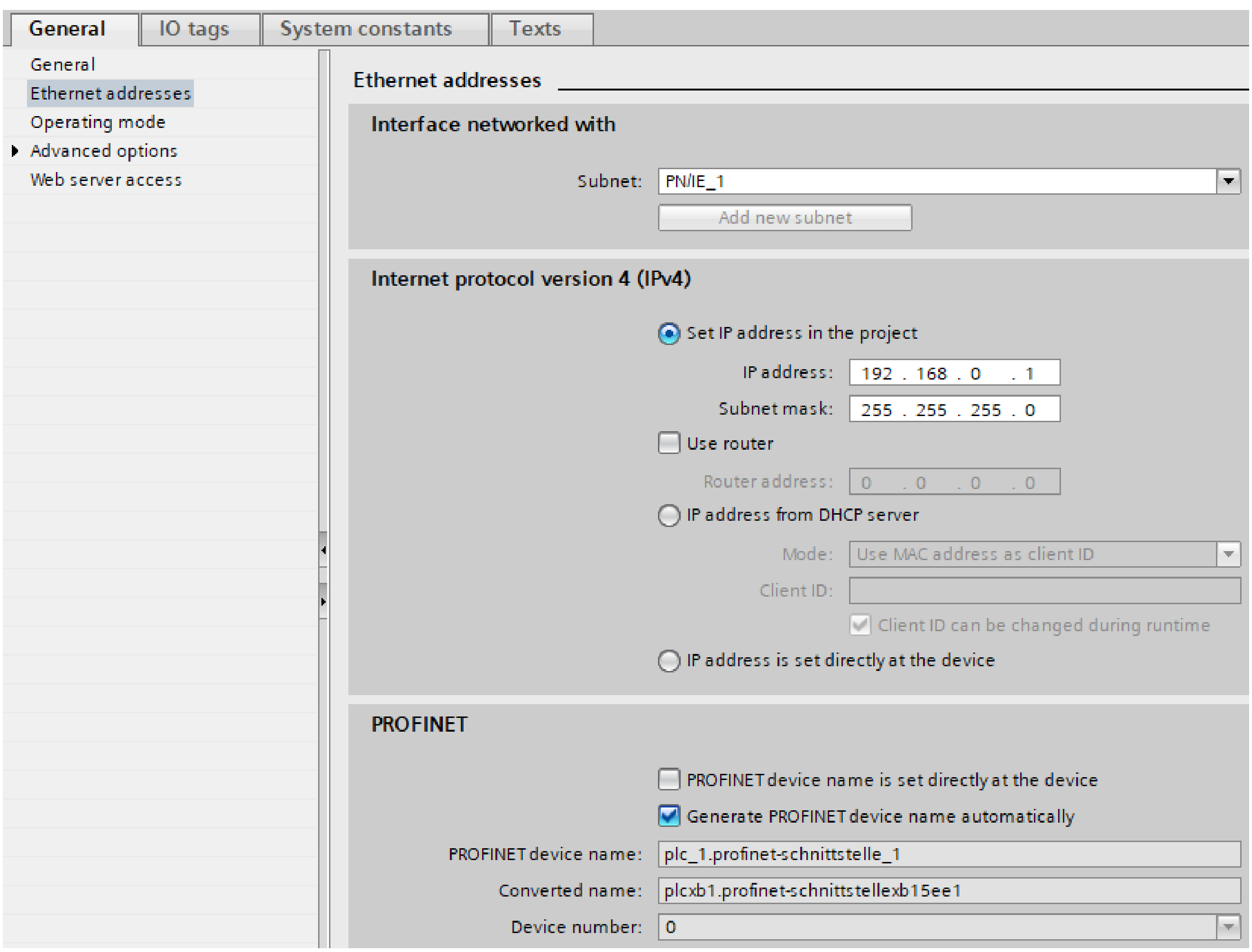


Figure 4-2 Device name and IP address in STEP 7

The function, the assignment and the changing of the device name and the IP address are described in the following sections.

4.2.1 Device name

Device names

Before an IO device can be addressed by an IO controller, it must have a device name. In PROFINET, this method was selected because it is simpler to work with names than with complex IP addresses.

The assignment of a device name for a specific IO device can be compared to setting the PROFIBUS address of a DP slave.

In delivery state, an IO device does not have a device name. A device name must first be assigned before an IO device can be addressed by an IO controller, for example, for transferring configuration data during startup or for exchanging user data in cyclic mode. You assign the device names to the IO device, for example, with the programming device / PC.

IO devices that have a slot for removable storage media provide the option of writing the device name directly to the removable storage medium in the programming device.

When a device is replaced by a device without removable medium, the IO controller assigns the device name based on topological configuration (see section Configuring topology (Page 72)).

Structured device names

The device name is automatically assigned by default for PROFINET devices S7-1200, S7-1500, ET 200MP, ET 200SP and ET 200AL when these are configured in STEP 7. The device names are formed from the name of the CPU or the name of the interface module. For devices with several PROFINET interfaces, the name of the interface is enhanced, for example, "plc_1.profinet-interface_2" or "io-device_1".

You can structure the device names using DNS conventions.

These conventions are defined by "Internationalizing Domain Names in Applications (IDNA). According to this, device names are written in lower case.

The "Domain Name System" (DNS) is a distributed database (http://iana.org), which manages the name space on the Internet. To structure the names, you use the dot ("."). The hierarchy is shown in ascending order from left to right.

...<Subdomain name>.<Domain name>.<Top-level domain name>

If the name is not DNS-compliant, the name will be converted by STEP 7, for example, to "plcxb1.profinet-schnittstellexb2022c" or "io-devicexb15b32".

Device number

In addition to the device name, STEP 7 also assigns a device number beginning with "1" when an IO device is allocated.
The device number is located in the Inspector window in the properties of the PROFINET interface, under "Ethernet addresses" in the area PROFINET.

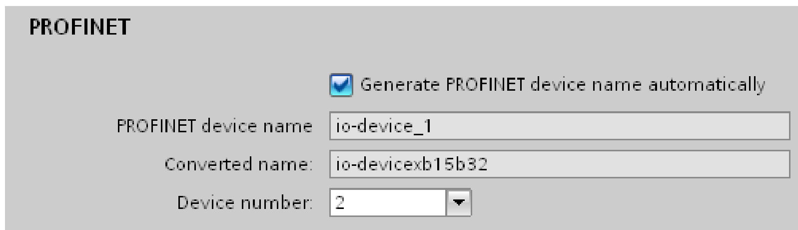


Figure 4-3 Device number

This device number can be used to identify an IO device in the user program (for example, with the instruction "LOG2GEO").

4.2.2 IP address

IP address

To allow a PROFINET device to be addressed as a device on Industrial Ethernet, this device also requires an IP address that is unique within the network. The IP addresses are typically assigned automatically by STEP 7 and assigned to the devices based on the device name. If it is a standalone network, you can apply the IP address and subnet mask suggested by STEP 7. If the network is part of an existing Ethernet company network, obtain the information from your network administrator.

Configuration of the IP address

In accordance with Internet Protocol version 4 (IPv4), the IP address is made up of four decimal numbers with a range of values from 0 through 255. The decimal numbers are separated by periods (for example, 192.162.0.0).
The IP address consists of the following:

- Address of the network
- Address of the device (PROFINET interface of the IO controller/IO device)

Generating IP addresses

You assign the IP addresses of the IO devices in STEP 7. The IO devices receive their IP addresses during parameter assignment by the IO controller.

In addition, for some IO devices (e.g., SCALANCE X, S7-300 CPs), it is possible not to obtain the IP address during startup of the IO controller, but rather to set it beforehand on the device (see Permitting changes to the device name and IP address directly on the device (Page 66)).

The IP addresses of the IO devices always have the same subnet mask as the IO controller and are assigned from the IP address of the IO controller in ascending order. The IP address can be changed manually, if necessary.

For devices with several PROFINET interfaces (e.g., CPU 1516-3 PN/DP), the IP addresses must be located in different subnets.

NOTE

Assignment of the IP address by a DHCP server

As of STEP 7 V17, you can select the option "IP address from DHCP server" for the PROFINET interface. With this option, the PROFINET interface is assigned an IP address by an DHCP server.

When you configure the option "IP address from DHCP server" for a PROFINET interface, this interface does no longer support the PROFINET IO functionality.

For more information on addressing by a DHCP server, please refer to the Communication (http://support.automation.siemens.com/WW/view/en/59192925) function manual.

Default router

The default router is used when data has to be forwarded via TCP/IP or UDP to a partner located outside the local network.

In STEP 7, the default router is named Router. You can activate the use of a router in the Inspector window of a CPU with the "Use router" check box in the "IP protocol" section. STEP 7 assigns the local IP address to the default router by default.

The router address that is set on the PROFINET interface of the IO controller is automatically transferred for the configured IO devices.

Subnet mask

The bits set in the subnet mask decide the part of the IP address that contains the address of the network.

In general, the following applies:

- The network address is obtained from the AND operation of the IP address and subnet mask.
- The device address is obtained from the AND NOT operation of the IP address and subnet mask.

Example of the subnet mask

Subnet mask: 255.255.0.0 (decimal) = 11111111.11111111.00000000.00000000 (binary)
IP address: 192.168.0.2 (decimal) = 11000000.10101000.00000000.00000010 (binary)
Meaning: The first 2 bytes of the IP address determine the network - i.e., 192.168. The last two bytes address the device, i.e. 0.2.

Relation between IP address and default subnet mask

An agreement exists relating to the assignment of IP address ranges and so-called "Default subnet masks". The first decimal number (from the left) in the IP address determines the structure of the default subnet mask with respect to the number of "1" values (binary) as follows:

IP address (decimal)	IP address (binary)	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

Range of values for the first decimal point

A value between 224 and 255 is also possible for the first decimal number of the IP address (address class D etc.). However, this is not recommended because there is no address check for these values.

Masking other subnets

You can use the subnet mask to add further structures and form "private" subnets for a subnet that is assigned one of the address classes A, B or C. This is done by setting other, less significant bits of the subnet mask to "1". For each bit set to "1", the number of "private" networks doubles and the number of devices they contain is halved. Externally, the network continues to function as an individual network.

Example:

You have a subnet of address class B (for example, IP address 129.80.xxx.xxx) and 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 devices with addresses from 129.80.001.000 to 129.80.127.254 are located in a subnet, all devices with addresses from 129.80.128.000 to 129.80.255.254 in a different subnet.

Reading out an IP address in the user program

You can read out the IP address of a PROFINET device in the user program of a S7-1500 CPU. You can find information in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/82947835).

4.2.3 Assigning a device name and IP address

Assigning an IP address and subnet mask for an IO controller for the first time

You have the following options:

- **Using a programming device or PC:**
  Connect your programming device/PC to the same network as the relevant PROFINET device. The interface of the programming device/PC must be set to TCP/IP mode. During the download, first of all display all available devices via the "Accessible devices" download dialog box. Select the target device via its MAC address and then assign its IP address before you download the hardware configuration including the configured IP address (IP address is saved retentively).
- **Using the display of a S7-1500 CPU:**
  The S7-1500 CPUs have a front cover with a display and operating keys. You can use this display to assign or change the IP address. To set the IP address, navigate on the display via the menu items "Settings" > "Addresses" > "X1 (IE/PN)" > "Parameters".
- **Using a memory card:**
  If your PROFINET device is equipped for a memory card (Micro Memory Card/SIMATIC memory card), plug this into your programming device/PC and save the hardware configuration together with the configured IP address on this memory card. Then plug the memory card into the PROFINET device. Once inserted, the PROFINET device automatically applies the IP address.
  If you have saved a configuration to the memory card with the "IP address is set directly at the device" option, you must assign the IP address using a different method after inserting the memory card (see section Permitting changes to the device name and IP address directly on the device (Page 66)).

Assigning device names and IP address for "Device replacement without exchangeable medium/programming device"

For devices without exchangeable medium (e.g., ET 200MP, ET 200SP) and devices that support "Device replacement without exchangeable medium/PG" (e.g., ET 200S), the IO controller can identify the device without name from the neighbor relationships specified by the set topology and from actual neighbor relationships determined by the real PROFINET devices. The IO controller then assigns the PROFINET device the configured name and incorporates the PROFINET device in the user data traffic. (See also Device replacement without exchangeable medium (Page 243)).

IP address assignment when replacing IO devices with exchangeable mediumPG

The following is contained on the memory card of the programmable logic controller:

- On the IO controller: Device name and IP address
- On the IO device: Device name

When you remove the memory card from a PROFINET device and insert it in another PROFINET device with exchangeable medium (e.g., ET 200S), device-specific information and the device name are loaded to the device.

If an IO device has to be replaced in its entirety due to a device or module defect, the IO controller automatically assigns parameters and configures the replaced device or module. The cyclic exchange of user data is then restarted. In addition to this, before the power ON of the IO device, the memory card with the valid name must be removed from the faulty IO device and added to the replaced device.

In the event of an error in the PROFINET device, the memory card allows you to replace a module without a programming device/PC. You can also transfer the device data directly from the programming device/PC to the memory card.

Procedure: Changing the device name using properties of the PROFINET interface

You can change the PROFINET name via the properties of the PROFINET interface. This is useful when the PROFINET device has not received its previous name from the automatic generation, for example, in the case of a migration.

1. In the network or device view of the STEP 7 hardware and network editor, select the PROFINET interface of a PROFINET device.
2. In the Inspector window, go to "Ethernet addresses" in the PROFINET area.
3. Clear the "Generate PROFINET device name automatically" check box.
4. Enter the new PROFINET device name in the relevant field.

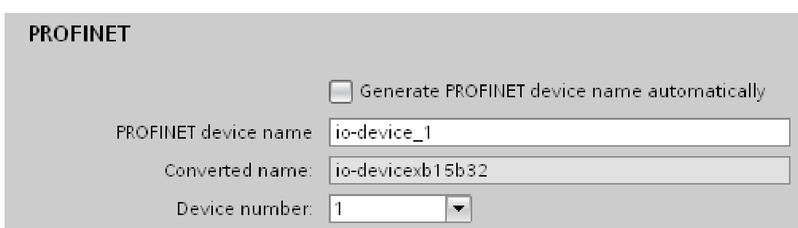


Figure 4-4 Changing the device name of a PROFINET device in the properties

Alternative procedure: Changing the device name of a PROFINET device in the network view

Requirement: The "Generate PROFINET device name automatically" check box is selected.

1. In STEP 7, select the "Network overview" tab in the tabular area of the network view.
2. In the "Device" column, overwrite the name in the row of the relevant PROFINET device.

The name is also changed accordingly in the graphic area of the network view.

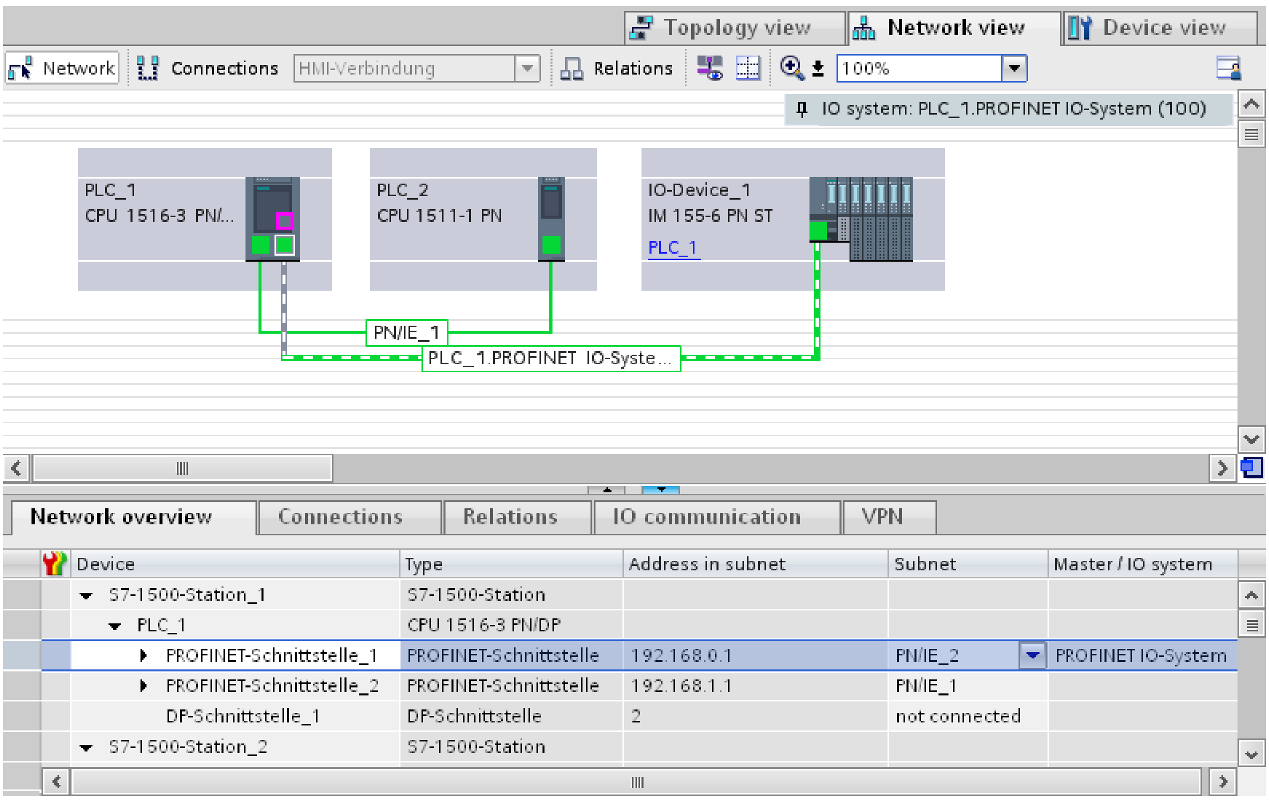


Figure 4-5 Changing the device name of a PROFINET device in STEP 7

Procedure: Changing the IP address

To change the IP address, follow these steps:

1. In the network or device view of the STEP 7 hardware and network editor, select the PROFINET interface of a PROFINET device.
2. In the Inspector window, go to "Ethernet addresses" in the "IP protocol" area.
3. Check that the option "Set IP address in the project" is selected.
4. Enter the new IP address in the relevant field.

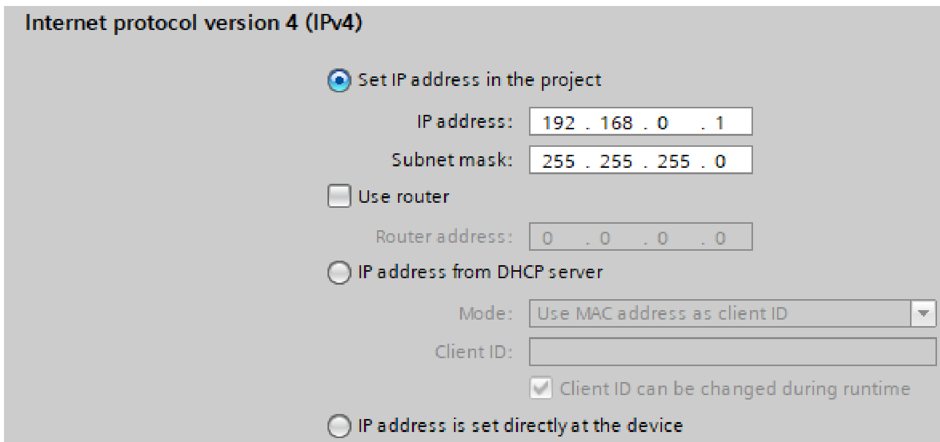


Figure 4-6 Changing the IP address of a PROFINET device in STEP 7

Downloading configured device name to IO device

To load the configured device names to the IO device, follow these steps:

1. Connect your programming device/PC to the same network as the relevant IO device. The interface of the programming device/PC must be set to TCP/IP mode.
2. In STEP 7, select the relevant IO device in the "Accessible devices" dialog based on the MAC address.
3. Click "Assign name" to download the configured device name to the IO device.
   The IO controller recognizes the IO device automatically by its device name and automatically assigns the configured IP address to it.

Identification of the PROFINET device

To clearly identify a device from several identical devices in a control cabinet, for example, you can flash the link LED of the PROFINET device.

To do this, select the menu command **Online > Accessible devices...** in STEP 7. In the "Accessible devices" dialog, set the "PG/PC" interface by means of which you are connected to the devices. STEP 7 now automatically searches for the accessible devices and displays them in the "Accessible devices in target subnet" table. Select the desired PROFINET device and click on the "Flash LED" button. The PROFINET device is identified based on its MAC address.

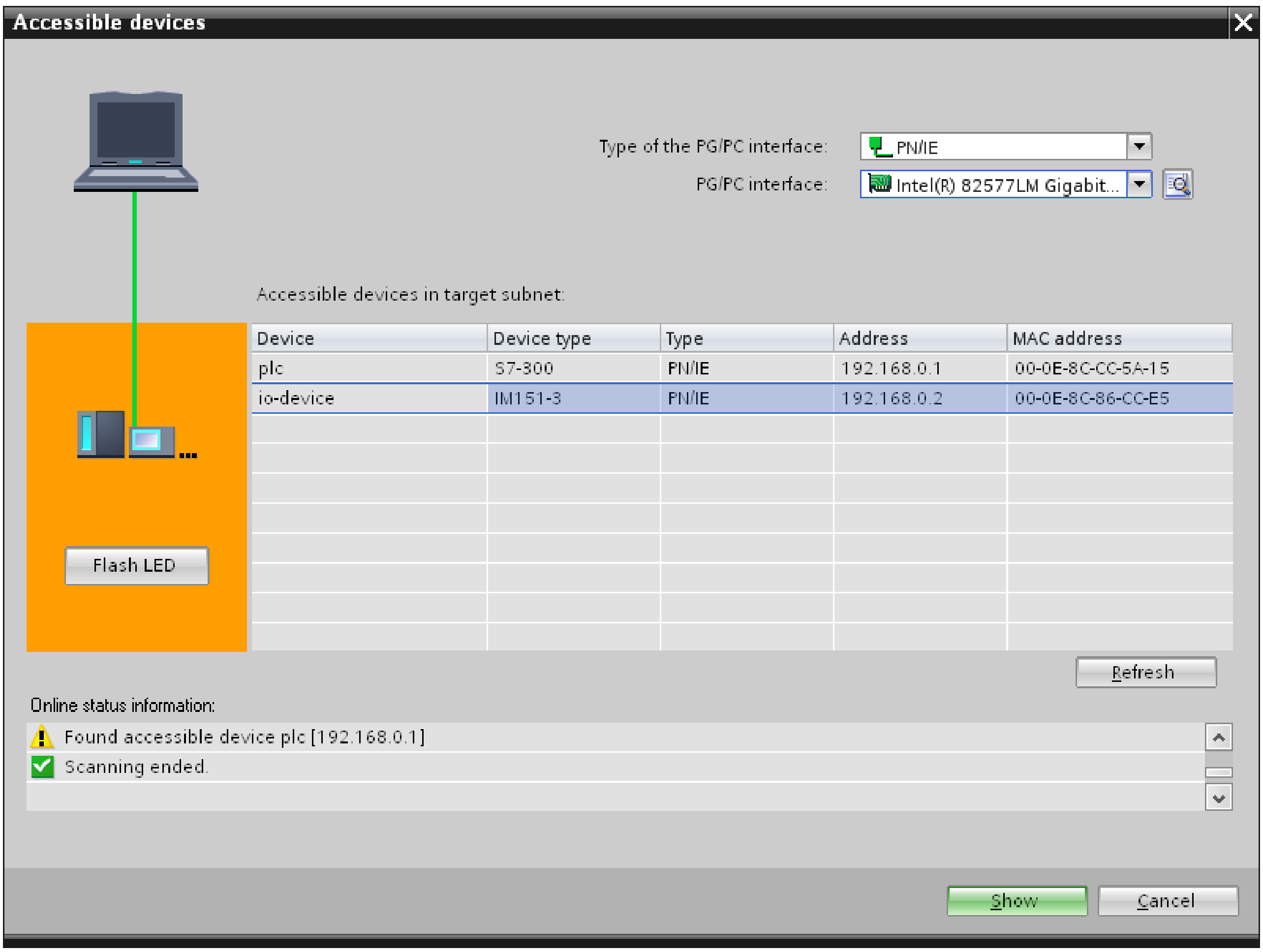


Figure 4-7 "Accessible devices" dialog

Using an different way to assign IP addresses for IO devices

Various IO devices, for example, SCALANCE X,S7-300 CPs, support the option of not having the IP addresses assigned by the IO controller during startup. In this case, the IP address is assigned in a different way. For additional information, refer to the manual of the respective PROFINET device of the SIMATIC device family.

Additional information

You can find a detailed description of the operation and functions of the display of the S7-1500 CPUs in the system manual S7-1500, ET 200MP (http://support.automation.siemens.com/WW/view/en/59191792).

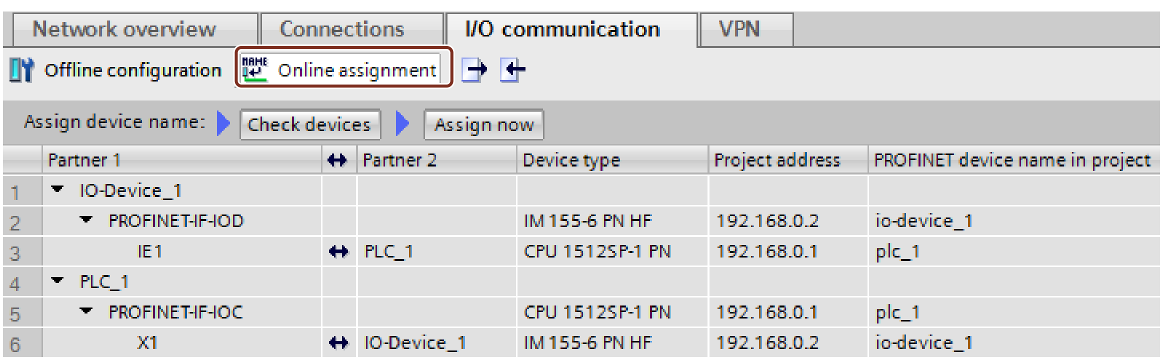
4.2.4 Assign device name via communication table

Introduction

You can assign the device names of PROFINET IO devices configured offline to the devices online. You can do this in the table area of the network view in the table "I/O communication". You can also assign the device names to several devices at the same time.

"Online assignment" tab

In the I/O communication table, you will find the tabs "Offline configuration" and "Online assignment". In the "Online assignment" tab, you can assign the PROFINET device names that were assigned offline to the corresponding IO devices online. To do this, use the buttons "Check devices" and "Assign now".



	Partner 1	↔	Partner 2	Device type	Project address	PROFINET device name in project
1	IO-Device_1					
2	PROFINET-IF-IOD			IM 155-6 PN HF	192.168.0.2	io-device_1
3	IE1	↔	PLC_1	CPU 1512SP-1 PN	192.168.0.1	plc_1
4	PLC_1					
5	PROFINET-IF-IOC			CPU 1512SP-1 PN	192.168.0.1	plc_1
6	X1	↔	IO-Device_1	IM 155-6 PN HF	192.168.0.2	io-device_1

Figure 4-8 Assign device name via communication table

The objects displayed in the table of the "Online assignment" tab depend on the setting of the filter function. If only selected objects should be displayed, only objects of the corresponding context are displayed depending on the selection in the network view.

- PROFINET subnet: All connected devices and their PROFINET interfaces
- IO system All devices involved and their PROFINET interfaces
- Sync domain: All devices involved and their PROFINET interfaces
- Devices: The device and any existing PROFINET interfaces
- Other subnets or interfaces such as MPI or PROFIBUS are not displayed

If the display is set for all devices using the filter function, all devices are displayed that have a PROFINET interface, regardless of whether they are connected via a PROFINET subnet or are part of an IO system. Devices without a PROFINET interface, for example only with a DP or MPI interface, are not displayed.

General procedure

To assign PROFINET device names, you must first detect the IO devices available online. With this procedure, it matters whether the MAC addresses are known or unknown. This results in a general procedure in two steps:

1. Detecting the IO devices available online
2. Assigning configured PROFINET device names to the IO devices available online

Requirements

- You are in the network view.
- There is an online connection to the devices.

Procedure (step 1)

To detect IO devices available online from the I/O communication table, follow these steps:

1. Optional: Entered known MAC addresses in the "MAC address" column. After every valid entry, the check box under "Assign device" is selected for the relevant row.

   NOTE

   You can enter, insert or import the MAC address in different formats. The correct format is automatically entered in the cell. The following entries are supported and then converted to the required format:

   – "08:00:06:BA:1F:20"
   – "08 00 06 BA 1F 20"
   – "080006BA1F20"

   The formats used in the example are automatically converted to "08-00-06-BA-1F-20".

2. Click "Check devices" to start the check of the IO devices available online.
3. Set the PG/PC interface in the dialog window and click "Start".

Intermediate result

After the check, the result is displayed for every device in the table. Online data found is automatically entered in the table and the check box "Assign device" is set to "checked" in the rows in which a MAC address was entered or found online. The result of the check is shown as an icon in the "Status" column.

Status	Meaning
	Matching device and compatible type
	Matching device and incompatible type
	Non-matching device
	Device cannot be reached (with a known MAC address)
	Ready for assignment (with known MAC address)

NOTE

The icon "Ready for assignment" appears when a MAC address exists and matching device data was found, but no PROFINET device name was found online.

You can update the data of the detected devices again via their MAC addresses at any time. To do this, you specify the MAC address and the status of the device is displayed immediately without having to re-detect the device.

Procedure (step 2)

All PROFINET device names configured offline will be assigned to the devices available online in a bulk operation.

1. Click the "Assign now" button.

   NOTE

   The bulk operation cannot be reversed. A message to this effect appears in a dialog window.

2. Click "Start" in the dialog window to start the assignment of the PROFINET device names.

Result

The PROFINET device names configured offline will be assigned to the devices available online. This relates to devices in whose row the check box under "Assign device" is selected, that have a MAC address and have the status "Ready for assignment".

Importing and exporting data

Using the import and export button, you can import or export the data of the I/O communication table for the online assignment:

- When you export, the currently displayed data of the table is exported to a CSV file. Using the filter function of the table, you can select which data will be exported.
- When you import, the data of the CSV file is written to the table. If there are conflicts with values already existing in the table, you can decide whether the data should be overwritten or whether the import needs to be stopped.

4.2.5 Permitting changes to the device name and IP address directly on the device

Introduction

Machines are frequently commissioned on site or integrated into the existing infrastructure without STEP 7. You can find typical applications in all areas of the series machine building. Alternative means for assigning the IP address are available for this.

Procedure

1. In the network or device view of the STEP 7 Hardware and Network editor, select the PROFINET interface of an IO controller.
2. Navigate in the Inspector window to "Ethernet addresses".
3. Select the "IP address is set directly at the device" option in the "IP protocol" area.
4. In the "PROFINET" area, select the " PROFINET device name is set directly at the device" check box.

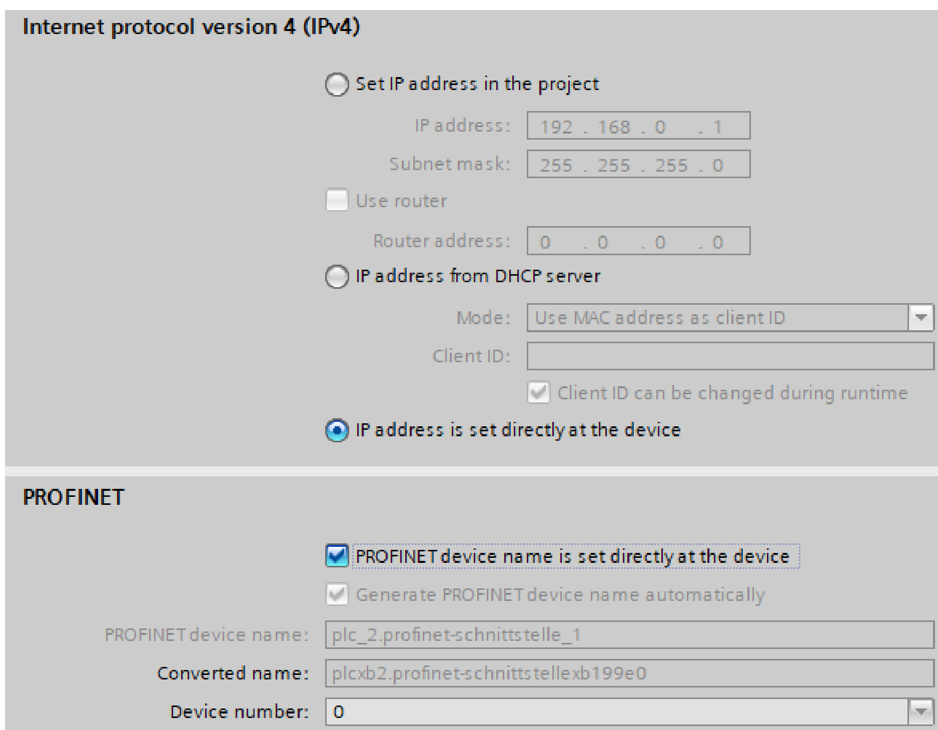


Figure 4-9 Setting the device name and IP address on the device

---

NOTE

Gateway

When you operate a PROFINET device with the option "Allow adaption of the device name/IP address directly on device", you cannot use this PROFINET device as gateway for S7 routing.

Options for assigning IP addresses and device names

Apart from the known address and device name assignment in the "Ethernet addresses" section of the Inspector window, there are other ways in which the IP address and name can be assigned:

- Assigning by means of the user program with the instruction "T_CONFIG"
- Assignment when downloading the configuration to the target system via the "Extended download to device" dialog box.
- Assignment via the Primary Setup Tool (PST)
- Assignment via the PRONETA ("PROFINET network analysis") commissioning and diagnostics tool
- Assignment via the SIMATIC Automation Tool

Additional information

For information on the "T_CONFIG" instruction and on downloading to the target system, refer to the STEP 7 online help.

A free Download (https://support.industry.siemens.com/cs/ww/en/view/109776941) of the Primary Setup Tool (PST) can be found on the Internet. On this Internet page, you will also find a list of devices for which the PST is approved.

4.3 Configuring an IO device through hardware detection

Introduction

As of STEP 7 V15, you have the possibility to detect a real existing IO device and to import it into your project.

You find the IO device in STEP 7 through the "Hardware detection" function. A detected device can be imported into your project. STEP 7 inserts the IO device with all the modules and submodules.

Requirements

- STEP 7 (TIA Portal) as of V15
- It must be possible to technically access the IO device via IP

Procedure

To detect one or more existing IO devices in STEP 7 and add them to the project, follow these steps:

1. In STEP 7, navigate to "Online" > "Hardware detection".
2. Click "PROFINET devices from network...".
   STEP 7 opens the "Hardware detection of PROFINET devices" window.
3. Select the interface of your programming device at "PG/PC interface:".
4. Click "Start search".
   STEP 7 begins with the hardware detection. When the hardware detection is completed, STEP 7 displays the detected IO devices.
5. Select the IO devices that you want to add to the project by clicking the corresponding check box before the IO device.
6. Click "Add devices".
   After a brief moment, a window is opens to report about the success or failure of the hardware detection.

Result of the hardware detection

If the hardware detection is successful, STEP 7 inserts the IO device with all the modules and submodules into the project.

An IO device configured via hardware detection responds as follows:

- Modules configured through the "Hardware detection" are configured as if they have been inserted from the catalog.
- MAC address: STEP 7 imports the MAC address of the detected IO device into the project.
- IP settings:
  – If the detected IO device already has an IP address, STEP 7 imports the IP address into the project.
  – If the detected IO device does not have an IP address, STEP 7 automatically assigns an IP address in the project.

  PROFINET device name:
  – If the detected IO device already has a PROFINET device name, STEP 7 imports the PROFINET device name into the project.
  – If the detected IO device does not have a PROFINET device name, STEP 7 automatically assigns a PROFINET device name in the project.
- IO devices configured through "Hardware detection" have neither an IP subnet nor an IO controller assigned.

4.4 Specifying the router for a PROFINET IO device

Introduction

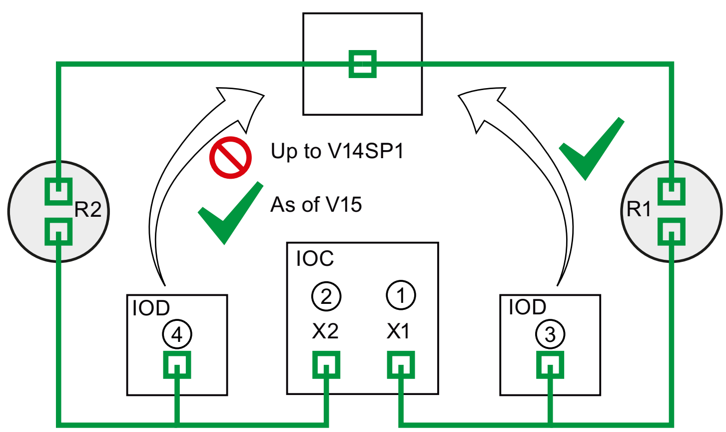
You always require a router (also referred to as a "Standard Gateway") when the PROFINET device has to communicate with a node whose IP addresses lie outside the own IP subnet. If the PROFINET device sends an IP packet to an IP address outside its own IP subnet, the IP packet first goes to the configured router. The router in turn checks the IP address. If this lies outside its own subnet, the router passes the IP packet on to the next router. The IP packet is routed to the next router until it has reached the target address.

Like all S7-1500 CPUs, S7-1500 CPUs with several PROFINET interfaces provide the possibility to configure the IP address of a router. However, there is the restriction that you can only enter the IP address of a router at a PROFINET interface.

You cannot configure an IP address of a router for the other PROFINET interfaces of the CPU. IO devices that are connected to this PROFINET interface adopt this setting. Up to and including STEP 7 V14 SP1, these IO devices did not have any possibility to reach devices in a different IP subnet.

As of STEP 7 V15, you have the possibility to assign the address of a router for an IO device independent of the setting of the IO controller. You can now, for example, set a router address at the IO device in the following cases as well:

- You have not set an IP address of a router for the interface of the associated IO controller.
- You have already set a router address for a different interface in the CPU.



① Router R1 configured at PROFINET X1

② If a router is configured at X1, you cannot configure a router at X2.

③ Because a router is configured at X1, the IO device adopts the IP address of the router R1. The IO device can be reached from a different IP subnet.

④ As of STEP 7 V15, you set the IP address of the router R2 at the IO device irrespective of the setting at the interface X2. The IO device can be reached from a different IP subnet.

Figure 4-10 Specifying the router for an IO device

Further information about the "User router" setting

You have the possibility to configure the use of a router including IP address of the router in the "IP protocol" section of the settings for the PROFINET interface (Ethernet addresses).

Rules

Observe the following rules if you want to configure a router for the PROFINET interface of an IO controller:

- A PROFINET IO device supports exactly one router, irrespective of the number of interfaces.
- You can configure a router for exactly one PROFINET interface. All IO devices that are assigned to the PROFINET interface adopt the configured router from the IO controller.
- You cannot configure a router for the further PROFINET interfaces of the CPU. The further PROFINET interfaces take on the IP address "0.0.0.0" as the router and pass it on to their IO devices.

As of STEP 7 V15, you can configure the use of a router for an IO device. This allows the IO device to communicate with a node outside its own IP subnet, irrespective of the setting of the PROFINET interface of the IO controller.

Configuration example: Configuring a router for an IO device

The following example shows a configuration in which you configure a router at the IO device so that the IO device reaches IP addresses in the higher-level network.

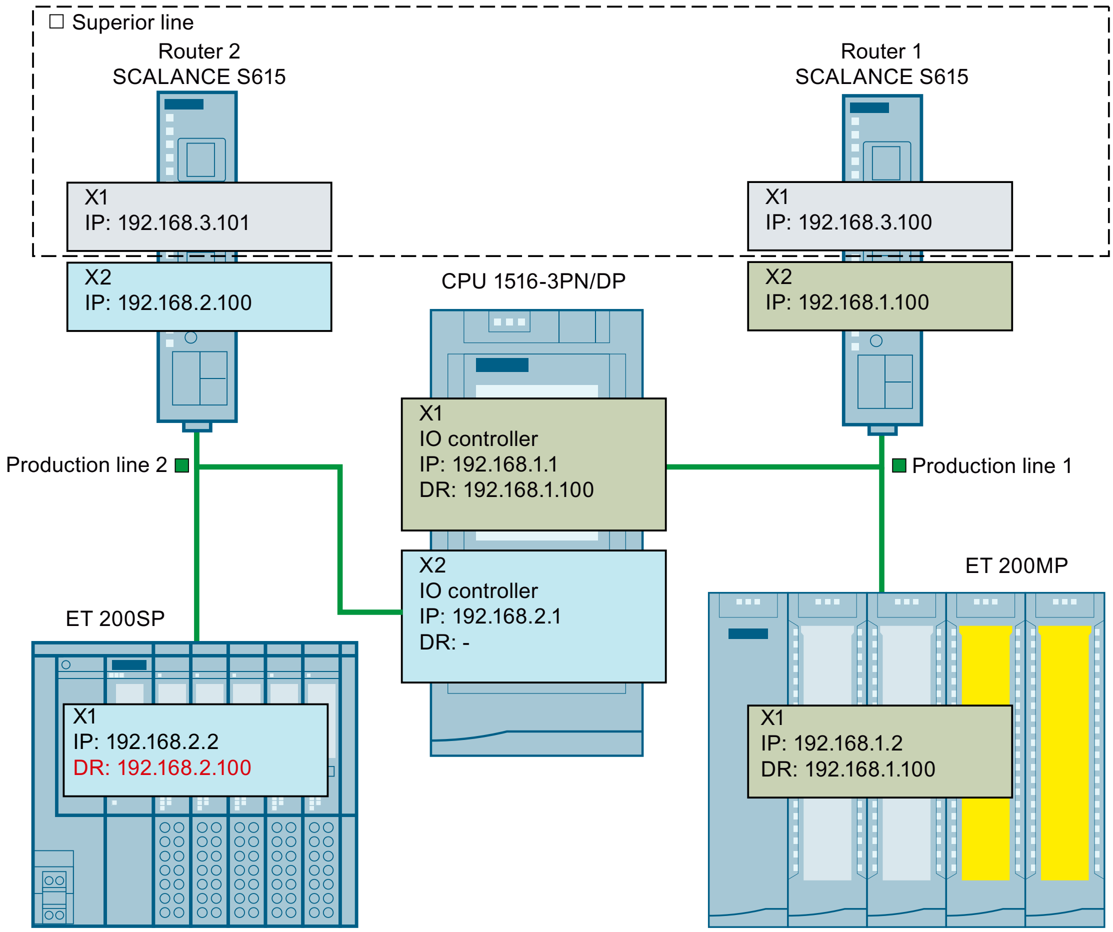


Figure 4-11 Configuration example: Configuring a router for an IO device

You have a CPU 1516-3PN/DP. The two PROFINET interfaces X1 and X2 of the CPU work in the "IO controller" operating mode. The PROFINET interface X1 is connected with the subnet "Production line 1". PROFINET interface X2 is connected with the subnet "Production line 2". The two subnets "Production line 1" and "Production line 2" are each connected via a router with the higher-level network "Superior line".

For PROFINET X1, you configure the router "Router 1" with the IP address 192.168.1.100.

The IO device (ET 200SP) in the "Production line 1" subnet adopts the router from the IO controller.

You cannot configure a router for the PROFINET interface X2 because you have already configured a router for the PROFINET interface X1 of the CPU.

No router is transferred by the PROFINET interface X2 to the IO device in the subnet "Production line 2".

In order for the IO device in the subnet "Production line 2" to reach nodes in the higher-level "Superior line", configure the router "Router 2" with the IP address 192.168.2.100 for the IO device.

Configuring the router for the IO controller

Requirements: You use the "Set IP address in the project" option for the PROFINET interface.

Follow these steps to configure a router for the IO controller in STEP 7:

1. In the network view of STEP 7, select the PROFINET interface of the IO controller.
2. In the Inspector window, navigate to "Properties" > "General" > "Ethernet addresses".
3. Select the "Use router" check box in the "IP protocol" field.
4. Enter the IP address of the router at "Router address".

Configuring a router for an IO device

Requirements:

- STEP 7 as of V15
- CPU 1500 as of firmware version V2.5
- IO device is assigned to the PROFINET interface of an IO controller. The PROFINET interface of the IO controller uses the "Set IP address in the project" option.

Follow these steps to configure a router for the IO device in STEP 7:

1. In the network view of STEP 7, select the PROFINET interface of the IO device.
2. In the Inspector window, navigate to "Properties" > "General" > "Ethernet addresses".
3. Clear the "Synchronize router settings with IO controller" check box.
4. Select the "Use router" check box.
5. Enter the IP address of the router at "Router address".

4.5 Configuring topology

Introduction

If an IO device is assigned to an IO controller, this does not yet specify how the ports are connected to each other.

A port interconnection is not required to use RT, but it provides the following advantages:

- A set topology is assigned with the port interconnection. Based on an online-offline comparison, it is possible to conduct a set-actual comparison with all devices that support this function.
- The "Device replacement without exchangeable medium" function can be used.

A port interconnection is an essential requirement for using IRT.

An overview of various options for setting up a PROFINET network is given below.

Line

All the communication devices are connected in a linear bus topology.

In PROFINET, the linear bus topology is implemented with switches that are already integrated into the PROFINET devices. Therefore, the linear bus topology at PROFINET is merely a special form of tree / star topology.

If a link element (switch, for example) fails, communication across the failed link element is no longer possible. The network is then divided into 2 subsegments.

Linear bus topology structures require the least cabling.

Star

If you connect communication devices to a switch with more than two PROFINET ports, you automatically create a star network topology.

If an individual PROFINET device fails, this does not automatically lead to failure of the entire network, in contrast to other structures. It is only if a switch fails that part of the communication network will fail as well

Tree

If you interconnect several star structures, you obtain a tree network topology.

Ring

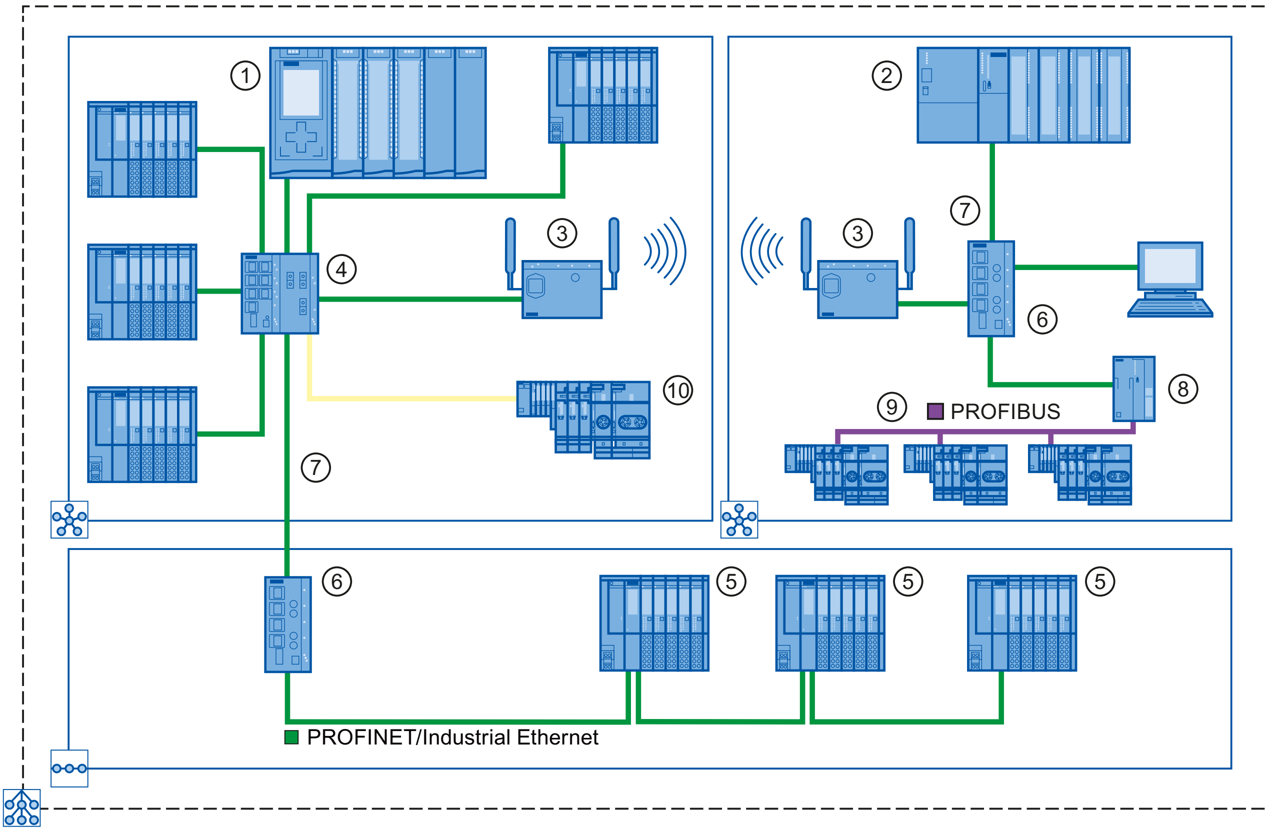
In order to increase the availability of a network, use ring structures. In principle, a linear bus topology is connected to a ring through a so-called redundancy manager.

The task of the redundancy manager is managed by an external switch SCALANCE X, a CPU that supports the Media Redundancy Protocol MRP (e.g., CPU 1516-3 PN/DP) or a CP (e.g., CP 343-1 Lean).

If there is a break in the network, the redundancy manager ensures that the data is redirected over an alternative network connection in the ring.

Example for topology

The following example shows a combination of different topologies.



Number	Meaning
①	S7-1500 as IO controller
②	S7-300 as IO controller
③	Industrial WLAN with SCALANCE W
④	SCALANCE X 307-3 with seven electrical and three optical ports
⑤	ET 200SP with integrated 2-port switch
⑥	SCALANCE X 204 with four electrical ports
⑦	PROFINET/Industrial Ethernet
⑧	IE/PB-Link PN IO
⑨	PROFIBUS DP
⑩	ET 200S with two optical ports
	Star topology
	Linear bus topology
	The combination of topology forms results in a tree topology.

Figure 4-12 Combined topology

Additional information

Observe the PROFINET Installation Guideline (http://www.profibus.com/nc/download/installation-guide/downloads/profinet-installation-guide/display/) of the PROFIBUS User Organization when planning your PROFINET topology.
For more detailed information, see the SIMATIC NET Twisted Pair and Fiber Optic Networks (http://support.automation.siemens.com/WW/view/en/8763736) manual.
You can find basic information in the Communication with SIMATIC (http://support.automation.siemens.com/WW/view/en/1254686) manual.

4.5.1 Topology view in STEP 7

Introduction

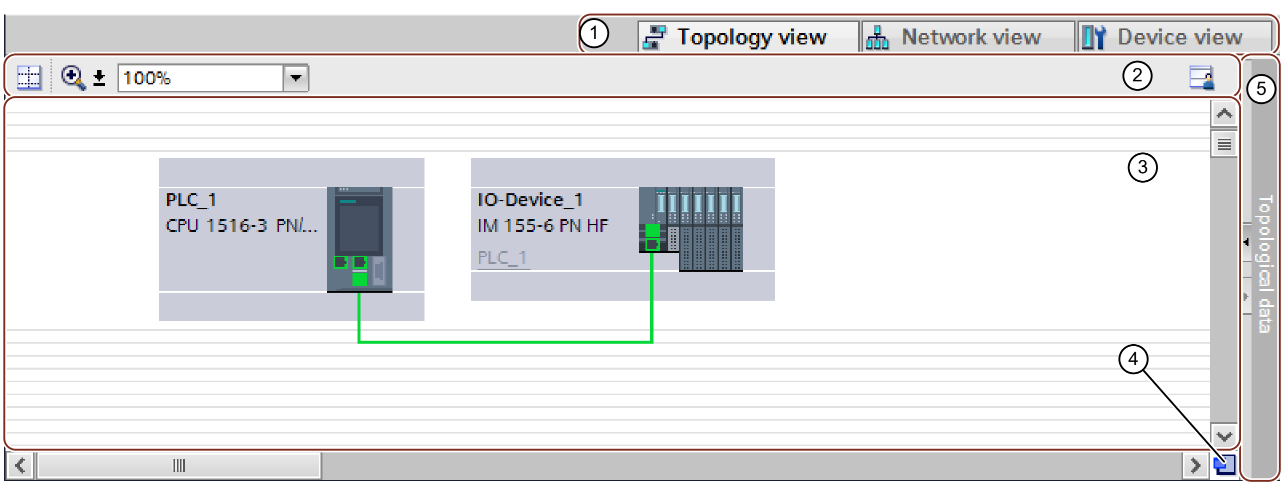
The topology view is one of three working areas of the hardware and network editor. You undertake the following tasks here:

- Displaying the Ethernet topology
- Configuring the Ethernet topology
- Identify and minimize differences between the set and actual topology (online)

The topology view in STEP 7 consists of a graphic area and a table area.

Graphic area

The graphic area of the topology view displays PROFINET devices with their appropriate ports and port connections. Here you can add more PROFINET devices.
The following figure shows the graphic area of the topology view.



① Selector: Device view/Network view/Topology view
② Toolbar
③ Graphic area of the topology view
④ Overview navigation
⑤ Selector for the table area of the topology view

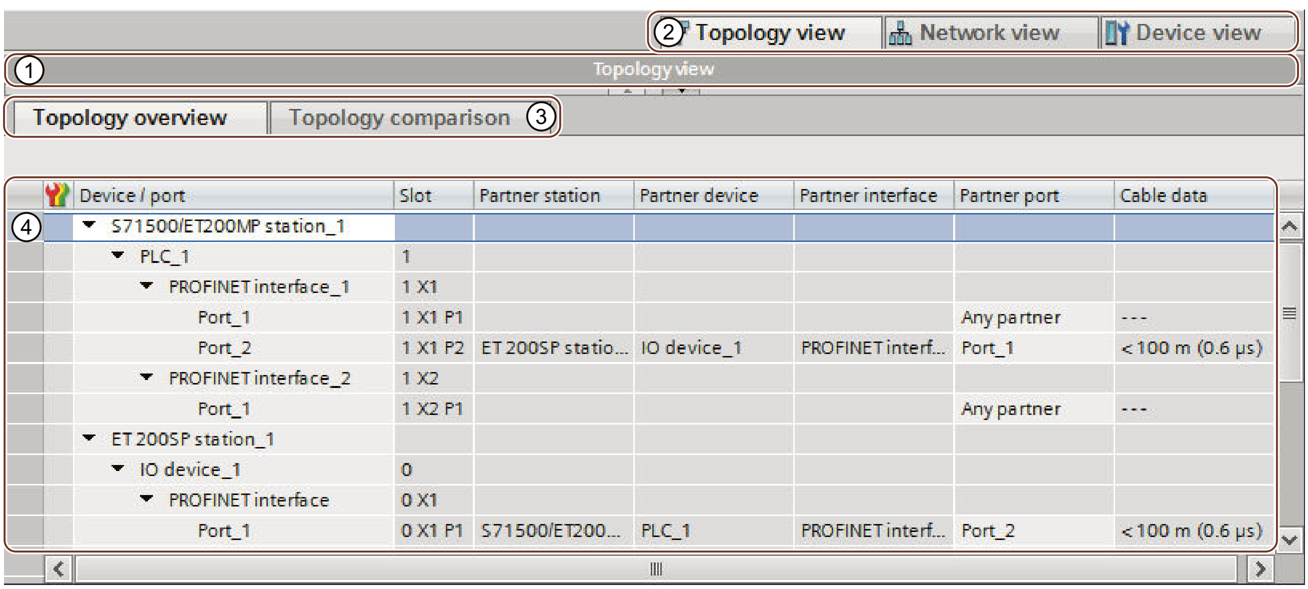
Figure 4-13 Graphic area of the topology view

Overview navigation

Click in the overview navigation to obtain an overview of the created objects in the graphic area. By holding down the mouse button, you can quickly navigate to the desired objects and display them in the graphic area.

Table area

- Topology overview: This displays the Ethernet or PROFINET devices with their appropriate ports and port connections in a table. This table corresponds to the network overview table in the network view.
- Topology comparison: Here you can import devices and port interconnections automatically through offline/online comparison or extended Offline/Online comparison into STEP 7.



Device / port	Slot	Partner station	Partner device	Partner interface	Partner port	Cable data
S71500/ET200MP station_1						
PLC_1	1					
PROFINET interface_1	1 X1					
Port_1	1 X1 P1				Any partner	---
Port_2	1 X1 P2	ET 200SP statio...	IO device_1	PROFINET interf...	Port_1	< 100 m (0.6 µs)
PROFINET interface_2	1 X2					
Port_1	1 X2 P1				Any partner	---
ET 200SP station_1						
IO device_1	0					
PROFINET interface	0 X1					
Port_1	0 X1 P1	S71500/ET200...	PLC_1	PROFINET interf...	Port_2	< 100 m (0.6 µs)

① Selector for the graphic area of the topology view
② Selector: Device view/Network view/Topology view
③ Selector: Topology overview/Topology comparison
④ Table area of the topology view

Figure 4-14 Table area of the topology view

4.5.2 Interconnecting ports in the topology view

Requirement

You are in the graphic view of the topology view.

Procedure

To interconnect ports in the topology view, follow these steps:

1. Place the pointer of the mouse on the port you want to interconnect.
2. Press and hold down the left mouse button.
3. Move the pointer.
   The pointer now uses the networking symbol to indicate "Interconnecting" mode. At the same time, you see the lock symbol on the pointer. The lock symbol disappears only when the pointer is over a valid target position.
4. Now drag the pointer to the target port. You can keep the left mouse button pressed or release it when performing this action.
5. Now release the left mouse button or press it again (depending on your previous action).

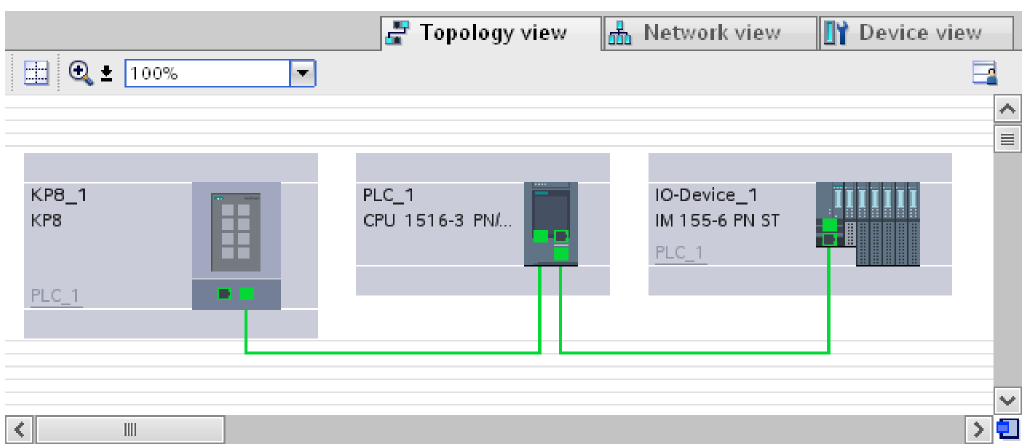


Figure 4-15 Interconnecting ports in the topology view

Result

You have created a port interconnection.

4.5.3 Interconnecting ports - Inspector window

Interconnecting ports in the Inspector window

To interconnect ports, follow these steps:

1. In the device or network view, select the PROFINET device or PROFINET interface.
2. In the Inspector window, navigate to the port property "Port interconnection".
   When the PROFINET interface is selected, you can find this setting in the Inspector window as follows: "Properties > General > Advanced Options > Port [...] > Port Interconnection."
3. In the "Local port" section, you can find the settings at the local port. In the case of fiber-optic cable you can, for example, set the cable names here.
   In the "Partner port" area, select the drop-down list for "Partner port" in order to display the available partner ports and make a selection.

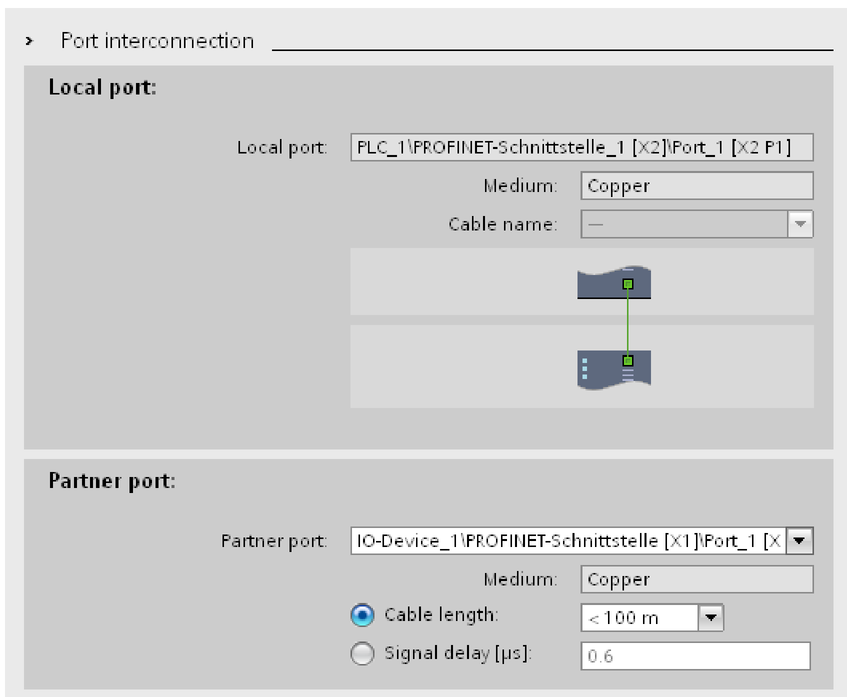


Figure 4-16 Interconnecting ports in the Inspector window in STEP 7

If the PROFINET interface was disconnected, it is automatically connected by this action. In the properties of the subnet you can set whether this subnet should or should not be used for the networking.

4.5.4 Automatic assignment of devices by offline/online comparison

Overview

During the offline/online comparison, the configured topology is compared with the actual existing topology. Devices identified online are automatically assigned to configured devices as far as this is possible.

Start of availability detection

You start the availability detection the first time by clicking the "Compare offline/online" button in the toolbar of the "Topology comparison" tab.
You restart availability detection by clicking the "Update" button.

NOTE

The availability detection can take several seconds. During this time, no user input is possible.

Automatic assignment of a PNIO device

A PNIO device identified online is automatically assigned to a configured device if the following properties of the two devices match up:

- Article no.
- Type
- PROFINET device name

No automatic assignment

In the following situations, no automatic assignment is possible:

- No device can be identified online to match a configured device. (This means that the corresponding columns in the "Online topology" area of the topology comparison table are empty.)
  In this case, you should add the already configured device to your system or delete the configured device from the configuration.
- A device identified online can be assigned to a configured device, but there are differences in the port interconnections.
  In this case, you can Apply the port interconnections identified online manually to the project (Page 80).
- A device identified online cannot be assigned to a configured device. (In this case, the corresponding columns in the "Offline topology" area of the topology comparison table are empty.)
  In this case, you can Include the devices identified online manually in the project (Page 81).

4.5.5 Apply the port interconnections identified online manually to the project

Requirements

You have run an offline/online comparison in the topology view. The result of this is that at least one device identified online was automatically assigned to a configured device, but that there are differences relating to the interconnection.

Procedure

To adopt one more port interconnections identified online in the project manually, follow these steps:

1. Select the row belonging to the port interconnection.
2. If applicable, select further roles using multi-selection.
3. Select "Apply" > "Use selected" in the shortcut menu.
   The content of the corresponding table cells in the "Action" column changes to "Apply".
4. If you have mistakenly prepared too many port interconnections to be included in the project:
   Select the rows belonging to the port interconnections you have mistakenly prepared for inclusion in the project using multi-selection.
   Select "Reset" > "Reset selected" in the shortcut menu.
   The content of the corresponding table cells in the "Action" column change to "No action".
5. Click the "Synchronize" button.

Result

The port interconnections identified online for the corresponding devices are included in the project. Successful adoption is indicated by the diagnostics icon "Identical topology information" for each port.

NOTE

If other port interconnections are recognized for a device identified online and these differ from those that exist in the project, adopting these in the project means that the port interconnections that were previously in the project are replaced by those identified online. If no port interconnections are detected for a device identified online, adopting in the project means that all the port interconnections of this device are deleted in the project.

4.5.6 Include the devices identified online manually in the project

Requirements

You have run an offline/online comparison in the topology view. The result of this is that at least one device identified online could not be assigned to any configured device.

Procedure

To adopt one more devices identified online in the project manually, follow these steps:

1. For a configured device without an online partner, move the mouse pointer to the "Device/port" column of the online topology.
2. Select the device you want to assign to the configured device from the drop-down list of this box.
3. Repeat the previous steps if necessary for other configured devices without an online partner.

Result

The selected device that was identified online is moved up from the end of the table. Following this, it is in the row of the configured device to which you have just assigned it.

4.5.7 Automatic assignment of devices by advanced offline/online comparison

Overview

With the advanced offline/online comparison, ICMP is also used alongside DCP to be able to detect devices that do not support DCP.

Automatic assignment of devices detected by ICMP

With devices detected by ICMP, no type is available.

With passive devices, no article number is available. For this reason, passive devices can only be assigned automatically if you have not assigned an article number in the configured data and the offline and online IP addresses match.

With switches, automatic assignment is possible if the offline and online article number, IP address and PROFINET device name match.

Diagnostics and maintenance

5

5.1 Diagnostics mechanisms of PROFINET IO

Totally Integrated Diagnostics Concept

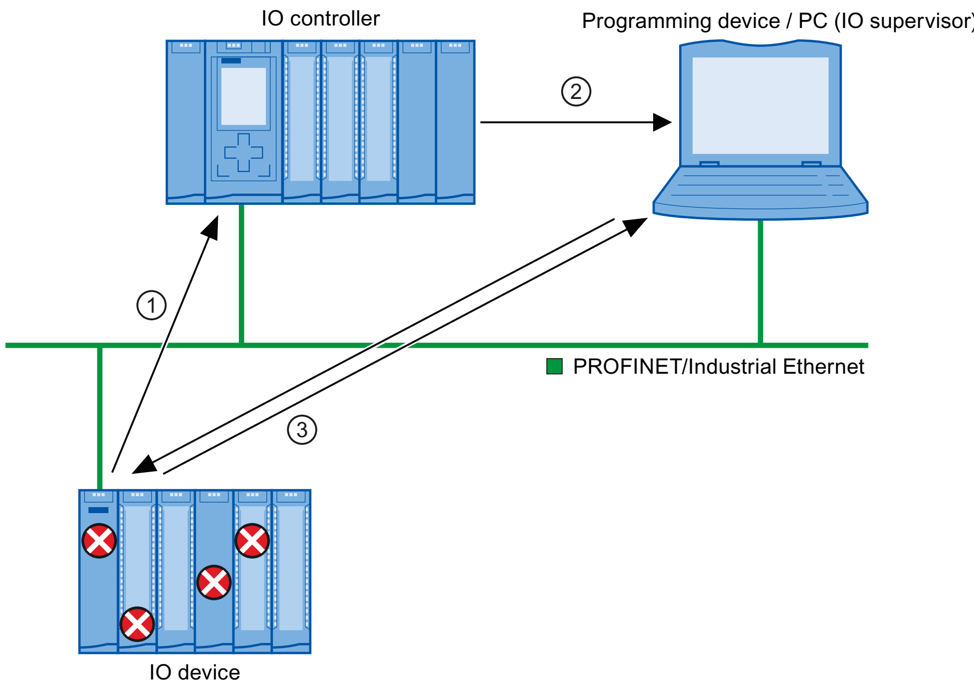
All SIMATIC products have integrated diagnostics functions that they can use to detect and remedy faults. The components automatically flag a potential fault in the operation and provide detailed information.

Each individual error or several errors occurring concurrently are transferred from the IO device to the IO controller. If you require the full status of the IO device including any pending errors, you can also read the status directly from the IO device.

The following sections provide basic information on using diagnostics via PROFINET IO. You can find a detailed description of the system diagnostics for S7-1500, ET 200MP, ET 200SP and ET 200AL in the Diagnostics (http://support.automation.siemens.com/WW/view/en/59192926) function manual.

Accessing the status of an IO device with a PG/PC or an HMI device

If you are connected to the Industrial Ethernet via a PG/PC with STEP 7 or an HMI device, you can also call up diagnostics information online. This is illustrated by the following graphic.



Number	Description
①	The IO device detects an error and sends diagnostics data to the IO controller.
②	The IO controller notifies the programming/HMI device. The display of the system diagnostics is updated.
③	In STEP 7, you can read the station status for "Accessible devices" directly from the IO device regardless of the IO controller. This is only possible if the programming device is connected to Industrial Ethernet. This means that you can access diagnostics information during the commissioning phase or for servicing even if the IO controller is not operational.

Figure 5-1 PROFINET IO diagnostics with PG/PC or HMI device

5.1.1 Diagnostics levels in PROFINET IO

Concept

The IO device sends all error messages that occur to the IO controller. The scope and volume of diagnostics information varies according to the level of diagnostics data evaluation and the PROFINET devices you are using.

Diagnostics levels

You can evaluate diagnostics data at different levels. The number and type of channels is selected, for example, at the diagnostics level 4.
The following figure shows the diagnostics levels with PROFINET IO.

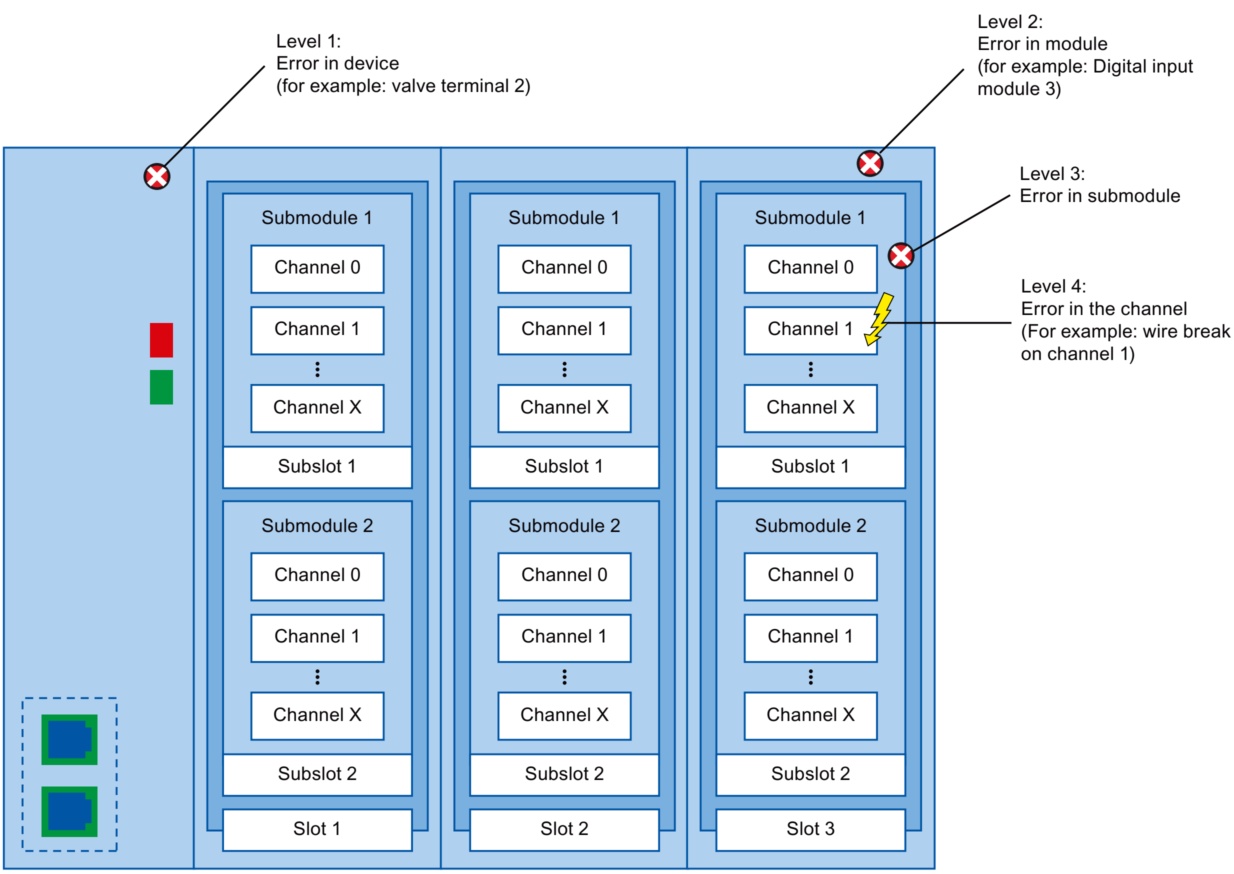
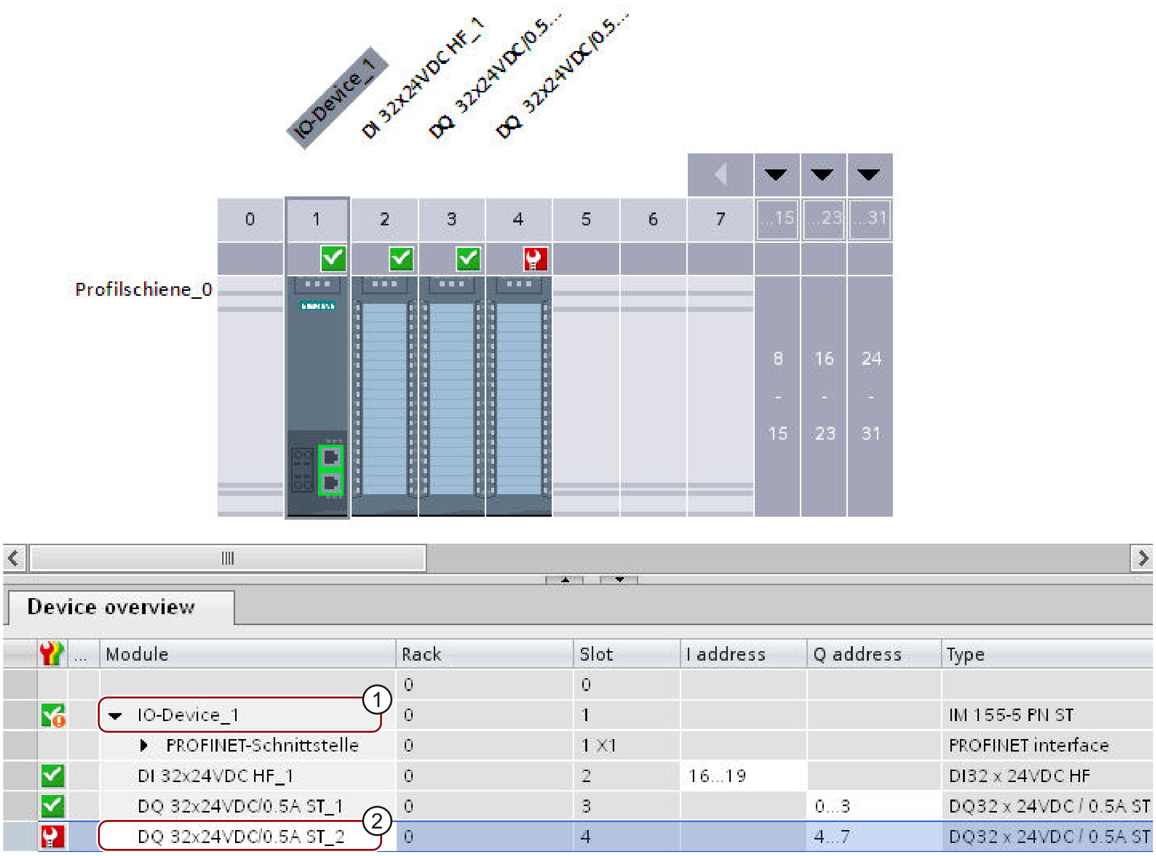


Figure 5-2 Diagnostics levels with PROFINET IO

Representation of diagnostics levels in the device view in STEP 7

The following figure shows the representation of the PROFINET device model in the device view of STEP 7, based on the example of a distributed I/O system ET 200MP:



Number	Description	
①	Level 1:	Error(s) in the device
②	Level 2:	Error(s) in the module

Figure 5-3 Diagnostics levels in the device view of STEP 7

Which PROFINET nodes support the extended PROFINET diagnostics?

An overview of the PROFINET nodes that support extended PROFINET diagnostics and of what you have to configure is provided in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/23678970).

LEDs for diagnostics on PROFINET

Each port of a PROFINET interface of a SIMATIC device has one LED.
The following table shows a summary of the meaning of these LEDs in the S7-1500, ET 200MP, ET 200SP and ET 200AL systems.

Table 5-1 S7-1500, ET 200MP, ET 200SP, ET 200AL: LEDs for diagnostics on PROFINET

LED image	Meaning	S7-1500	ET 200MP	ET 200SP	ET 200AL
		LINK/TX/RX LED		LK LED	P1 Link LED P2 Link LED
LED off	There is no Ethernet connection between the PROFINET interface of the PROFINET device and the communication partner. No data is currently being sent/received via the PROFINET interface.	X	X	X	X
LED green	There is an Ethernet connection between the PROFINET interface of your PROFINET device and a communication partner.	X	X	X	X
LED flashes green	The "LED flashing test" is being performed.	X	X	X	X
LED flickers yellow	Data is currently being received from or sent to a communications partner on Ethernet via the PROFINET interface of the PROFINET device.	X	X	-	-

Additional information

You can find a detailed description of all LEDs of the module with cause of the error and remedies in the relevant documentation for the module.

5.3 Diagnostics via the display of the S7-1500 CPUs

Display

The S7-1500 CPU has a display and operating keys. The display of the CPU shows you the control and status information in different menus. You use operating keys to navigate through the menus and make a variety of settings in the process.

Diagnostics via the display

The following indicators can be evaluated for diagnostics purposes on the display:

- Error and alarm texts (system diagnostics, alarm messages)
- Module status for central and distributed modules

In the following example of a display of the CPU 1516-3 PN/DP, you can see a warning on the diagnostics icon and an exclamation mark on the icon for module.

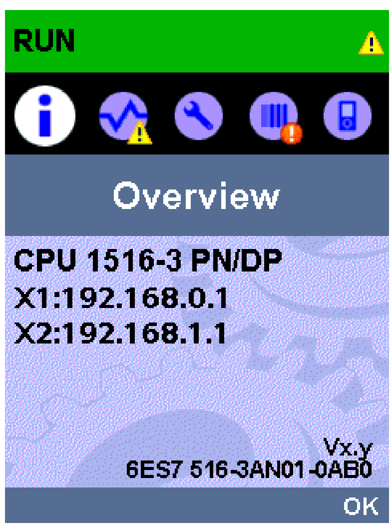


Figure 5-4 Display of overview

Module status

To show the module status, navigate on the display through the menu items "Module" > "PROFINET I/O (X1)" > "Station" > "Slot" > "Status" > "Module status".

The module status indicates that a fault has occurred in the module. The "lower-level status" is the status of the module in the diagnostics level below this. In the example, the status is "good", i.e., the fault is not in the lower diagnostics level submodule or channel, but instead in the module.

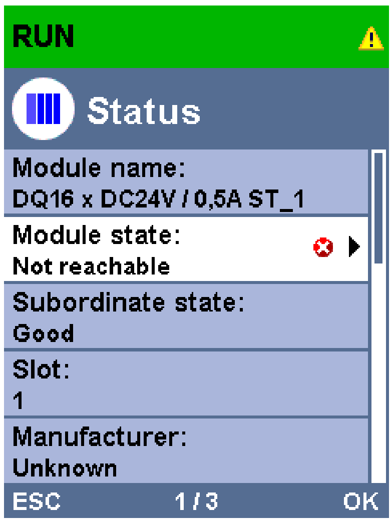


Figure 5-5 Display of module status

Error and alarm texts

You can show diagnostics buffer entries and alarm messages for the relevant automation system on the display.

To show the diagnostics buffer entries of the CPU, navigate on the display via the menu items "Diagnostics" > "Diagnostics buffer".

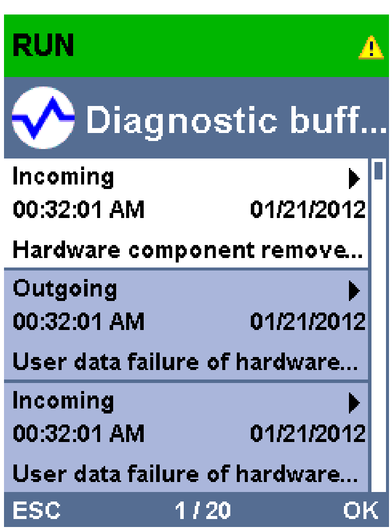


Figure 5-6 Display of diagnostics buffer

To show the alarm messages of the automation system, navigate through the menu items "Diagnostics" > "Alarms" > "Alarm text" on the display.

NOTE

Updating the alarm display

The display shows the currently read status of the CPU in static form, the display is not automatically updated. The alarm display is updated after it has been exited and opened again.

You set the automatic updating of the diagnostics information under: "Display" > "DiagnosticRefresh".

Figure 5-7 Display of alarms

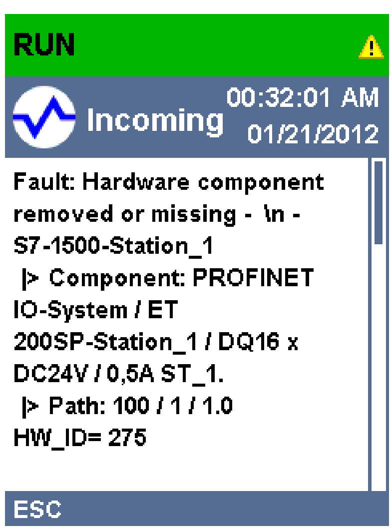


Figure 5-8 Display of alarm message

Additional information

You can find the description of the operation and functions of the display in the SIMATIC S7-1500 Display Simulator (http://www.automation.siemens.com/salesmaterial-as/interactive-manuals/getting-started_simatic-s7-1500/disp_tool/start_de.html).

5.4 Diagnostics via Web server

The CPUs belonging to the S7 family have their own integrated Web server with a wide range of functions.

The following diagnostics options are available to you:

- Start page with general CPU information
- Information on diagnostics
- Contents of the diagnostics buffer
- Module information
- Alarms
- Information about communication
- PROFINET topology
- Motion Control diagnostics
- Trace
- Contents of the diagnostic buffer
- Module status
- Actual topology of the PROFINET system
- Set topology of the PROFINET system (from the configuration)

Set topology and actual topology - graphic view

Requirements for displaying the set and actual topology:

- You have configured the PROFINET ports in the topology editor of the hardware and network editor of STEP 7.
- You have loaded the entire project with STEP 7 in the CPU.

The following shows an example of the graphic view.

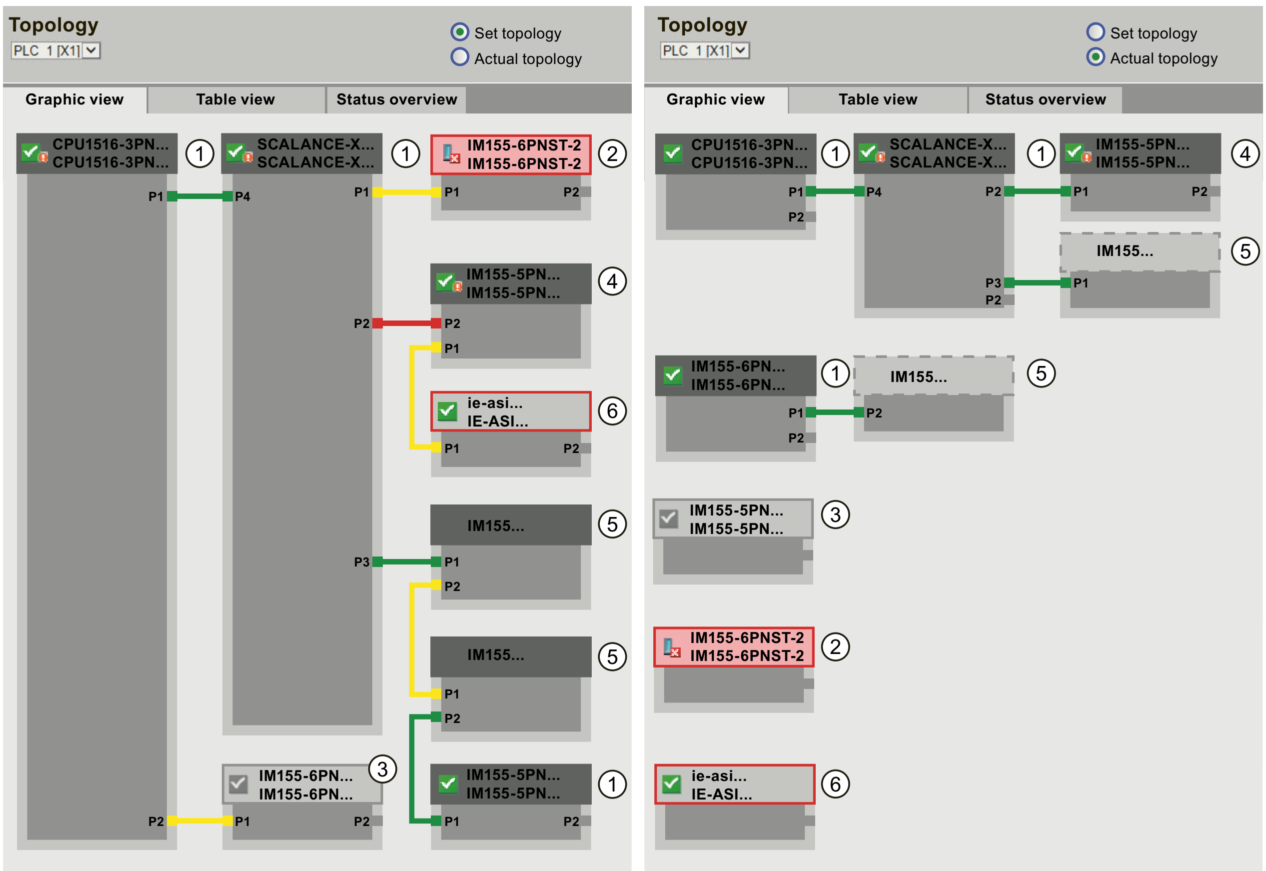


Figure 5-9 Topology - graphic view via the Web server

Meaning of the colored connections in the set/actual topology:

Table 5-2 Meaning of the colored connections in the set/actual topology:

Connection	Meaning	
	Set topology	Actual topology
green	The current actual connection matches the configured set connection.	detected connections
red	The current actual connection does not match the configured set connection (e.g., port interchanged).	-

Connection	Meaning	
	Set topology	Actual topology
yellow	The connection cannot be diagnosed. Causes: • The communication to an IO device has been disrupted (e.g., cable removed) • Connection to a passive component • Connection to PROFINET devices of another IO controller or PROFINET system	-

① Configured and accessible PROFINET devices

Configured and accessible PROFINET devices are displayed dark-gray. Connections show the ports through which the PROFINET devices of a station are connected.

① Configured but not accessible PROFINET devices

The configured but not accessible PROFINET devices are displayed in pink with red border (e.g., device has failed, cable disconnected).

③ Disabled devices

All disabled, configured PROFINET devices are displayed light-gray.

④ Interchanged ports

Interchanged ports are marked red in the set topology view. The currently connected ports are displayed in the actual topology, the configured set connection in the set topology.

⑤ PROFINET devices of another PROFINET IO system

- In the set topology:
  A PROFINET device of another PROFINET IO system is displayed with a green connection (or red connected if the ports have been interchanged), when it is directly adjoining a configured and accessible PROFINET device ① and it is also accessible. When the PROFINET device of another PROFINET IO system is not accessible, a yellow connection line is displayed.
  The connection between two PROFINET devices that both belong to a different PROFINET IO system, cannot be determined and is always displayed in yellow.
- In the actual topology:
  A PROFINET device of another PROFINET IO system is only displayed if the PROFINET device is in direct proximity to a configured PROFINET device. The PROFINET device is displayed light-gray and with dashed line.

For PROFINET devices of a different PROFINET IO system, **no** status display is shown on the device head.

⑥ Representation of faulty neighbor relations

The devices whose neighbor relations cannot be read out completely or correctly are displayed light-gray with red border.

Additional information

The tabular view of the actual topology and the status overview of the PROFINET devices in the project are possible.
You can find these views, additional topology examples, and detailed information on the operation and the functions of the Web server in the Web server (http://support.automation.siemens.com/WW/view/en/59193560) manual.

5.5 Online diagnostics with STEP 7

For PROFINET, you have the following options to evaluate diagnostics in STEP 7:

- Online & diagnostics - Devices & networks
- Online & diagnostics - diagnostics of PROFINET ports

Online & diagnostics network view

In the hardware and network editor (launched from the "Project tree" by double-clicking "Devices & networks"), you can get an overview of the current state of your system by clicking on the "Go online" button. It also shows the configuration information (for example, non-configured modules). This option is also available in the topology view in similar form.
Schematic drawing of the network view (online):

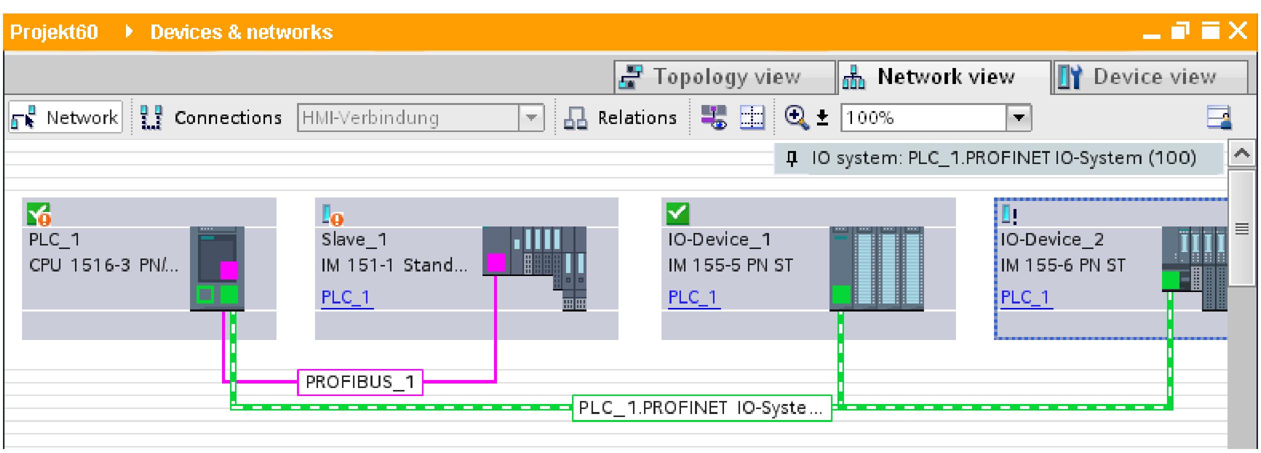


Figure 5-10 Online & diagnostics network view

Online & diagnostics device view

In STEP 7, you can display an overview of the modules in which faults have occurred. To do this, select the menu command "Online > Online & diagnostics". Once you are connected, you can see the status of the accessible devices in the project tree.

Double-click the device which displays an alarm message to access the faulty module directly. The device view is opened in the work area. In the device view of the device that reports the fault you can see directly in which module the fault occurs.

Open the "Diagnostics" tab and the subordinate "Device information" tab in the Inspector window for a more detailed error description.

Schematic drawing of the device view (online):

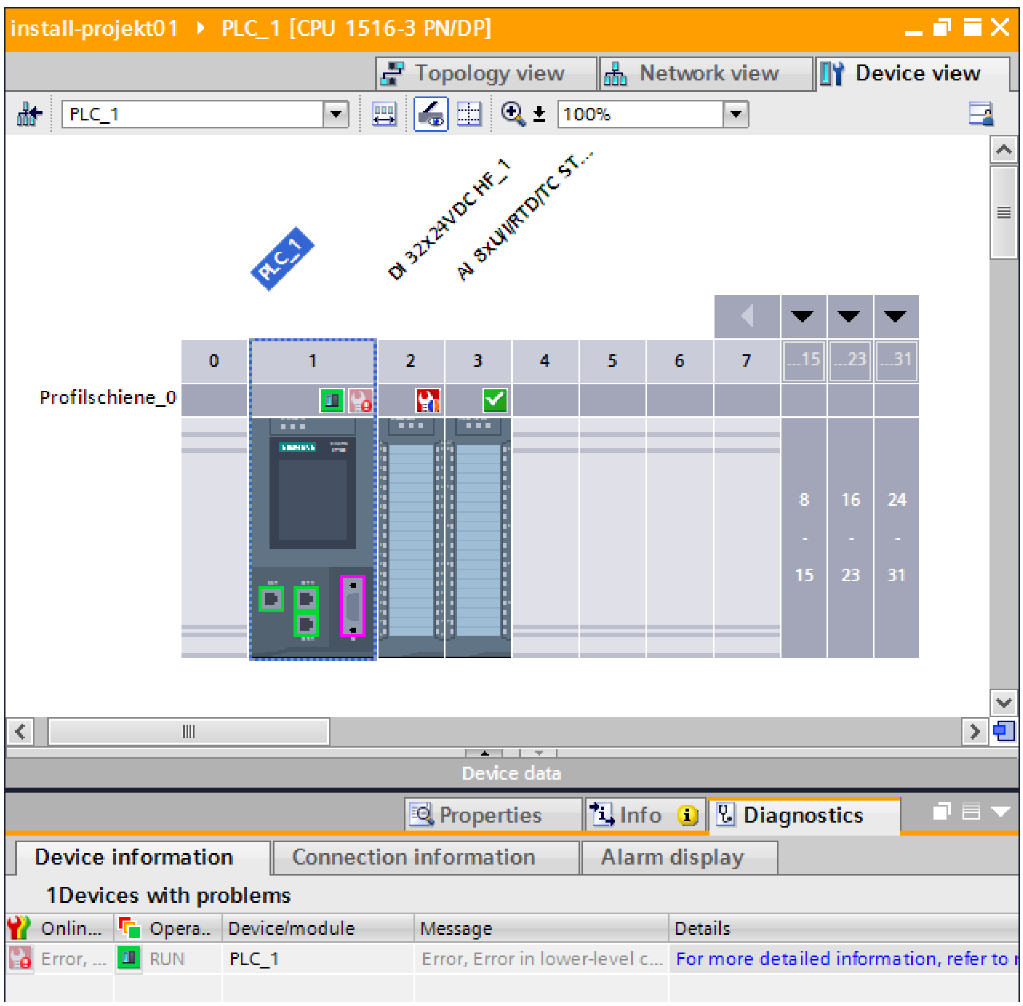


Figure 5-11 Online & diagnostics device view

Diagnostics of PROFINET ports

If you select "PROFINET interface > Ports" in the Diagnostics area in the online & diagnostics device view of a PROFINET device, the ports of the PROFINET interface are listed in a table.

The table provides you with the following information about the ports of the PROFINET interface.

- Name
- Status
- Settings
- Operating mode

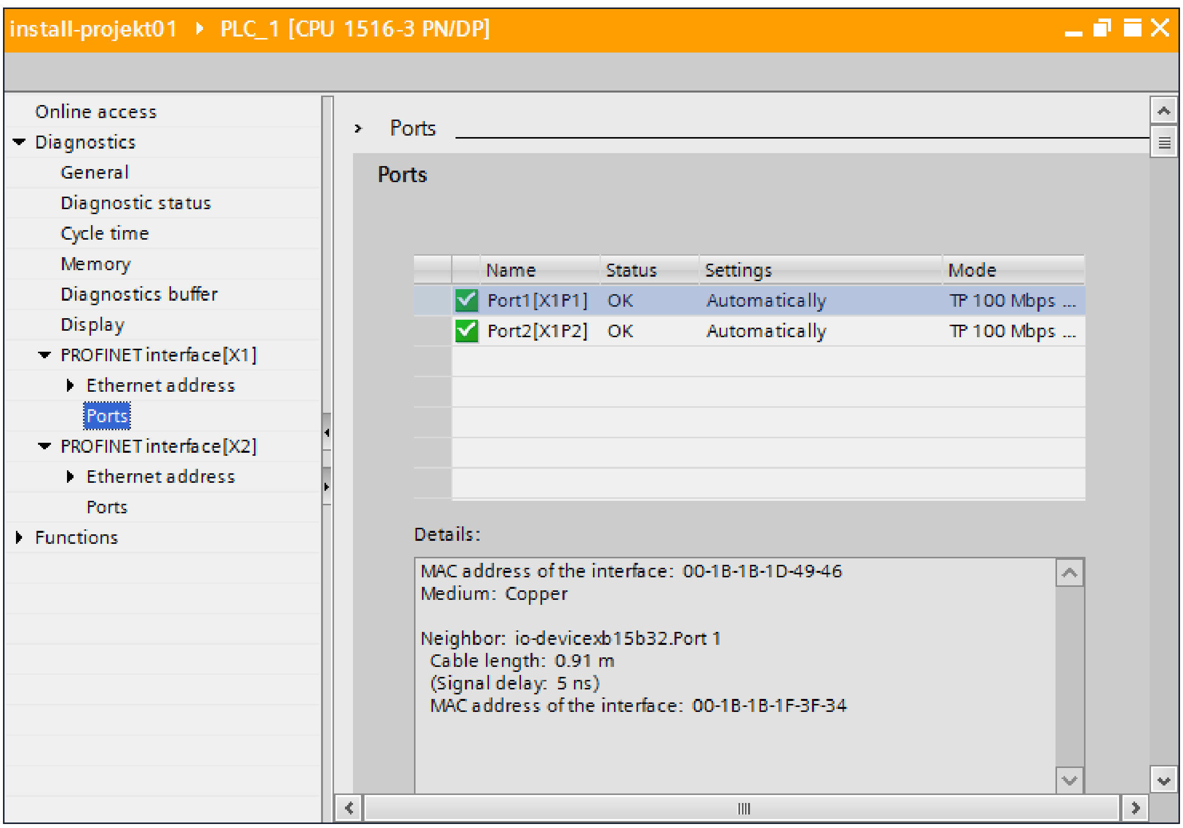


Figure 5-12 Diagnostics of PROFINET ports in STEP 7

Additional information

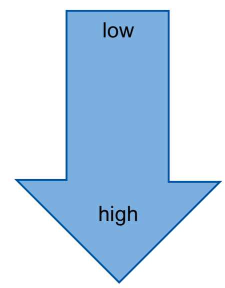
You can find information on the system diagnostics for S7-1500, ET 200MP, ET 200SP and ET 200AL in the Diagnostics (http://support.automation.siemens.com/WW/view/en/59192926) function manual and online help for STEP 7.

5.6 Extended maintenance concept

Extended maintenance concept

The PROFINET interfaces with integrated switch of the SIMATIC devices support the four-level diagnostics concept in accordance with PROFINET specification Version V2.3 or higher with the following status:

Table 5-3 Classification of the diagnostic status

Diagnostic status	Symbol		Severity of the error
Good	Green checkmark		
Maintenance required	Green wrench		
Maintenance demanded	Yellow wrench		
Bad	Red wrench		

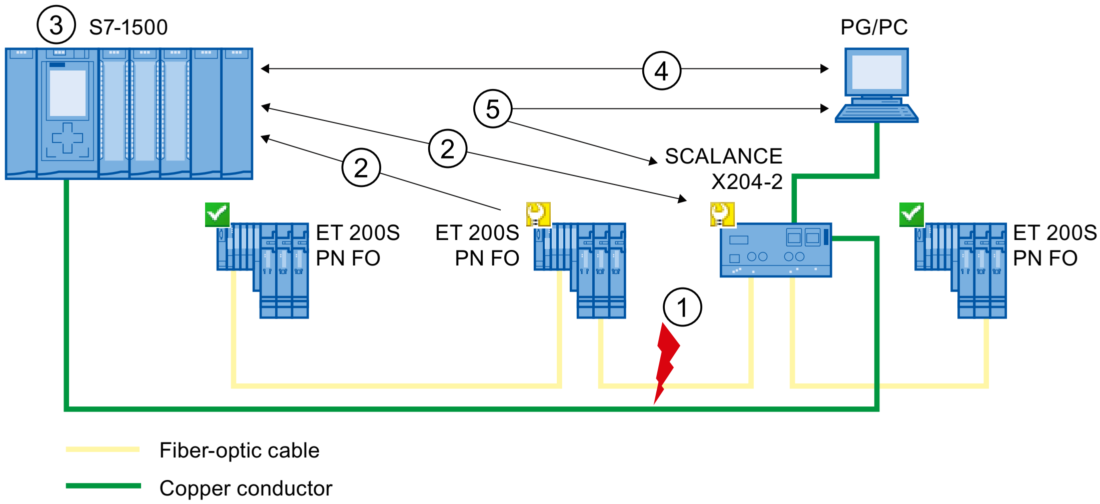
The aim of the diagnostics concept is the early detection and elimination of potential faults - before they cause a production outage.
Other status information is defined in addition to the Good (no fault) and Bad (fault) status information for a PROFINET device.
The maintenance information is generated with the following system alarms:

- Maintenance required (symbolized by a green wrench) and
- Maintenance demanded (symbolized by a yellow wrench)

The times at which the two system alarms are generated can be customized for most wear parameters.

Example: Maintenance demanded for a PROFINET cable

The following graphic illustrates how diagnostics information is exchanged when the transmission quality on the optical cable decreases due to ageing, for example. In this example, the scenario is considered after a maintenance required has already been diagnosed.



Number	Description
①	The system reserve of the fiber-optic cable drops below 0 dB.
②	Both the ET 200S PN FO and the switch send the maintenance demanded alarm to the IO controller.
③	Based on the interrupts, the IO controller detects the maintenance demanded from the switch and from the IO device. The module information data is updated in the IO controller and the corresponding error OBs are called. Note: To be able to start the error OBs in the IO controller, the "Call the user program if communication errors occur" property must be selected in STEP 7 for the relevant IO controller.
④	In STEP 7 (on the programming device/PC), the maintenance demanded message is indicated on the IO device and at the switch by a yellow wrench symbol.
⑤	STEP 7 can also read out detailed information directly from the switch.

Figure 5-13 Maintenance demanded for a PROFINET cable

5.7 Diagnostics of the network topology

Availability

As an open standard, you can use any SNMP based systems or software solutions for diagnostics in PROFINET.

Network diagnostics

The network management protocol SNMP (Simple Network Management Protocol) uses the wireless UDP transport protocol. It consists of two network components, similar to the client/server model. The SNMP manager monitors the network nodes and the SNMP clients collect the various network-specific information in the individual network nodes and store it in a structured form in the **MIB** (Management Information Base). This information allows a network management system to run detailed network diagnostics.

MIB

The MIB (Management Information Base) is the database of a device. SNMP clients access this database in the device. The S7 device family supports the following standard MIBs:

- MIB II, standardized in the RFC 1213
- LLDP-MIB, standardized in the international standard IEC 802.1AB
- LLDP-PNIO-MIB, standardized in the international standard IEC 61158-6-10

You will find the MIBs for ET 200 interface modules and couplers with PROFINET interface in this product note (https://support.industry.siemens.com/cs/ww/en/view/109770525).

Detecting the network topology

LLDP (Link Layer Discovery Protocol) is a protocol that is used to detect the closest neighbor. LLDP enables a device to send information about itself and to receive information from its neighbor devices. This information is then saved in the LLDP MIB, for example, and can be queried using SNMP. This information allows a network management system to determine the network topology.

Use of SNMP (Simple Network Management Protocol)

SNMP can be used as follows:

- By users to integrate network diagnostics into a central HMI/SCADA system using the SIMATIC NET OPC server
- By the IT administration of machine and plant operators to monitor their Industrial Ethernet network using standard network management systems.
- By the IT administration, to monitor the automation network, alongside the office network, using standard network management systems.

Use of SNMP in the SIMATIC NET environment

SNMP-compliant devices from the SIMATIC NET family can be monitored and operated via a conventional standard Internet browser. The management system known as web-based management offers a wide range of device-specific information (network statistics, status of redundant supply, for example).

5.8 Diagnostics in the user program

5.8.1 Diagnostics and configuration data records

Diagnostics mechanism

The IO device outputs a diagnostics interrupt to the IO controller when it detects faults such as wire break on an IO module. This interrupt calls a corresponding organization block in the user program (diagnostics interrupt OB82), in order to generate a defined (programmed) response to the fault and passes a diagnostics data record.

Diagnostics data records in PROFINET IO

There are two different types of diagnostics data record:

1. Channel diagnostics data records
   Channel diagnostics data records are generated if a channel is in an error state and / or has triggered an interrupt.
   A diagnostics data record of length 0 is returned if there is no fault.
2. Vendor-specific diagnostics data records
   The structure and size of vendor-specific diagnostics data records depend on the vendor's settings.
   For information about vendor-specific diagnostics data records, refer to the appropriate device manual.

Addressing levels of diagnostics data records

Diagnostics and configuration data is evaluated at the following addressing levels:

- Device level
- AR (Application Relation)
- API (Application Process Identifier)
- Slot
- Subslot

A group of diagnostics and configuration data records are available for each address level (exception: device level always 0xF80c). In HEX representation, the individual groups of data records are distinguished by the first letter of the data record number.

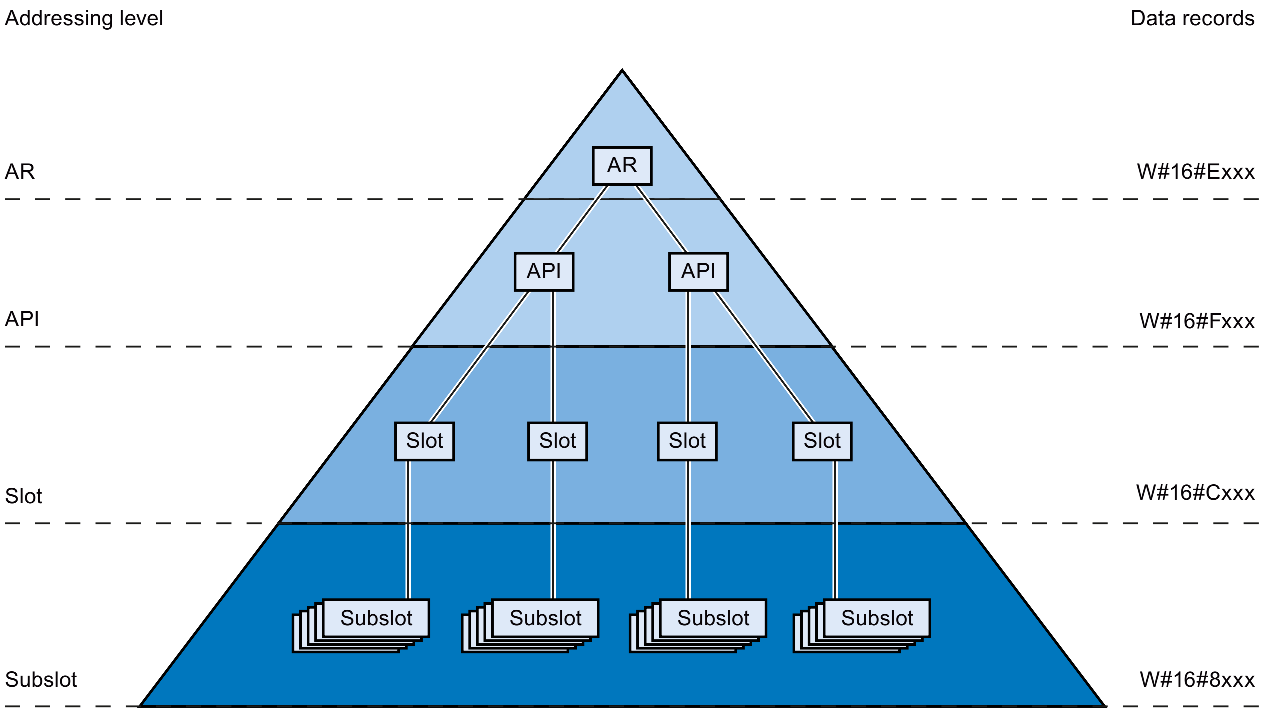


Figure 5-14 Addressing levels of diagnostics data records

The information for each IO device (addressing level AR), module (addressing level slot) or submodule (addressing level subslot) is always transferred in separate diagnostics or configuration data records. The data record returns diagnostics data or configuration data for one or more subslots, slots and APIs, depending on the addressing level.

NOTE

The diagnostics information is only generated for configured modules / submodules / channels.

Library for PROFINET data records

The "LPNDR" block library contains various function blocks. You can use these blocks to read out various device and status information from a PROFINET device and write parameters.

You can execute the following functions using the library blocks:

- Read device information:
  - Information about the interface, e.g. IP and MAC address
  - Information about the device interface, e.g. status, medium, name
  - Information about the link status of the interfaces, e.g. link down, link up
  - Information about the role of the device for MRP, e.g. client, manager
  - Information about the port statistics, e.g. number of received bytes
- Read the MRP status
- Modification of the parameters of analog input module of the ET 200SP in runtime, e.g. in order to disable/enable the channel diagnostics.

The finished functions are freely adaptable and can therefore be used universally.

The "LPNDR" block library with an example project can be found on the Internet (https://support.industry.siemens.com/cs/ww/en/view/109753067).

5.8.2 Evaluate diagnostics in the user program

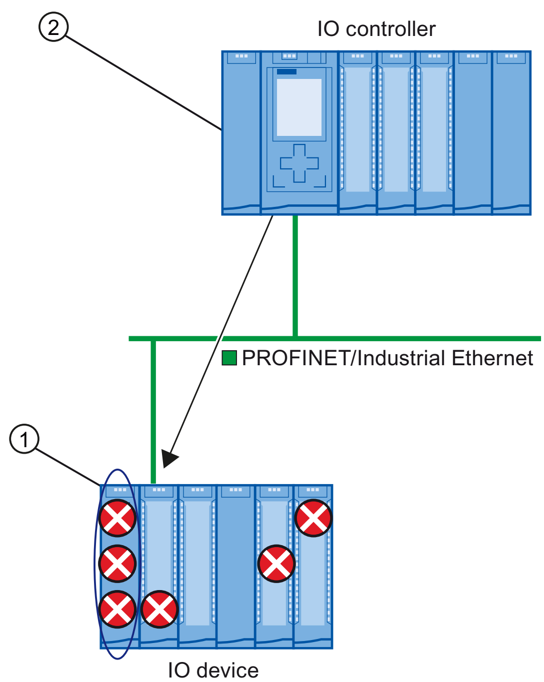
Diagnostics in the user program

For PROFINET IO, a cross-vendor structure of data records with diagnostics information applies. Diagnostics information is created only for channels on which a fault has occurred. With PROFINET, there are two basic ways to obtain diagnostics information.

1. Evaluating the diagnostics status

Read out the diagnostics of your PROFINET IO system using the instructions "DeviceStates" and "ModuleStates" to localize faulty stations / modules or station / modules with maintenance demanded or maintenance required within a PROFINET IO system.

The instruction RDREC (read data record) is then used to read various diagnostics data records directly from the module concerned and thus obtain detailed information about the error.

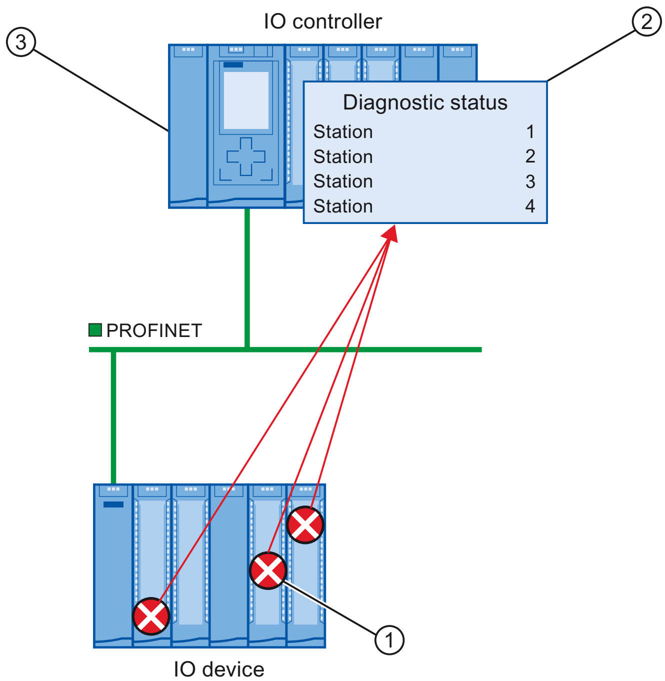


Number	Description
①	All individual errors are collected in a single data record on the interface module.
②	In your user program, the instruction "RDREC" reads the entire station status asynchronously directly from the IO device.

Figure 5-15 Example: Evaluating diagnostics data records with the instruction "RDREC"

2. Evaluation of interrupts

When the error OB (OB 82) is called, the OB's start information provides you with information on the cause and location of the error. Detailed information on the error event can be obtained in the error OB using the instruction "RALRM" (read additional interrupt information).



Number	Description
①	Every error is sent to the IO controller individually as channel diagnostics information in the form of an interrupt.
②	In the IO controller, the module status data is updated automatically and the error OB (OB 82) is started.
③	In your user program in the error OB (OB 82), the instruction "RALRM" reads the error synchronously from the IO controller without addressing the IO device.

Figure 5-16 Diagnostics with OB 82 and the instruction "RALRM"

Instructions and OBs

You will find information on the instructions and OBs in the STEP 7 online help.

5.9 Maintenance

5.9.1 I&M data (identification and maintenance)

Definition and properties

Identification and maintenance data (I&M) is information saved to module memory in order to provide support when:

- Checking the plant configuration
- Locating hardware changes in a plant

Identification data (I data) is module information (some of which may be printed on the module housing) such as the order and serial number. I data is read-only vendor-specific module data.

Maintenance data (M data) is plant-specific information such as the location identifier and installation date. M data is created during configuration.

The modules can be uniquely identified in online mode by means of the I&M data.

Further information

To find out whether and to what extent a PROFINET device supports I&M data, refer to the documentation of the relevant device.

5.9.2 Loading I&M data to PROFINET IO devices and your modules

Which I&M data can be loaded to PROFINET IO devices and your modules?

You can load I&M 1 data (plant designation and location identifier) and/or I&M 2 data (installation date) and/or I&M 3 data (additional information) to the actual hardware.

Requirements

- In the project settings (Options > Settings, Hardware configuration > Compiling and downloading), the option "Download I&M data" must be enabled.
- There is an online connection to the PROFINET IO devices and the modules to which you want to load I&M data.
- You have entered the I&M data you want to download in the properties of the respective PROFINET IO devices and your modules (Inspector window: "Properties" tab > "General" tab, Settings > Identification & Maintenance).

Where do I specify which I&M data is downloaded to which PROFINET IO devices?

You specify which I&M data you want to download to which PROFINET IO devices in the "Load preview" dialog. You will find the following alternatives in the drop-down list of the "Identification and maintenance data (I&M)" row:

- Load nothing
  The check boxes for all PROFINET IO devices as well as the check boxes for the loadable I&M data are cleared.
  No I&M data is transferred to the actual hardware during loading with this setting.
- Load data
  The check boxes for all PROFINET IO devices as well as the check boxes for the loadable I&M data are selected.
  The respective I&M 1, I&M 2 and I&M 3 data is transferred to all PROFINET IO devices during loading with this setting.
- Load selected
  You select the check boxes of those PROFINET IO devices to which you want to load I&M data. You also select the check boxes of the identification data you want to load.
  With this setting, you transfer the selected I&M data to the selected PROFINET IO devices during loading.

NOTE

Language dependency of the I&M data to be loaded

The I&M data are loaded to the real hardware in the form that you specified in the properties of the relevant PROFINET IO devices and your modules. There is no language dependency.

5.9.3 Asset management

5.9.3.1 Further information about asset management at PROFINET

Operation of machines and plants without a detailed knowledge of devices and assets is difficult to imagine.

Maintenance requires data for this – data that is extensive and up-to-date as far as possible.

The requirement of greater transparency with regard to the data made available by plant components has been fulfilled by PROFIBUS & PROFINET International (PI): The identification and maintenance data familiar since PROFIBUS times have been extended.

The current PROFINET fulfills this requirement through the definition of a special data record: the Asset Management Record (AMR).

The aim of this definition is to enable you to acquire **all** the components to be maintained online - and not only those components that can be addressed and accessed through the PROFINET device model (Device / Module / Submodule). PROFINET now also reaches non-PROFINET components!

Assets general and asset management data records

Assets are components (hardware and software / firmware) of a machine, for example a laser welding machine, or a plant.

A large number of these device components can already be identified through tried-and-tested I&M functions or corresponding data records (I&M0 to I&M5) in the PROFINET context: The device itself as well as its modules and submodules. Meaning all components that can be addressed through the PROFINET device model.

Components that **cannot** be addressed via the PROFINET device model, but whose data it should be possible to acquire online for operation and maintenance, can be identified through **asset management** functions. This asset management data (short: AM data) is stored in a defined structure in a special data record, the **asset management record** (AMR) mentioned above.

The PROFINET Guideline "Identification & Maintenance Functions" differentiates here between I&M functions (I&M data) and asset management functions (AM data): The following sections only deal with the AM data.

The components that can be read additionally online through asset management data records include both hardware components, such as backplane bus modules of a device as well as firmware components such as a drive control unit with own versioning.

Application examples

Importing asset management records enables you to read the following information during installation or operation, for example:

- Are only approved devices being used (whitelist check)?
- A firmware update is due. Obtain a fast overview: Which devices or components are affected and have to be upgraded?

Making asset management data available

The concept for the asset management of PROFINET devices stipulates that the manufacturers of PROFINET devices have to ensure that non-PROFINET automation components are made available through an asset management record. This data record is assigned to the PROFINET device.

In contrast to a "standard" IO device, with an I-device the project engineer has to make the asset management record available. In this case, the central modules of the I-device are assets. From the perspective of PROFINET, these central modules are not visible from the point of view of the higher-level IO controller. The higher-level IO controller only "sees" the transfer areas through which it exchanges IO data with the I-device.

The principle of this provision is explained in the section.

Reading asset management data

The asset management record has the index 0xF880 and is read with standard PROFINET mechanisms by the user of the records, for example a tool or program for evaluating these data.

A user program in the S7-1500 IO controller, for example, can read out the AMR of an IO device with the RDREC instruction (Index 0xF880).

It is not possible to write to this data record.

Further information

Whether and to which extent a PROFINET device supports asset management data, meaning whether it makes an AMR available, is specified in the documentation of the respective device.

5.9.3.2 Content and structure of an asset management record

Basic structure of the asset management record

You are first provided with an overview of the general structure of the record. The following table describes the framework within which the asset management data blocks are embedded. Each data block represents an asset, a terminal block for example.

Element of the data structure	Designation according to IEC 61158-6-10	Code	Data type / length in bytes
Header AssetManagementData	BlockType	0x0035	UINT / 2
	BlockLength	Number of bytes without counting the bytes for BlockType and BlockLength	UINT / 2
	BlockVersion	0x0100	UINT / 2
AssetManagementInfo AssetManagementBlocks (n)	NumbersOfEntries	Number of AssetManagementBlocks	UINT / 2
	AssetManagementBlock 1	See the table below	
	AssetManagementBlock 2		
	...		
	AssetManagementBlock n		

Structure of asset management blocks

Each AssetManagementBlock contains identification data and localization information for an asset. An AssetManagementBlock has a substructure with basic characteristics described below.

The header of an AssetManagementBlock contains the coded information about which of the three possible AM data compilations the data record contains. Devices make a suitable BlockType available in accordance with the various device types:

- Complex devices with information about the hardware and firmware (BlockType "AM_FullInformation")
- Complex devices with information about the hardware (BlockType "AM_HardwareOnlyInformation")
- Devices with information about the firmware (BlockType "AM_FirmwareOnlyInformation")

The differentiation provides an efficient data structure below the header. Nevertheless, the data record can have a considerable size (maximum of 64 KB, depending on the number of assets that the IO device supplies).

Table 5-4 Structure of AssetManagementBlock

Element of the data structure	Designation according to IEC 61158-6-10	Code	Data type / length in bytes
Header AssetManagementBlock	BlockType	0x0036 (AM_FullInformation) 0x0037 (AM_HardwareOnlyInformation) 0x0038 (AM_FirmwareOnlyInformation)	UINT / 2
	BlockLength	Number of bytes without counting the bytes for BlockType and BlockLength	UINT / 2
	BlockVersion	0x0100	UINT / 2
	Padding	0x0000 (padding byte)	USINT / 1
	Padding	0x0000 (padding byte)	USINT / 1
AssetManagementBlock (Structure depends on the BlockType. Here it is shown using AM_FullInformation as an example)	IM_UniqueIdentifier	Manufacturer-generated Universal Unique Identifier (UUID) conforming to ISO/IEC 9834-8. Used as a reference key to uniquely identify this asset. Example: 550c5300-d34a-22b4-11d3-5533991111b3	Array of Byte / 16
	AM_Location	Description of the location of the asset: Either slot-oriented ("Slot and SubslotNumber format") or hierarchical ("Twelve level tree format"). See following description	Array of Byte / 16
	IM_Annotation	Manufacturer-specific notation Example: "Terminal block, Type xyz123 ". 64 bytes are always used. Spaces are used for padding if the string is shorter.	Array of Char / 64
	IM_OrderID	Manufacturer-specific article number Example: "6ES7 131-6BF00-0BA0 ". 64 bytes are always used. Spaces are used for padding if the string is shorter.	Array of Char / 64
	AM_SoftwareRevision (not at AM_HardwareOnlyInformation)	Manufacturer-specific SW version Example: "V6.3.8 ". 64 bytes are always used. Spaces are used for padding if the string is shorter. If the asset supports IM_Software_Revision, the AM_SoftwareRevision is padded with spaces.	Array of Char / 64
	AM_HardwareRevision (not at AM_FirmwareOnlyInformation)	Manufacturer-specific hardware version Example: "A4 ". 64 bytes are always used. Spaces are used for padding if the string is shorter. If the asset supports IM_Hardware_Revision, the AM_HardwareRevision is padded with spaces.	Array of Char / 64

Element of the data structure	Designation according to IEC 61158-6-10	Code	Data type / length in bytes
AssetManagementBlock (Structure depends on the BlockType. Here it is shown using AM_FullInformation as an example)	IM_Serial_Number	Manufacturer-specific unique production-related number. The characters come from the visible range (0x20 ... 0x7E), no control characters. Example: "A78C-1C82 ". 16 bytes are always used. Spaces are used for padding if the string is shorter.	Array of Char / 16
	IM_Software_Revision (not at AM_HardwareOnlyInformation)	Software version, follows a strict structure (SW version prefix, for example "V", digits for functional extension, digits for BugFix, digits for internal change). Example: 'V' 0x01 0x2 0x3 If AM_SoftwareRevision is padded with spaces, you should evaluate IM_Software_Revision. If the asset does not support any hardware, the coding 'V' 0x00 0x00 0x00.	Array of Byte / 4 Prefix (character "V", "R", "P", "U", or "T"), then 3 digits "0" to "9"
	AM_DeviceIdentification	Identification of the device. The structure is as follows: AM_DeviceIdentification.DeviceSubID (for SIEMENS e.g. 0x0000) AM_DeviceIdentification.DeviceID (Device ID from manufacturer, 0x0000 to 0xFFFF) AM_DeviceIdentification.VendorID (Example for Siemens assets: 0x002A) AM_DeviceIdentification.Organization: Example for Siemens assets: 0x0000 (PROFINET)	Array of Byte / 8
	AM_TypeIdentification	Manufacturer-allocated type identification: 0x0000: Unspecified 0x0001: Controller (PLC) 0x0002: PC-based 0x0003: IO module, IO submodule 0x0004: Communications module / submodule 0x0005: Interface module / submodule 0x0006: Active network component 0c0007: Media attached unit (bus adapter) 0x0100 to 0x7FF: Manufacturer-specific	UINT / 2
	IM_Hardware_Revision (not at AM_FirmwareOnlyInformation)	Version of the hardware (0x0000 to 0xFFFF) Example: 0x0003 If AM_HardwareRevision is padded with spaces, you should evaluate IM_Hardware_Revision.	UINT / 2

AM_Location

Asset management at PROFINET supports two formats for coding the location of an asset:

- Slot-oriented format ("Slot and SubslotNumber format")
- Hierarchical format ("Twelve **L**evel **T**ree format" abbreviated **LT** format)

Assets that are part of the PROFINET device use the slot-oriented format. These assets are bound completely to the PROFINET modules and submodules.

Assets that are located outside the PROFINET device use the hierarchical format (LT format) for coding the location of an asset.

These assets are localized by their tree level. The tree level begins with Level 0. The value of Level 0 provides information about the proximity to the PROFINET device:

- If the asset is connected to a module that can be addressed through the PROFINET device model, Level 0 has the value 0. The subsequent levels (Level 1 to Level 3) then have the meaning of slot address, subslot address and channel number. If further assets are connected to this asset, the next Level 4 is used. The limit is reached at Level 11.
- If the asset belongs to a PROFINET device but is not connected to a module that can be addressed through the PROFINET device model, Level 0 has a value between 1 and 0x1FF. An example of such an asset is a power supply unit in the PROFINET device. If a further asset is connected to this power supply unit, for example a sensor, the next tree level is used to localize this sensor (Level 1).
- If the asset is located outside the PROFINET device, but, for example, belongs to a machine into which the PROFINET device is installed, Level 0 has a value between 0x200 and 0x3FE.

The value 0x3FF for a tree level shows that this tree level is not used. This means that no further asset is connected. In this case, all the lower tree levels down to Level 11 must also have this value.

Example AM_Location slot-oriented

A rack and the terminal blocks located on it each supply AM data. The slot assignments are shown in the figure.

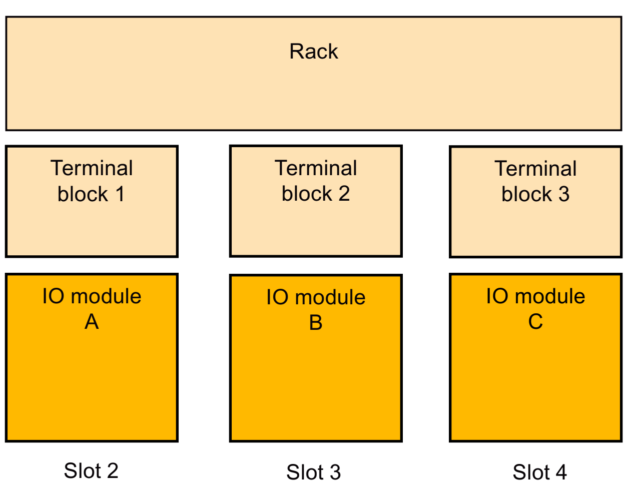


Figure 5-17 Example of assets with slot-oriented AM_Location coding.

Code the AM_Location as an asset for each module as follows:

Bit 0 – 7: AM_Location.Structure = 0x02 (coding "Slot and SubslotNumber format")

Bit 8 – 15: AM_Location.Reserved1 = 0x00 (padding byte)
Bit 16 – 31: AM_Location.BeginSlotNumber = 2 (the "Rack" asset begins from Slot 2 on)
Bit 32 – 47: AM_Location.BeginSubslotNumber = 0xFFFF (the asset encompasses all the subslots of Slot 2. Otherwise you specify the no. of the subslot at which the asset begins)
Bit 48 – 63: AM_Location.EndSlotNumber = 4 (the asset ends at Slot 4)
Bit 64 – 79: AM_Location.EndSubslotNumber = 0xFFFF (the asset encompasses all the subslots of Slot 4. Otherwise you specify the no. of the subslot at which the asset ends)
Bit 80 – 95: AM_Location.Reserved2 = 0x0000 (padding byte)
Bit 96 – 111: AM_Location.Reserved3 = 0x0000
Bit 112 – 127: AM_Location.Reserved4 = 0x0000

Example AM_Location level-oriented

A complex sensor is connected to an IO module (Slot 5, Subslot 1, Channel 1). Two simple sensors, in turn, are connected to the complex sensor. The module can be addressed within the PROFINET device model. Level 0 therefore has the value 0x0000. The next level (Level 1) is specified by the assigned slot. This is followed by the further levels for the subslot and channel and, if appropriate, further subordinate layers.

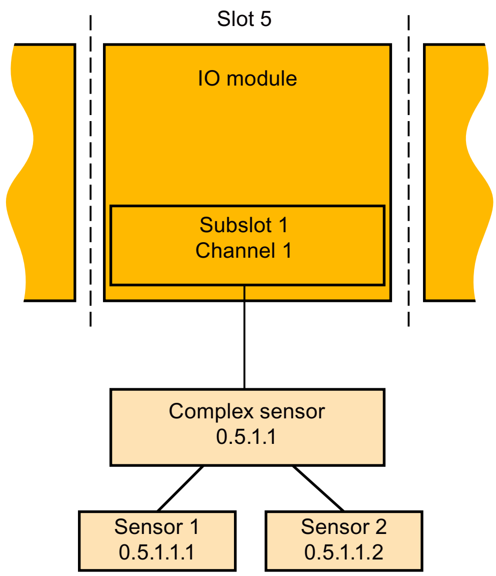


Figure 5-18 Example of assets with hierarchical AM_Location coding.

Detailed coding for the example:
Bit 0 – 7: AM_Location.Structure = 0x01 (LT format)
Bit 8 – 17: AM_Location.Level0 = 0x000 (assets that are assigned to modules always have the Level 0 value 0x000)
Bit 18 – 27: AM_Location.Level1 = 0x005 (Slot 5)
Bit 28 – 37: AM_Location.Level2 = 0x001 (Subslot 1)
Bit 38 – 47: AM_Location.Level3 = 0x001 (Channel 1)
Bit 48 – 57: AM_Location.Level4 = 0x3FF (coding for "Level not used")
Bit 58 – 67: AM_Location.Level5 = 0x3FF (coding for "Level not used")
...
Bit 118 – 127: AM_Location.Level11 = 0x3FF (coding for "Level not used")
Notation used in the screen for the LT coding of complex sensors: 0.5.1.1

The following correspondingly applies for the remaining sensors:
LT coding for simple Sensor 1 at complex sensor: 0.5.1.1.1
LT coding for second simple Sensor 2 at complex sensor: 0.5.1.1.2

5.9.3.3 Asset management data record for I-devices

With STEP 7 (TIA Portal) as of V15 and with S7-1500 CPUs as of firmware V2.5.2, you can compile an asset management data record via a user program. Configured as an I-device, these CPUs then supply the data from centrally plugged modules to a requesting IO controller as assets.
"S7-1500 CPUs" also refers to the CPU variants S7-1500F, S7-1500T, S7-1500C, S7-1500 SW Controller, S7-1500pro CPUs and ET 200SP CPUs.

Asset management records for I-devices

I-devices often represent machines. The PROFINET IO controller to which the I-device is assigned only sees the PROFINET interface (also configured as an IO device) and the transfer areas of the I-device configured by the machine manufacturer. The local modules of the I-device are not visible or cannot be accessed.
The assigned IO controller can read the central modules as assets of the I-device by means of an asset management record that the user program of the I-device compiles.

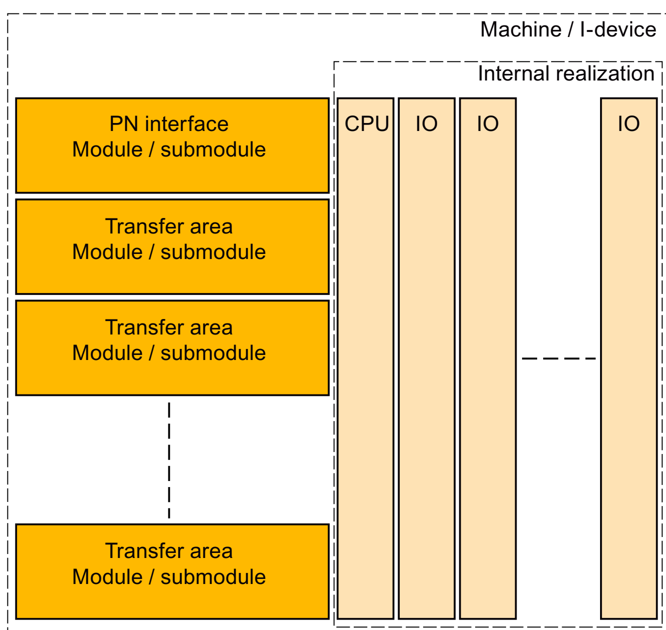


Figure 5-19 Assets of an I-device

Requirement

- S7-1500 CPU as of firmware V2.5.2, configured as I-device
- STEP 7 (TIA Portal) as of V15
- If an IO controller is to read the asset management record:
  The PROFINET IO controller is programmed correspondingly to read an asset management record.
  For a SIMATIC IO controller, for example, you call a read instruction (RDREC) with record index 0xF880. The instruction addresses any submodule of the I-device, for example the first configured transfer area submodule.

Basic procedure

The following steps are fundamentally required to create the requirements so that an I-device can make its local modules available as an asset management record to a requesting IO controller:

1. Make the settings in the properties of the PROFINET interface of the CPU.
   – Activate "IO device" operating mode
   – Activate the "Activate asset management using user program" option
   PROFINET interface forwards a request of an IO controller to the user program of the I-device for reading the asset management record only if the option is selected.

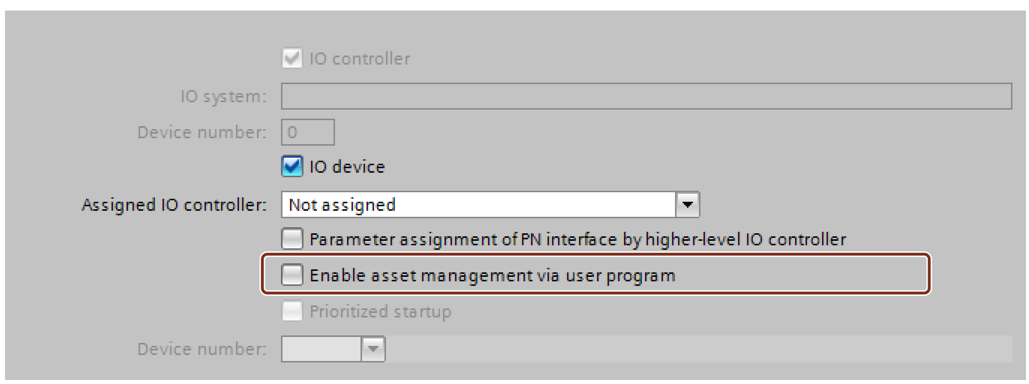


Figure 5-20 Activating asset management using a data record

2. Configure the program routine for compiling the asset management record. The program part collects the required I&M0 data of the plugged central modules and stores them in the corresponding fields of the data record structure of the asset management record.
3. Configure the program part for coordinating the data record provision:
   For this, call the instruction PRVREC (Provide Record) in accordance with the following templates in the corresponding modes:
   – Cyclic calling (for example in the cycle OB) of the PRVREC instruction with Mode 0, in order to recognize the AMR request.
   – When the AM record request is recognized, the PRVREC program has to acknowledge within one second that the request has been recognized. This means that PRVREC must be called with the Mode 2, and with the required AM record. If the I-device does not adhere to the time frame, the I-device acknowledges the record request of the IO controller as negative.

Particular aspect for configuration of the PRVREC call: PRVREC has to be called with **F_ID = 0**. This codes that this is an IO-device-specific data record. The SLOT and SUBSLOT output parameters therefore also return the value 0.

– Within 10 seconds the AM record now has to be completed and PRVREC be called with Mode 3 (positive response to the IO controller with provision of the AM record). If the I-device does not adhere to the time frame, the I-device acknowledges the record request of the IO controller as negative.

A detailed description of the PRVREC instruction and possible error codes for evaluating the function can be found in the online help of STEP 7 (TIA Portal).

Compilation of the asset management record

You have various possibilities for compiling the asset management record for an I-device:

- Recommendation: The Siemens Industry Online Support makes an application available to you that helps you to compile the asset management record.
  The data area of the asset management record is divided in two. The first part consists of an automatically determined area that packages the IM0 data of the slots of the I-device into an asset management block. The second part consists of the user-specific asset management blocks. You configure the user-specific asset management blocks based on pre-configured asset management record structures, fill them with information and make them available to the application.
  The application performs the following:
  – The application determines the required size of all the asset management blocks.
  – The application fills the data block in accordance with the specifications of an asset management record with the automatically determined asset management blocks and your user-specific asset management blocks.
  – The application makes this asset management record available to the higher-level IO controller.

  The application is described in this application example (https://support.industry.siemens.com/cs/ww/en/view/109748894)
- You create the asset management record yourself.
  The following section describes how you can compile an asset management record for an I-device yourself.
  The concept assumes you yourself determine the I&M data for each centrally plugged module and fill the asset management record with this information. The I&M0 data of a module contains basic information about the module such as the manufacturer's code, article number, serial number, hardware and firmware version. These are the data that are also required in the AM record for an asset.

Determine the I&M data of centrally plugged modules

The central structure consists of an optionally plugged power supply unit (Slot 0), followed by the I-device CPU (Slot 1), and then followed by the further modules, such as digital modules, analog modules, etc. (as of Slot 2).

You determine the I&M data with the "Get_IM_Data" instruction for the plugged modules with exception of the CPU:

To assign parameters for the "Get_IM_Data" instruction, you require the hardware identifier (LADDR input parameter). You determine the hardware identifier for each occupied slot with the "GEO2LOG" instruction (Determine hardware identifier from slot).

Summary of the theoretical steps:

1. In a loop, determine the hardware identifiers of the plugged modules with the "GEO2LOG" instruction.
2. For each hardware identifier found, determine the I&M data by using the "Get_IM_Data" instruction and store these data in a data block that you address with the input parameter DATA. Use ARRAY of BYTE for the data storage. This corresponds to the description of the AM record contents in the preceding section.

Forming an AM record with the determined I&M data

The following sections are based on the description of the fundamental structure of the AM record, see the preceding section.

Since each module of an S7-1500 contains hardware and firmware information, select the coding for "AM_FullInformation" for the assigned BlockType.

For the data types used:

- IM_Annotation, IM_OrderID, AM_SoftwareRevision and AM_HardwareRevision: Characters (UTF-8)
- IM_Serial_Number: Characters ("ASCII characters") with the exception of the characters for DEL (0x7F)
- Do not use String data types. They require additional bytes and therefore do not conform to the conventions of the PROFINET standard 61158-6-10 "Application layer protocol specification".

Form the AM_FullInformationBlock for each module as follows:

Table 5-5 AM_FullInformationBlock for modules

Data record element	Description
IM_UniqueIdentifier	Generate a (pseudo) random UUID (hash value) in accordance with ISO 9834-8 as follows: • Generate an 8-byte hash value across the I&M0 data of the module (as of Slot 2). Use the algorithm Fowler-Noll-Vo (in short: FNV); an algorithm for generating variance coefficients (hash values) across data field, see corresponding example code in the Internet or online support. • Generate an 8-byte hash value across the I&M0 data of the CPU. (Use the algorithm Fowler-Noll-Vo (in short: FNV) as described above) • IM_UniqueIdentifier Byte 0 to 7: Hash value of module I&M0 data Bytes 8 to 15: Hash value for CPU-I&M0 data Required customizations to ISO 9834-8: Byte 8, Bit 7 has to be set to 1, and Byte 8, Bit 6 to 0 (result of the AND operator with **00**11 1111, subsequent OR operator with **10**00 0000) Byte 6, Bit 4 to 7 have to be set to 0100 (result of the AND operation with 0000 1111, then OR operation with 0001 0000)

Data record element	Description
	Since this algorithm is based on the I&M0 data of the CPU as well as of the modules, it generates a constant IM_UniqueIdentifier for an individual module. When the configuration changes the IM_UniqueIdentifier also changes.
AM_Location	Byte 0 = 0x02 (slot-oriented coding), see description in the preceding section.
IM_Annotation	Example: "S7-1500 module" and pad the remaining bytes of IM_Annotation with spaces (0x20).
IM_OrderID	Copy 20 bytes of the I&M0 data of the module (beginning with offset 2 of the I&M0 data). Pad the remaining 44 bytes with spaces (0x20)
AM_SoftwareRevision	Pad the field with 64 spaces (0x20)
AM_HardwareRevision	Pad the field with 64 spaces (0x20)
IM_Serial_Number	Copy 16 bytes of the I&M0 data of the module (beginning with offset 22 of the I&M0 data)
IM_Software_Revision	Copy 4 bytes of the I&M0 data of the module (beginning with offset 40 of the I&M0 data)
AM_DeviceIdentification	Byte 0, 1, 2, 6, 7 = 0x00 Byte 3 = 0x2A (Vendor = Siemens) Byte 4 = 01, Byte 5 = DeviceID (e.g. CPU 15xx = 0x0E)
AM_TypeIdentification	Copy 2 bytes of the I&M0 data of the module (beginning with offset 48 of the I&M0 data)
IM_Hardware_Revision	Copy 2 bytes of the I&M0 data of the module (beginning with offset 38 of the I&M0 data)

Functions

6

Introduction

The following section describes the PROFINET IO functions for field of application, properties and configuration in STEP 7.
Refer to the documentation for the respective device to see to what extent the PROFINET devices support the described functions.
You can find a tabular overview of the PROFINET devices and the functions these support on the Internet (https://support.industry.siemens.com/cs/ww/en/view/102325771).

6.1 Connecting other bus systems

Fieldbus integration

PROFINET allows you to use a proxy-capable PROFINET device to integrate existing fieldbus systems (for example, PROFIBUS, AS interface). The devices of these fieldbus systems are mapped on proxy PROFINET devices. In this way, you can set up any hybrid systems consisting of fieldbus and Ethernet-based subsystems. This allows a continuous exchange of information.

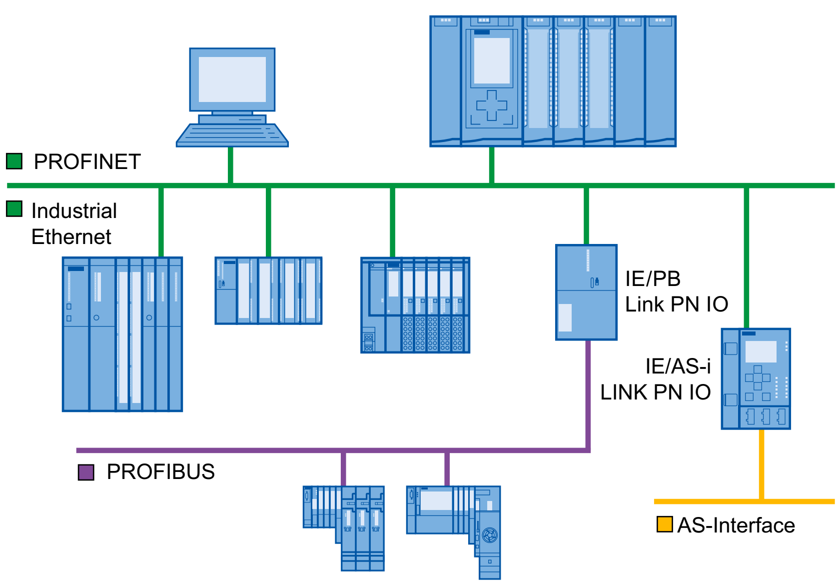


Figure 6-1 Gateways on PROFINET IO

Gateways of an S7-1500 CPU

An overview of the gateways at an S7-1500 CPU is provided in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/88778900).

Connecting to building busses

- BACnet: You can find the procedure for configuring communication between a SIMATIC PLC and a BACnet network with a gateway in this application example (https://support.industry.siemens.com/cs/ww/en/view/109476182).
- M-bus: You can find the procedure for configuring communication between a SIMATIC PLC and an M-bus network with a gateway in this application example (https://support.industry.siemens.com/cs/ww/en/view/109478527).
- DALI: You can find the procedure for configuring communication between a SIMATIC PLC and a DALI network with a gateway in this application example (https://support.industry.siemens.com/cs/ww/en/view/109740160).
- KNX: You can find the procedure for configuring communication between a SIMATIC PLC and a KNX network with a gateway in this application example (https://support.industry.siemens.com/cs/ww/en/view/109739689).

6.1.1 Linking PROFINET and PROFIBUS

Linking PROFINET and PROFIBUS

With a proxy-capable PROFINET device which is equipped with a PROFINET interface in addition to a PROFIBUS interface (for example, IE/PB Link PN IO), you can integrate existing PROFIBUS configurations into the PROFINET configuration.

The following figures shows how a PROFIBUS system is connected via IE/PB Link to a CPU S7-1500 (as of firmware version 1.7).

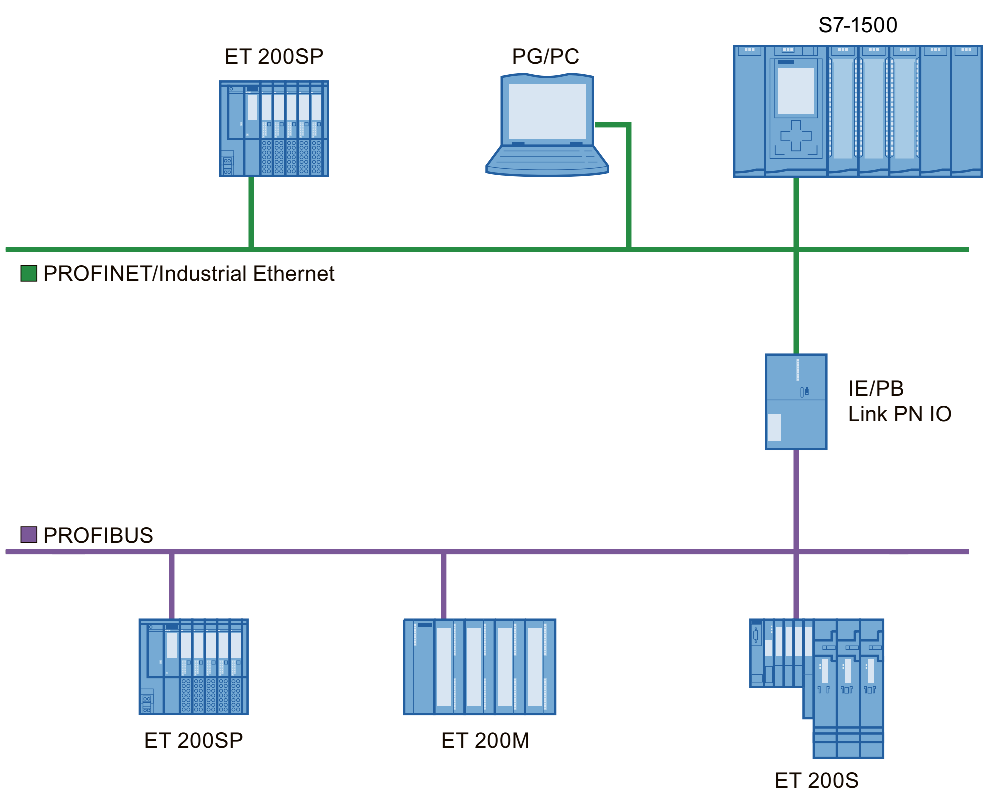


Figure 6-2 Gateway from PROFINET and PROFIBUS via IE/PB link

PROFINET device with proxy functionality

The PROFINET device with proxy functionality is the substitute for a PROFIBUS device on Ethernet. The proxy functionality allows a PROFIBUS device to communicate not only with its master but also with all devices on PROFINET.

With PROFINET you can connect an existing PROFIBUS system to an IO controller, for example with the help of an IE/PB Link PN IO.

From the IO controller perspective, the PROFIBUS DP slaves are connected to the same network as the IE/PB Link PN IO. These slaves have the same device name and IP address as the IE/PB Link PN IO, but different device numbers. Furthermore, each also has a specific PROFIBUS address.

In this way, you can link both DPV0 and DPV1 slaves to PROFINET.

For information on how to connect a DP slave to a PROFINET IO system, refer to section Connect the DP slave via the IE/PB Link to a PROFINET IO system (Page 119).

Diagnostic options with a CPU S7-1500 as IO controller

The CPU S7-1500 (as of firmware version 1.7) as IO controller detects disrupted DP slaves behind the IP/PB link.

6.1.2 Connect the DP slave via the IE/PB Link to a PROFINET IO system

Requirements

- STEP 7 as of V13 SP1
- CPU supports IE/PB link, e.g.:
  - S7-1500 CPU as of firmware version 1.7
  - S7-1500 Software Controller as of firmware version 1.7
  - S7-300/400 CPU

Procedure for connecting a DP slave via an IE/PB Link

To connect a DP slave to a PROFINET IO system via an IE/PB Link in STEP 7, follow these steps:

1. Drag-and-drop a PROFINET CPU, for example CPU 1513-1 PN, from the hardware catalog into the network view of STEP 7.
2. Drag-and-drop an IE/PB Link PN IO from the hardware catalog into the network view of STEP 7. The IE/PB Link PN IO is located under Network components > Gateways > IE/PB Link PN IO.
3. Assign the IE/PB Link PN IO to the CPU.
4. Drag a PROFIBUS interface module e.g. IM155-6 DP HF, from the hardware catalog to the network view.

5. Assign the interface module to the IE/PB Link.

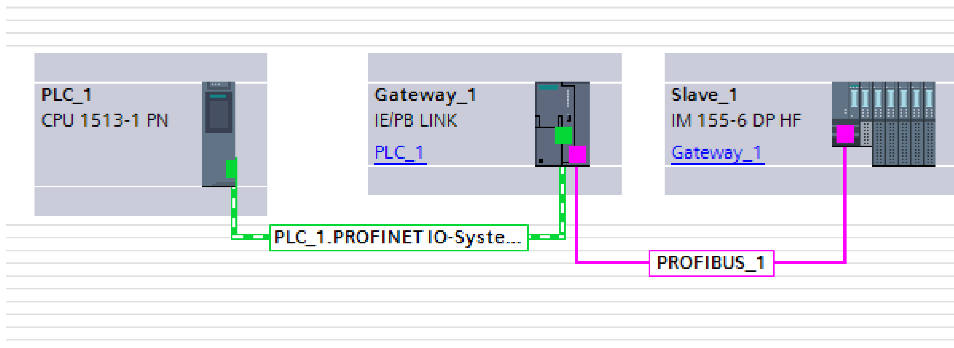


Figure 6-3 Configuring an IE/PB link

6. Select the IE/PB Link PN IO in the network view of STEP 7.
7. In the Inspector window, go to the "Gateway" area and select the "Network gateway as PROFINET IO proxy" option.

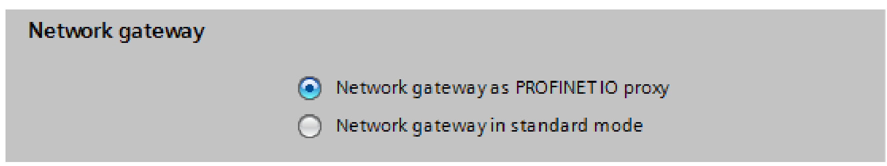


Figure 6-4 Setting a gateway

8. In the PROFINET device number area, you can assign a PROFINET device number for the DP slave.
   If you have selected the "Device number = PB address" check box (default), STEP 7 automatically assigns the device number according to the PROFIBUS address of the slave. In addition, you no longer need to update the device number if the PROFIBUS address changes.

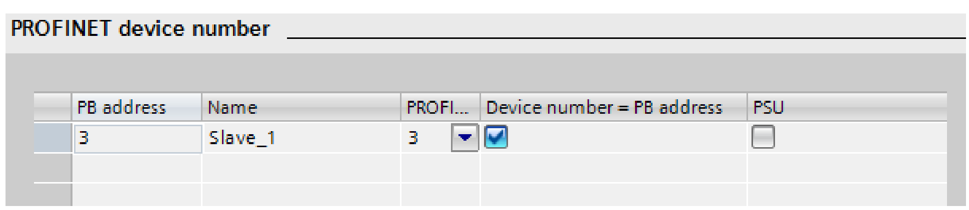


Figure 6-5 Assigning PN device numbers for IE/PB link

Result

You have connected the DP slave to the PROFINET IO system.

Reference

Additional information on the IE/PB link is available in the manual Gateway IE/PB Link PN IO (http://support.automation.siemens.com/WW/view/en/19299692).

6.2 Intelligent IO devices (I-devices)

6.2.1 I-device functionality

I-device functionality

The "I-device" (intelligent IO device) functionality of a CPU facilitates data exchange with an IO controller and operation of the CPU as intelligent preprocessing unit of sub processes, for example. The I-device is linked as an IO device to a "higher-level" IO controller.

The preprocessing is handled by the user program in the I-device. The process values acquired in the centralized or distributed (PROFINET IO or PROFIBUS DP) I/O are preprocessed by the user program and made available to the IO controller.

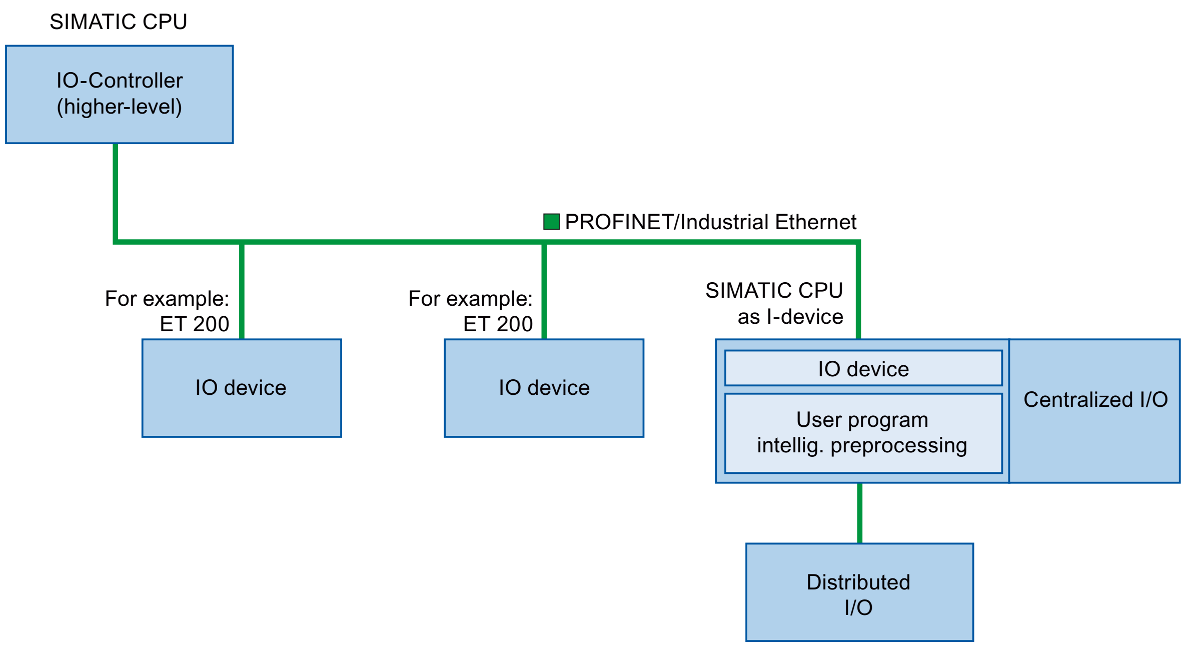


Figure 6-6 I-device

"I-device" naming conventions

In the remainder of this description, a CPU or a CP with I-device functionality is simply called an "I-device".

Application example: Configuration and application of the PROFINET I-device function

A detailed application example is available here (https://support.industry.siemens.com/cs/ww/en/view/109478798).

6.2.2 Properties and Advantages of the I-Device

Fields of application

Fields of application of the I-device:

- Distributed processing
  A complex automation task can be divided into smaller units/subprocesses. This results in manageable processes which lead to simplified subtasks.
- Separating subprocesses
  Complicated, widely distributed and extensive processes can be subdivided into several subprocesses with manageable interfaces by using I-devices. These subprocesses can be stored in individual STEP 7 projects if necessary, which can later be merged to form one master project.
- Know-how protection
  Components can only be delivered with a GSD file for the I-device interface description instead of with a STEP 7 project. The know-how of the user program may no longer be published.

Properties

Properties of the I-device:

- Unlinking STEP 7 projects
  Creators and users of an I-device can have completely separated STEP 7 automation projects. The GSD file forms the interface between the STEP 7 projects. This allows a link to standard IO controllers via a standardized interface.
- Real-time communication
  The I-device is provided with a deterministic PROFINET IO system via a PROFINET IO interface and therefore supports RT (real-time communication) and IRT (isochronous real time).

Advantages

The I-device has the following advantages:

- Simple linking of IO controllers
- Real-time communication between IO controllers
- Relieving the IO controller by distributing the computing capacity to I-devices.
- Lower communication load by processing process data locally.
- Manageable, due to processing of subtasks in separate STEP 7 projects

6.2.3 Characteristics of an I-Device

Principle

An I-device is included in an IO system like a standard IO device.

I-device without lower-level PROFINET IO system

The I-device does not have its own distributed I/O. 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).

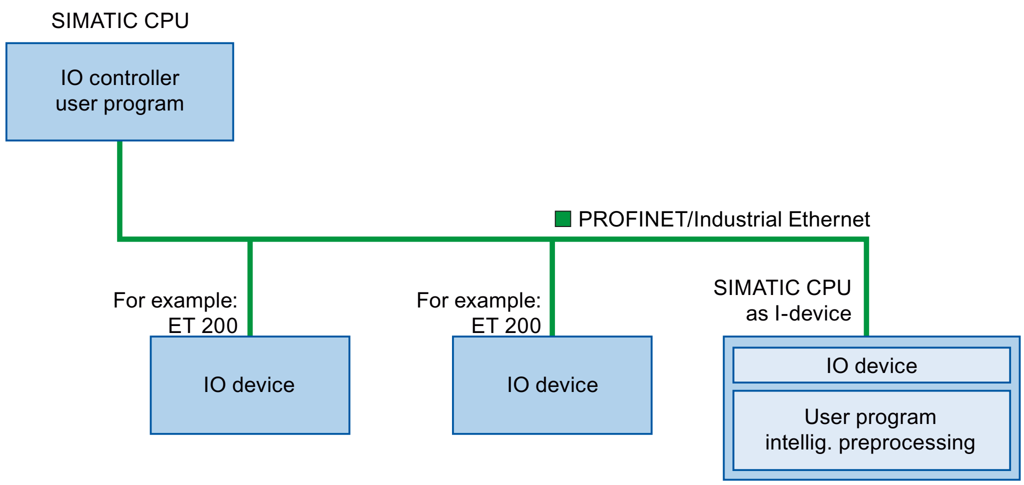


Figure 6-7 I-device without lower-level PROFINET IO system

I-device with lower-level PROFINET IO 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 an 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.

The lower-level IO system can, in turn, contain I-devices (see figure below). This makes hierarchically structured IO systems possible.

In addition to its role as IO controller, an I-device can also be used via a PROFIBUS interface as DP master for a lower-level PROFIBUS system.

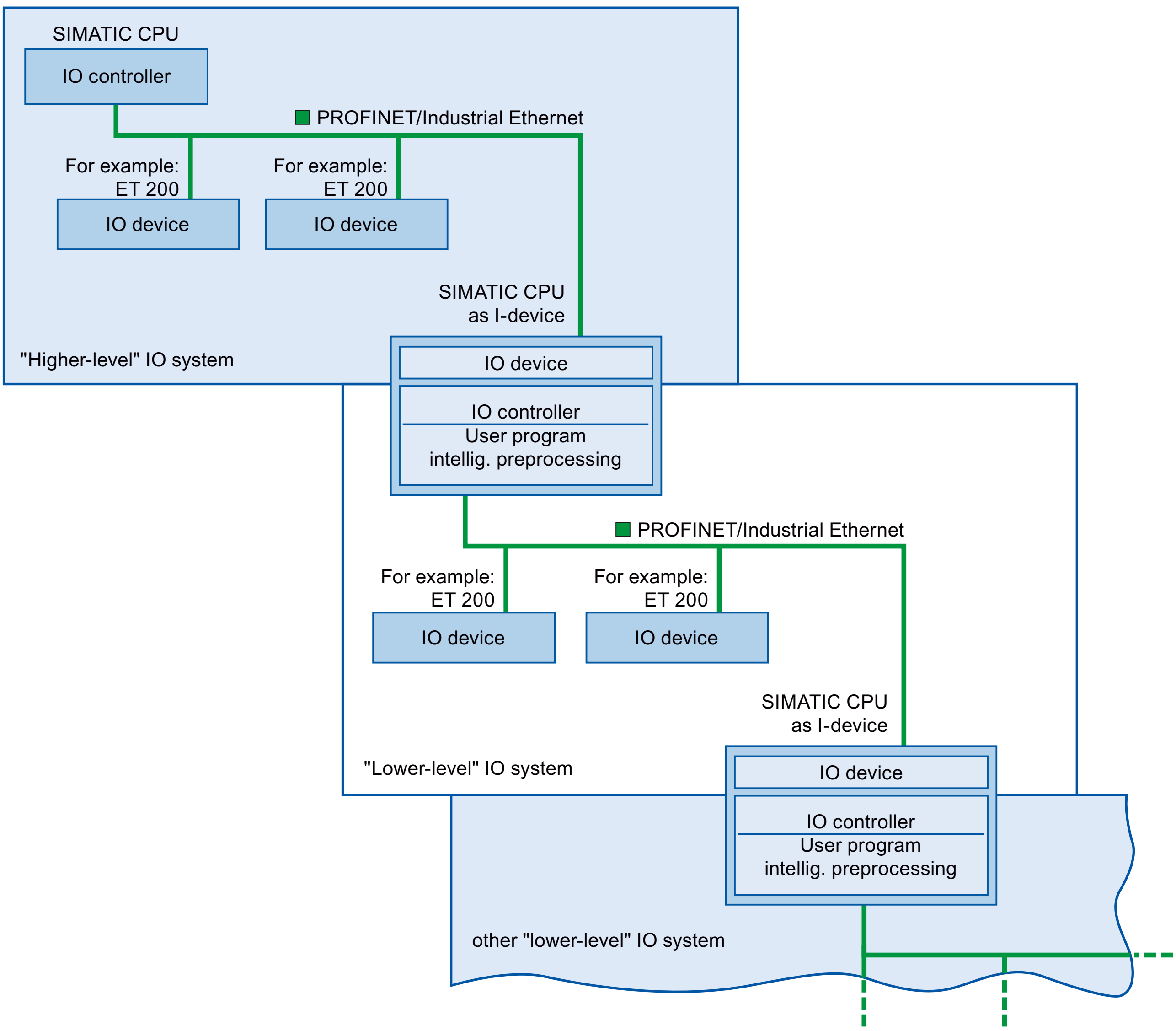


Figure 6-8 I-device with lower-level IO system

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.

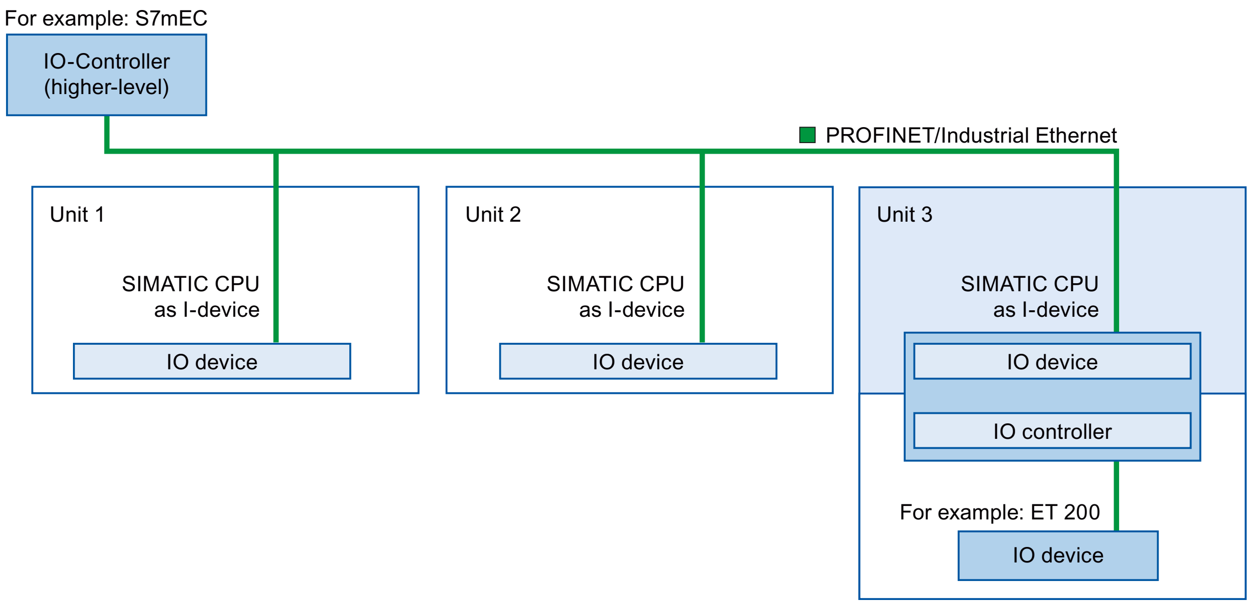


Figure 6-9 Example - the I-device as IO device and IO controller

Unit 1 and unit 2 each consist of an I-device with centralized 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).

I-device as a shared device

An I-device can also be used simultaneously by multiple IO controllers as a shared device.

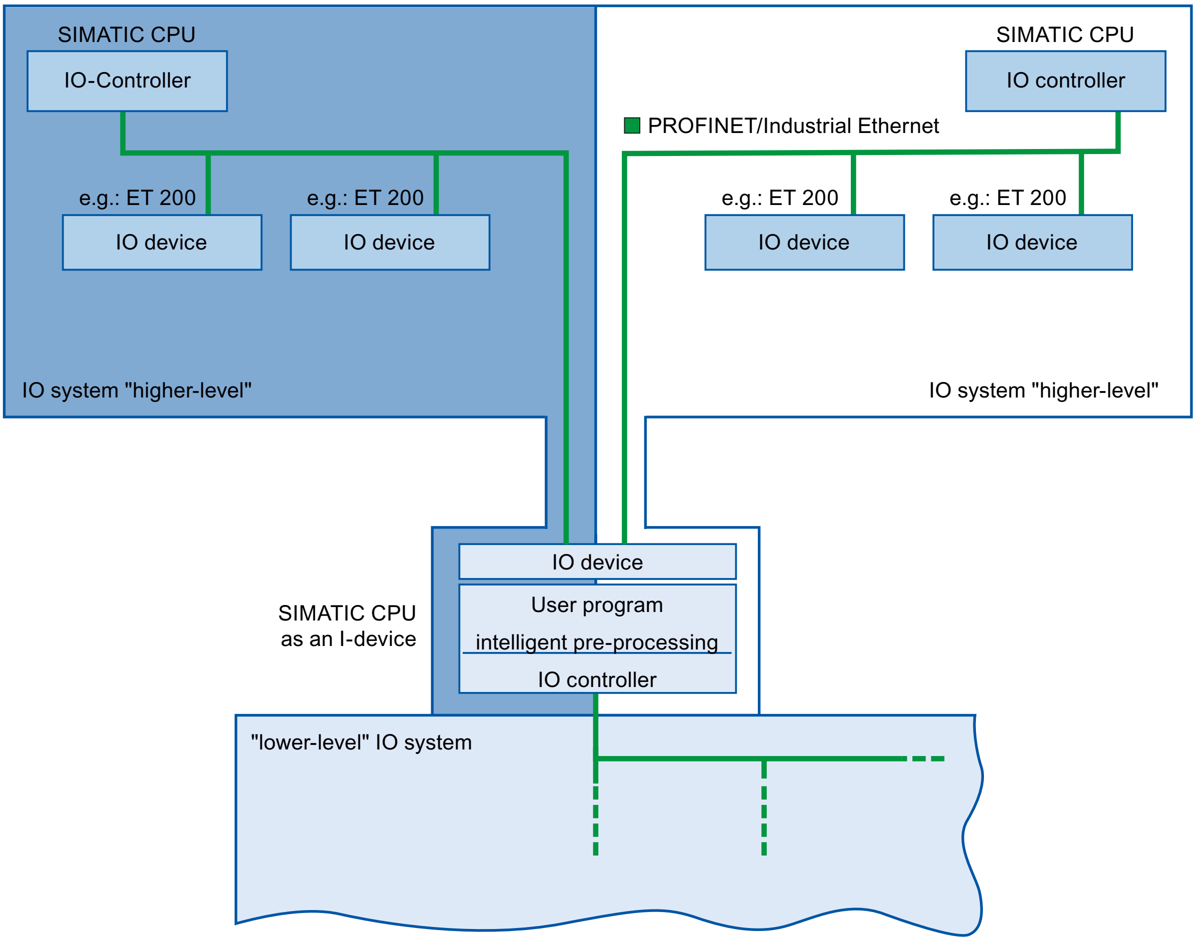


Figure 6-10 I-device as a shared device

Information about configuring an I-device as a shared device is available in the section Configuring a shared I-device in different projects (Page 166).

6.2.4 Data Exchange between higher- and lower-level IO system

Introduction

The next chapter shows the data exchange between the higher- and lower-level IO system.

Transfer areas

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.

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 can find more information on configuration and use of transfer areas in the section Configuring the I-device (Page 129).

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.

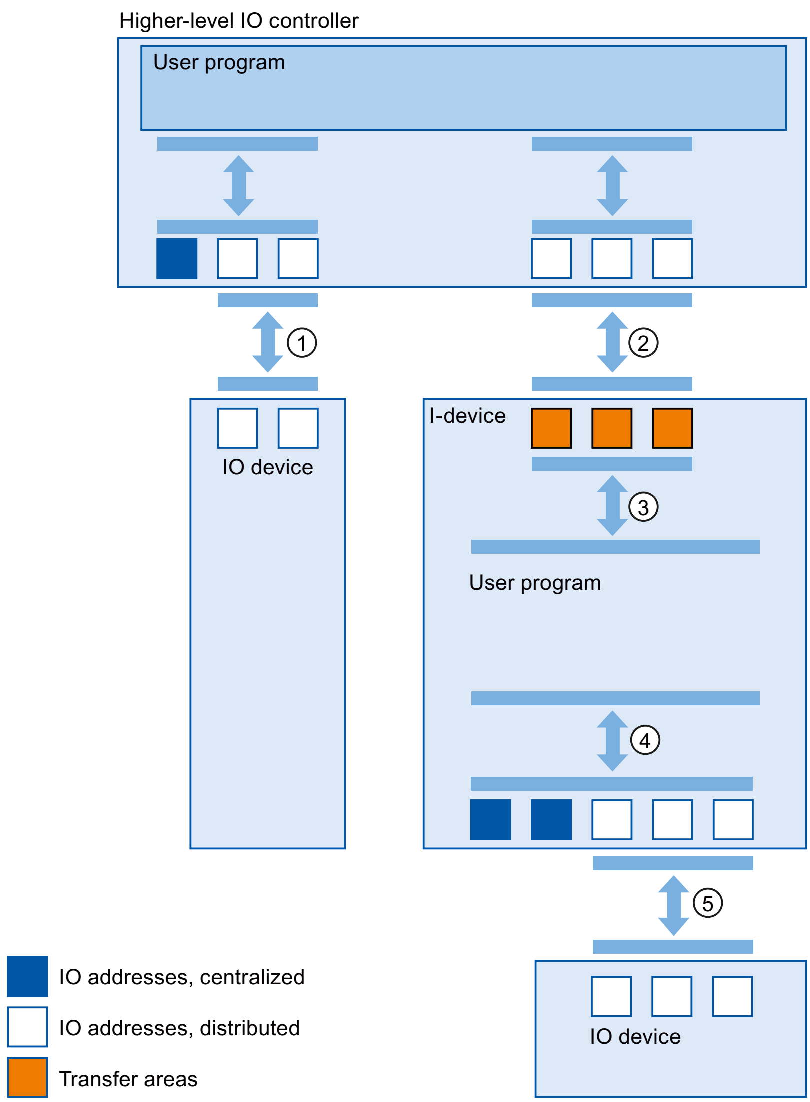


Figure 6-11 Data exchange between higher- and lower-level IO system

① Data exchange between higher-level IO controller and normal IO-device

In this way, the IO controller and IO devices exchange data through PROFINET.

② Data exchange between higher-level IO controller and I-device

In this way, the IO controller and the I-device exchange data through 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 relationship between the user program and the transfer area**

In this way, the user program and the transfer area exchange input and output data.

④ **Data exchange between the user program and the I/O of the I-device**

In this way, the user program and the centralized / distributed I/O exchange input and output data.

⑤ **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.

6.2.5 Configuring the I-device

Introduction

There are basically two possibilities for configuration:

- Configuration of an I-device within a project
- Configuration of an I-device that is used in another project or in another engineering system.

STEP 7 allows you to configure an I-device for another project or for another engineering system by exporting a configured I-device to a GSD file. You import the GSD file in other projects or engineering systems as with other GSD files. The transfer areas for the data exchange, among other data, are stored in this GSD file.

Configuration of an I-device within a project

1. Drag-and-drop a PROFINET CPU from the hardware catalog into the network view.
2. Drag-and-drop a PROFINET CPU, which can also be configured as an IO device, from the hardware catalog into the network view. This device is configured as I-device (e.g., CPU 1516-3 PN/DP).
3. Select the PROFINET interface for the I-device.
4. In the Inspector window in the area navigation choose "Operating mode" and select the check box "IO device".
5. For CPUs as of FW version V3.1:
   Click on "Assign new IO controller" in the "Assigned IO controllers" table, and select in the drop-down list the PROFINET interface of an IO controller that you want to assign to the I-device.
   For CPUs up to FW version V3.0:
   In the "Assigned IO controller" drop-down list, you now have the possibility to select the IO controller.

Once you have chosen the IO controller, the networking and the IO system between both devices are displayed in the network view.

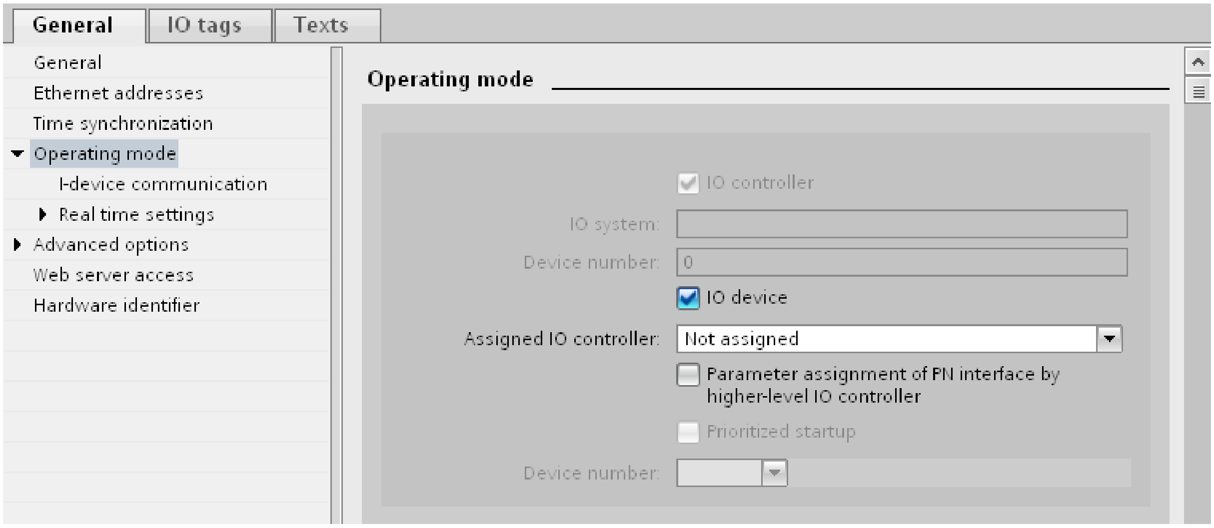
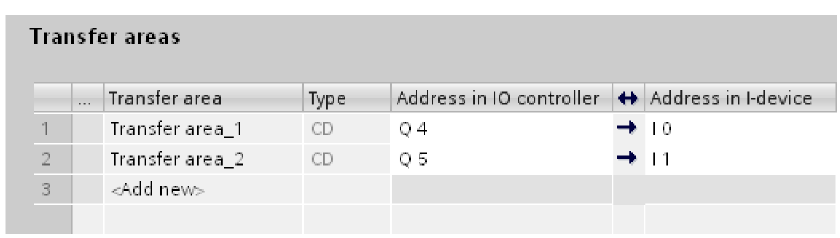


Figure 6-12 Configuring the I-device

6. With the "Parameter assignment of PN interface by higher-level IO controller" check box, you specify whether the interface parameters will be assigned by the I-device itself or by a higher-level IO controller.

   If you operate the I-device with a lower-level IO system, then the parameters of the I-device PROFINET interface (for example, port parameter) cannot be assigned with the higher-level IO controller.
7. Configure the transfer areas. The transfer areas are found in the area navigation section "I-device communication".
   – Click in the first field of the "Transfer area" column. STEP 7 assigns a default name that you can change.
   – Select the type of communication relation: you can currently only select CD or F-CD for "Controller-device communication relation".
   – Addresses are automatically preset; you can correct addresses if necessary, and determine the length of the transfer area which is to be consistently transferred.



Transfer areas

	...	Transfer area	Type	Address in IO controller	↔	Address in I-device
1		Transfer area_1	CD	Q 4	→	I 0
2		Transfer area_2	CD	Q 5	→	I 1
3		<Add new>				

Figure 6-13 Configuring the transfer areas

8. A separate entry is created in the area navigation for each transfer area. If you select one of these entries, you can adjust the details of the transfer area, or correct them and comment on them.

NOTE

Device replacement in the configuration

If you want to exchange a configured CPU (I-device) with FW version < V3.1 for a CPU with FW version V3.1, the new I-device cannot be used as a shared I-device within the project.

To use the new I-device as a shared I-device within the project, you must deselect the "IO device" option at the relevant PROFINET interface of the current device before changing the module.

Configuring an I-device with a GSD file

If you use an I-device in another project, or if the I-device is used in another engineering system, then configure the higher-level IO controller and the I-device as described above.

However, click on the "Export" button after configuring the transfer areas so a new GSD file is created from the I-device. This GSD file represents the configured I-device in other projects.

The "Export" button is found in the "I-device communication" section of the Inspector window.

The hardware configuration is compiled and the export dialog opened.

Assign a name for the I-device proxy as well as a description in the fields provided. Click the "Export" button to complete your process.

Finally, import the GSD file, for example, in another project.

6.2.6 Program examples

Introduction

This simple program example shows how you use the transfer areas of an I-device.

Requirements

You have configured an I-device.

Task

The result of an "AND logic operation" of two inputs (preprocessing) in the I-device is to be provided to the higher-level IO controller. This result is to be assigned to a local output in the IO master (further processing).

Use a transfer area with the following addresses for this purpose:

- Address in the I-device: Q568
- Address in the IO controller: I68

Required steps

The following steps to solve the task:

1. Configuring the transfer area
2. Programming I-device
3. Programming IO controller

Configuring the transfer area

Configure a transfer area with the following properties in the I-device:

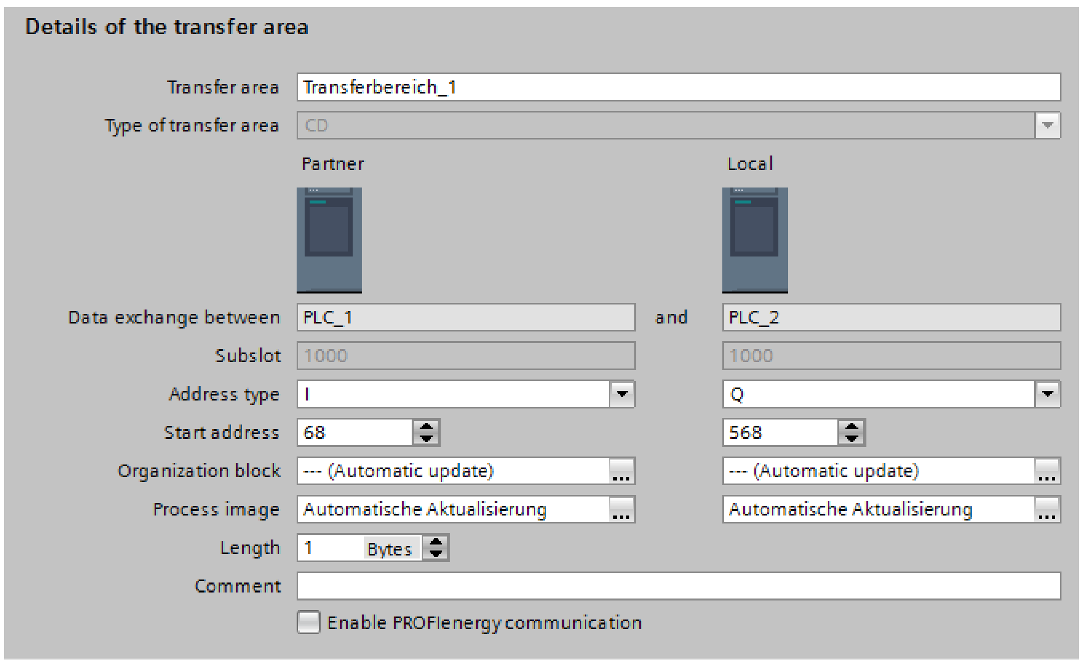


Figure 6-14 I-device transfer area, sample program

Programming I-device

To program the sample program for the I-device, follow these steps:

1. Using the SCL programming language, create a new function with the name "preprocessing" in the project tree in "Program blocks" > "Add new block". Open the function.
2. In the interface of the function "preprocessing", create the following tags:

Name	Data type	Input / output type
input 1	bool	Input
input 2	bool	Input
result	bool	Output

3. In the instruction window of the function "preprocessing", write the following program code:
   #result:=#input 1&#input 2;

4. Call the function "preprocessing" in a program cycle OB, for example, in OB1.
5. Wire the function "preprocessing" in the program cycle OB as follows:

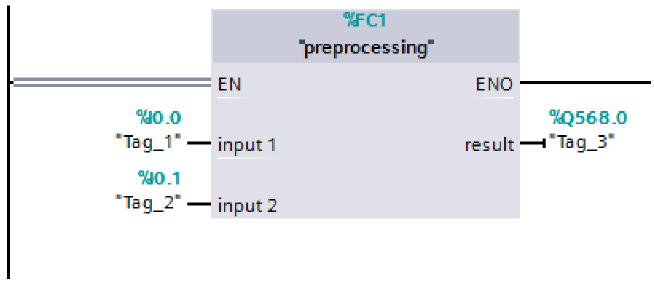


Figure 6-15 I-device sample program

Programming IO controller

To program the sample program for the IO controller, follow these steps:

1. Using the SCL programming language, create a new function with the name "further processing" in the project tree in "Program blocks" > "Add new block". Open the function.
2. In the interface of the function "further processing", create the following tags:

Name	Data type	Input / output type
result	bool	Input
output	bool	Output

3. In the instruction window of the function "further processing", write the following program code:
   #output:=#result;
4. Call the function "further processing" in a program cycle OB, for example, in OB1.
5. Wire the function "further processing" in the program cycle OB as follows:

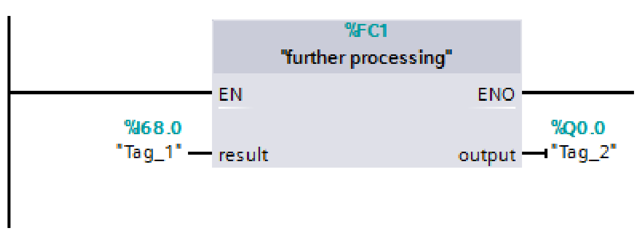


Figure 6-16 IO controller sample program

Result

The IO controller and the I-device can exchange data with each other through the input/output transfer areas.

6.2.7 Diagnostics and interrupt characteristics

Diagnostics and interrupt characteristics

S7 CPUs have numerous diagnostics and interrupt functions that can, for example, report errors or failures of lower-level IO systems. These diagnostics messages reduce down times and simplify localization and elimination of problems.

Diagnostics options in the higher-level IO controller and in the I-device

The following diagnostics functions are available to the higher-level IO controller and the I-device CPU:

- OB 83 (pull/plug)
- OB 86 (rack failure)
- OB 122 (I/O access error)

NOTE

The diagnostics 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.

Operating state changes and station failure / station return

In the following table, you can see which consequences an operating state change or the failure of an IO controller / I-device of the SIMATIC family has on the respective other(s):

Table 6-1 Operating state changes and station failure / station return

Initial status	Event	I-device response	Higher-level IO controller
I-device CPU is in RUN, higher-level IO controller is in RUN	The I-device CPU changes to STOP	-	During the updating of the process image with the instructions "UPDAT_PI" and "UPDAT_PO" an error report is returned at the parameter RET_VAL. With direct IO access to all transfer areas to the I-device: depending on the type of error handling, e.g., call of OB 122 (IO access error).
I-device CPU is in STOP higher-level IO controller is in RUN	The I-device CPU is starting up	Call of OB 100 (startup). Call of OB 83 (pull/plug) for input transfer areas to the higher-level IO controller. Until call of OB 83 in the case of direct access to the input transfer areas to the higher-level IO controller: depending on the type of error handling, e.g., call of OB 122 (IO access error).	Call of OB 83 (pull/plug) for all transfer areas to the I-device. Until call of OB 83 in the case of direct access to the transfer areas to the I-device: depending on the type of error handling, e.g., call of OB 122 (IO access error).

Initial status	Event	I-device response	Higher-level IO controller
I-device CPU is in RUN, higher-level IO controller is in RUN	The higher-level IO controller changes to STOP	During the updating of the process image with the instructions "UPDAT_PI" and "UPDAT_PO" an error report is returned at the parameter RET_VAL. With direct IO access to the input transfer areas to the higher-level IO controller: depending on the type of error handling, e.g., call of OB 122 (IO access error). Note: Output transfer areas can still be accessed.	-
Higher-level IO controller is in STOP, I-device CPU in RUN	The higher-level IO controller starts up	Call of OB 83 (pull/plug) for input transfer areas to the higher-level IO controller. Until call of OB 83 in the case of direct access to the input transfer areas to the higher-level IO controller: depending on the type of error handling, e.g., call of OB 122 (IO access error).	Call of OB 100 (startup).
I-device CPU is in RUN, higher-level IO controller is in RUN	Station failure I-device, for example, through bus interruption	If the I-device continues to run without a bus connection: Call of OB 86 (rack failure). During the updating of the process image with the instructions "UPDAT_PI" and "UPDAT_PO" an error report is returned at the parameter RET_VAL. With direct IO access to all transfer areas to the higher-level IO controller: depending on the type of error handling, e.g., call of OB 122 (IO access error).	Call of OB 86 (rack failure). During the updating of the process image with the instructions "UPDAT_PI" and "UPDAT_PO" an error report is returned at the parameter RET_VAL. With direct IO access to all transfer areas to the I-device: depending on the type of error handling, e.g., call of OB 122 (IO access error).
I-device CPU is in RUN, higher-level IO controller is in RUN, communication connection between IO controller and I-device is interrupted (bus interruption).	The bus connection between the IO controller and I-device has been reestablished and the I-device is again included in the user data traffic.	Call of OB 86 (rack failure). Call of OB 83 (pull/plug) for input transfer areas to the higher-level IO controller. Until call of OB 83 in the case of direct access to the input transfer areas to the higher-level IO controller: depending on the type of error handling, e.g., call of OB 122 (IO access error).	Call of OB 86 (rack failure). Until reporting of station re-integration by OB 86 with direct IO access to all transfer areas to the I-device: depending on the type of error handling, e.g., call of OB 122 (IO access error)

NOTE

Special characteristic during startup of the higher-level IO controller

In contrast to the station return message from IO devices in the IO controller, which are covered completely by the call of the OB 86, the station return message of a higher-level IO controller in the I-device is separated into 2 parts:

1. Call of the OB 86: The initial values for the outputs of the I-device are set. However, the input values are not yet valid. These values will first be valid with the opening of OB 86 in the higher-level IO controller.
2. Call of OB 83 for each input transfer area; with this call, the validity of an input transfer area is displayed. The starting up of the I device is first complete when the OB 83 has been called for the input transfer areas. This step can be delayed or not occur at all in the following situations:
   - Higher-level IO controller is in STOP: OB 83 is first called at the STOP-RUN transition of the higher-level IO controller.
   - The IRT communication has been disrupted (sync-master failure, topology error, ...). OB 83 is opened only after the IRT communication has taken place.

Reference

For more information on error handling in the case of direct I/O access, refer to "Error handling" in the STEP 7 online help.

6.2.8 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 keeping communication small.

The following communication paths should not overlap:

- Communication paths between the IO controller and the IO device of its IO system.
- Communication paths of the I-device CPU and the IO devices of your IO system.

I-device with one port

Connect an I-device with only one port to a PROFINET 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.

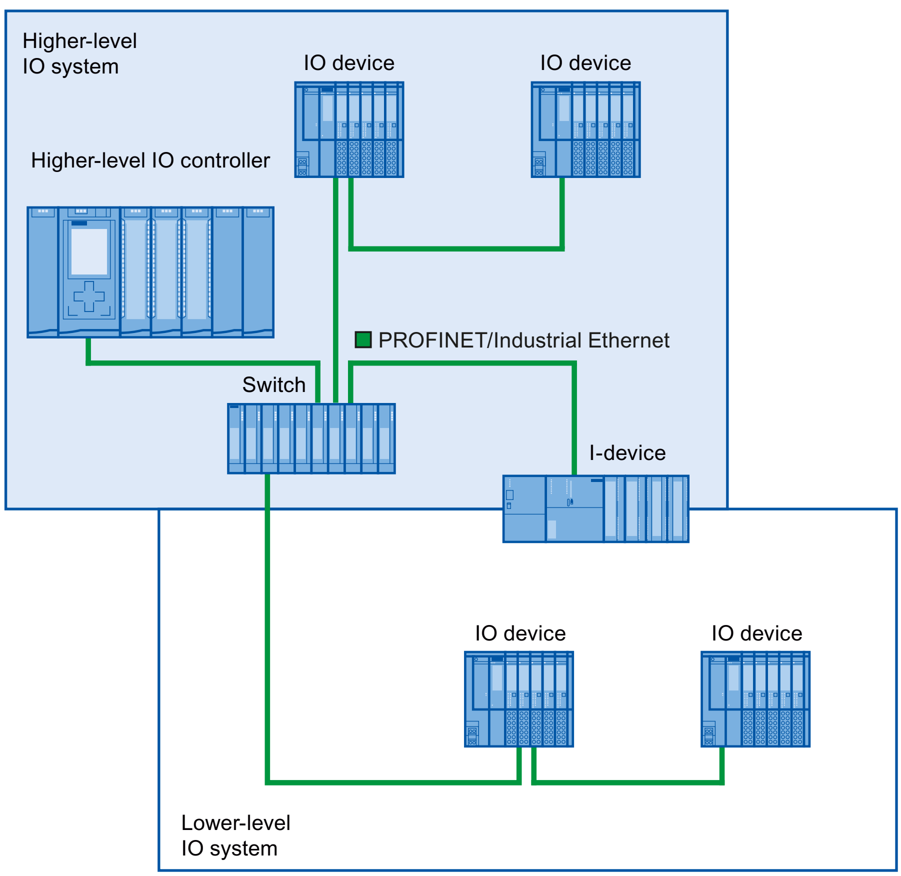


Figure 6-17 I-device with one port

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 PROFINET switch. Use the second port for the lower-level IO system as shown in the following figure.

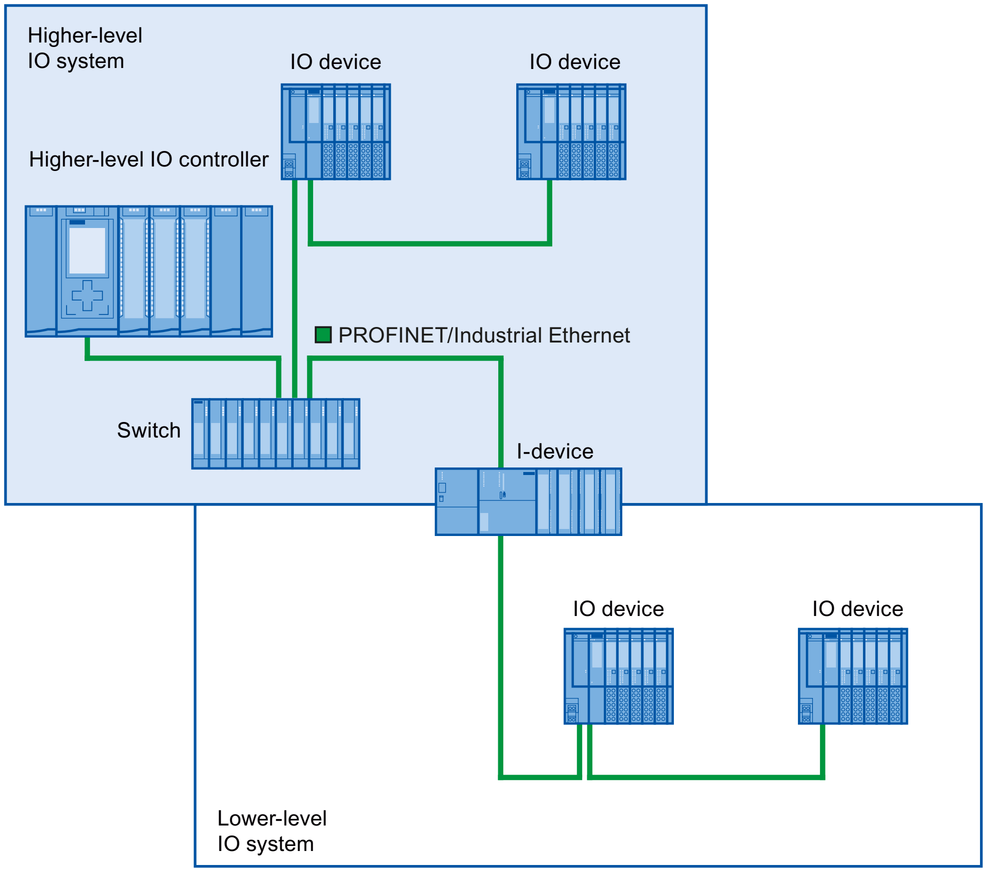


Figure 6-18 I-device with two ports

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 linear bus topology as shown in the following figure.

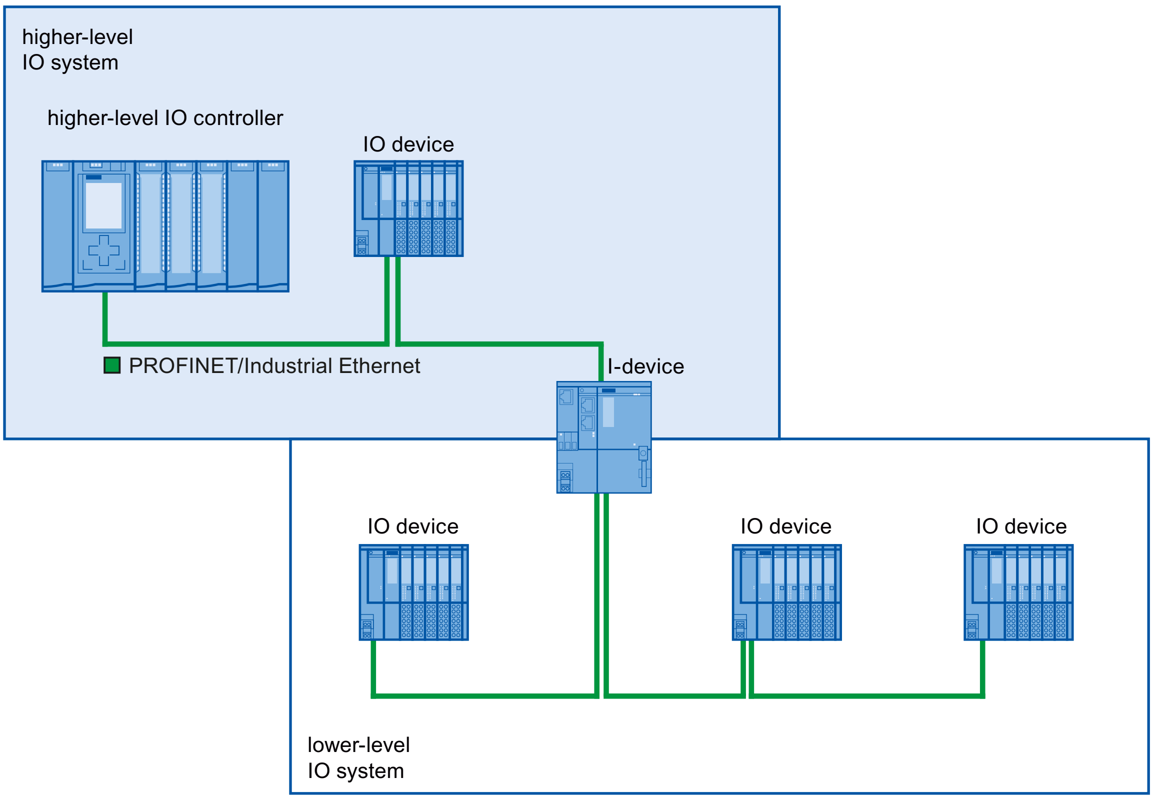


Figure 6-19 I-device with three or more ports

6.2.9 Boundary conditions when using I-devices

Note the following boundary conditions when using I-devices:

Bandwidth

The number of addresses of the configured transfer areas affects 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, this indicates a larger bandwidth requirement and can thus lead to longer update times.

Tip: Keep the address space of the transfer area as small as possible.

Rules for RT and IRT communication.

IO systems with I-devices are also suitable for setting up real-time applications with RT and IRT communication. The following rules must be followed for this:

- 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. You can use IRT communication in one of the two IO systems. Use IRT either in a higher-level or in a lower-level IO system.

6.2.10 Configuring PROFIenergy with I-devices

The requirement for program-controlled pauses for saving energy with PROFINET devices is that the PROFINET devices support the PROFIenergy protocol.

Only if a PROFINET device (I/O device) supports the PROFIenergy protocol does an I/O controller actually send PE commands to this I/O device, for example to start or stop pauses.

If an I/O device supports the PROFIenergy protocol, this property is saved in its PROFINET GSD file and is available for configuration in an engineering system.

For S7-1500 CPUs as intelligent I/O devices (I-devices), you have the option with STEP 7 V13 service pack 1 or later to set PROFIenergy support for each transfer area.

If you have selected the "Enable PROFIenergy communication" option for a transfer area and import the generated PROFINET GSD file into another project, you can handle an I-device as a PE entity there.

Requirements

- STEP 7 as of V13 service pack 1
- CPU supports PROFIenergy with I-devices, for example CPU 1215C DC/DC/DC as of firmware version 4.2
- You use the PROFINET IO interface as an I-device and have created transfer areas.
- The user program in the I-device handles PROFIenergy commands

  Background: You need to program PROFIenergy functions with I-devices in the user program using the "PE_I_DEV" instruction and corresponding auxiliary blocks; this is different compared with IO devices for which this functionality is made available by the firmware. You may therefore only activate the PROFIenergy support for transfer areas if the user program in the I-device is configured correspondingly as well.

Enabling PROFIenergy for transfer areas of I-devices

Proceed as follows to assign parameters for the support of PROFIenergy:

1. Select the PROFINET interface (X1) of the CPU.
2. Select the required transfer area in the area navigation, for example:
   Operating mode > I-device communication > Transfer_area_1.
3. Select the check box "Enable PROFIenergy communication".

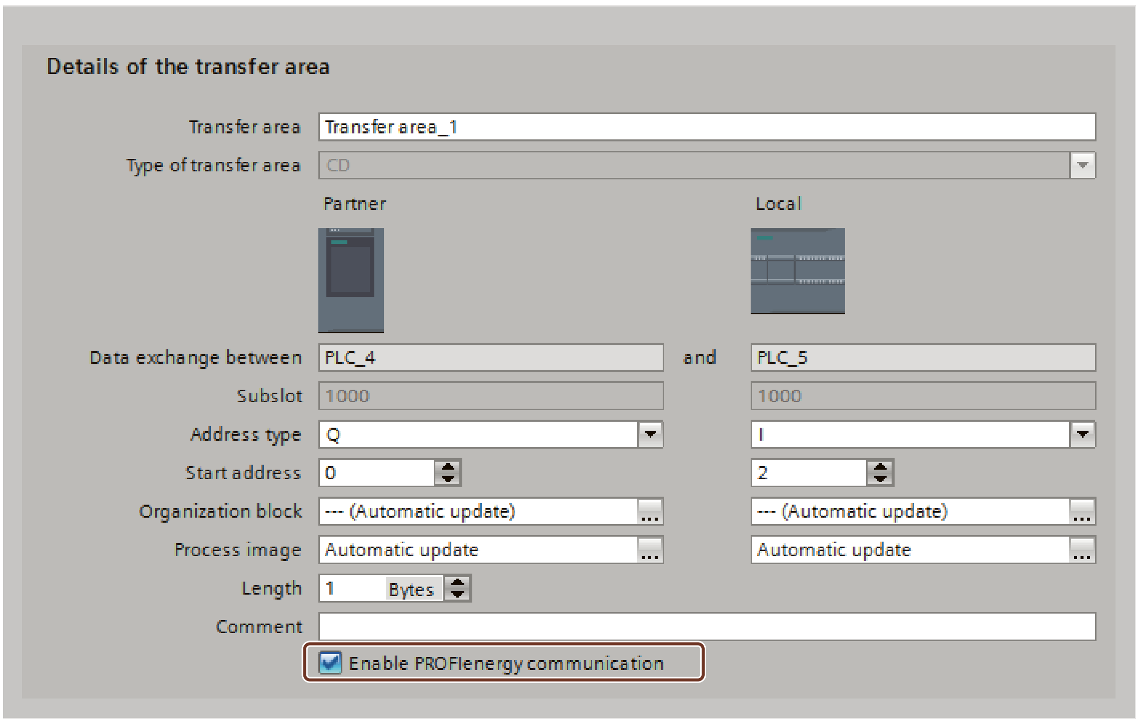


Figure 6-20 Configuring PROFIenergy with I-devices

Once the I-device is fully configured, generate the GSD file for the I-device and import this file in the project for the I/O controller. The GSD file generated contains an entry that specifies that the I-device supports the PROFIenergy profile.

To address the I-device, for example for the PE command "PE_START_END", use the hardware identifier of the "PROFIenergy supporting" transfer area in the I-device.

To address the IO controller for the PE command "PE_I_DEV", use the hardware identifier of the transfer area that is supplied with the data for PROFIenergy on the IO controller.

You can find more information on PROFIenergy in the sectoin Saving energy with PROFIenergy (Page 276).

6.2.11 Enabling/disabling I-device in the user program of the I-device CPU

When a PROFINET interface of a CPU was configured as an I-device in STEP 7, the I-device function is active after a status transition of the CPU from STOP > RUN. If no higher-level IO controller can be reached in this case, the I-device CPU signals an error using its ERROR LED.

As of firmware version 2.9 of an S7-1500/ET 200SP/ET 200pro CPU, you can locally disable or enable the I-device function in the user program of the I-device CPU. You use the "D_ACT_DP" instruction for this. After disabling the function, the I-device CPU no longer signals an error using its ERROR LED.

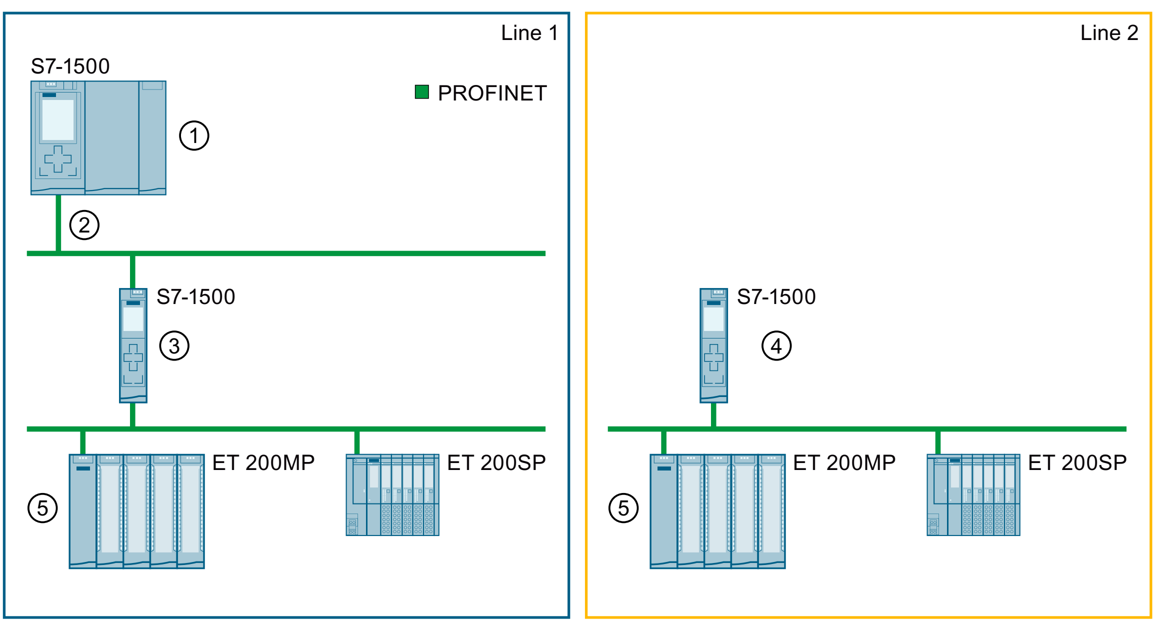
You can enable/disable the function with the user program at the CPU-internal PROFINET interfaces and for S7-1500 CM 1542-1 as of firmware version V3.0.

Application examples

From a machine OEM's point of view, there are numerous device options possible in **series production of machines**. However, each delivered machine includes only one combination of selected options. All possible machine options can be configured as I-devices by the manufacturer to create and maintain a common user program having all possible options.

A higher-level IO controller used for line control, for example, is not necessary in every machine. In this case, the user program can disable the I-device function in the I-device CPU with the instruction "D_ACT_DP", for example, during startup.

Two production lines are shown schematically in the figure below. The same user program runs in both CPUs with I-device configuration. The I-device in line 1 is controlled by a higher-level IO controller.



① Higher-level IO controller

② ... controls I-device CPU S7-1500 in the line

③ I-device function is enabled in the user program of this CPU

④ I-device function is disabled in the user program of this CPU

⑤ IO devices

Figure 6-21 Enable/disable the I-device function in the I-device CPU

Automation task control of vehicles in a ride

In a ride (ghost train), the vehicles are connected as I-devices to a higher-level controller via PROFINET.

Feature

The higher-level IO controller takes over the entire coordination and monitoring of the vehicles in the station. Among other things, the higher-level IO controller registers the following states:

- When the vehicles are available in the station
- When the vehicles exit the station

As soon as passengers are seated in the vehicle and the vehicle is closed, the higher-level IO controller issues the start signal.

Along the entire track, the vehicles are traveling independently and do not have a connection to the higher-level IO controller. Because this state does not represent an error, no errors should be signaled via the ERROR LEDs of the CPUs.

Solution

Depending on the position of the vehicle, inside or outside the station, the user program enables or disables the I-device function in the I-device CPU of the vehicle. No diagnostics were signaled via LED on the I-device CPU for enabling and disabling.

Requirements

You operate the CPU as an I-device on one or multiple IO controllers or you have configured the CPU as I-device ("IO device" check box is selected). In the following figure, the right CPU 1518-4 PN/DP is the I-device at the higher-level IO controller CPU 1518-4 PN/DP (PLC_2).

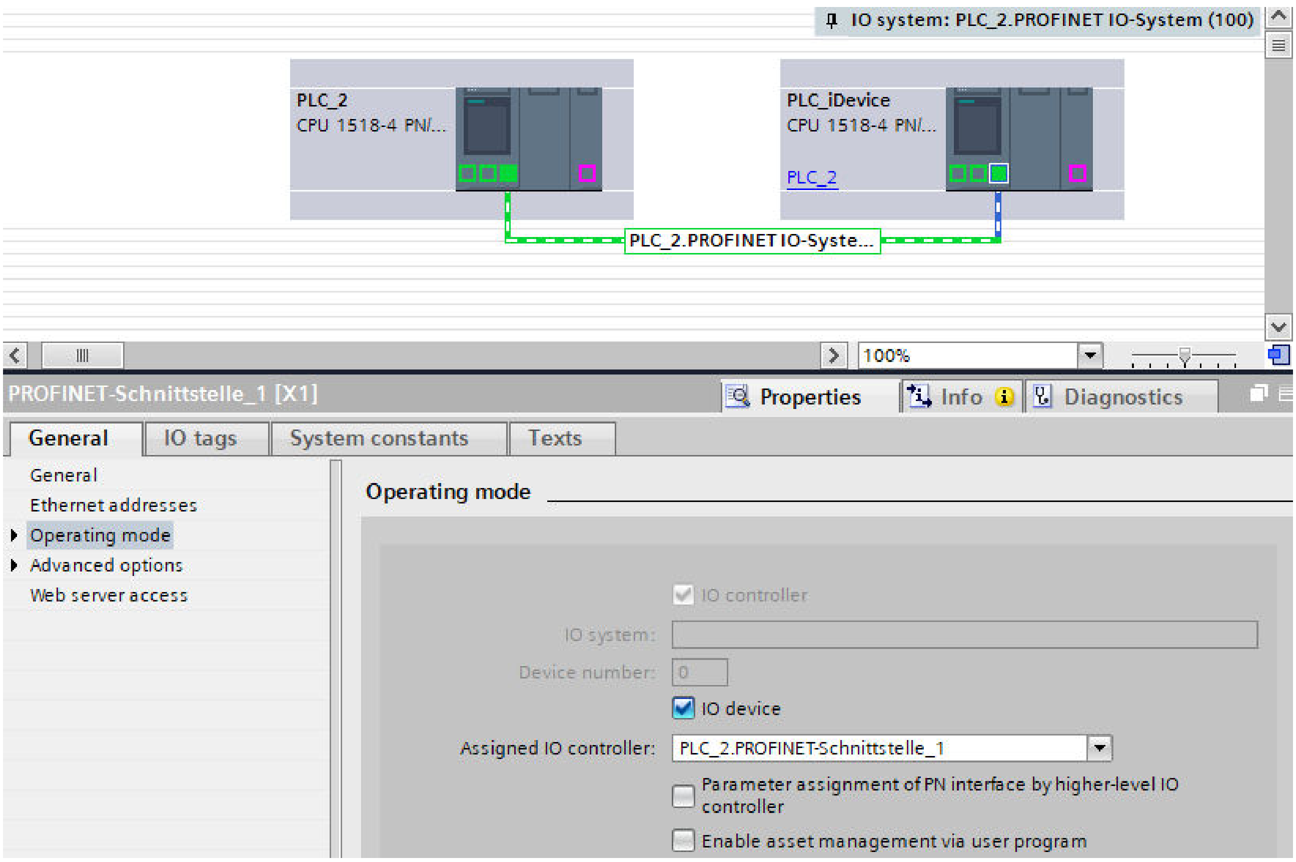


Figure 6-22 Example: Enable I-device

"D_ACT_DP" is an asynchronous instruction. Processing extends over multiple calls. You have started the job by calling "D_ACT_DP" with REQ = 1 in the startup OB of the I-device CPU.

You can find additional information on the "D_ACT_DP" instruction in the STEP 7 online help.

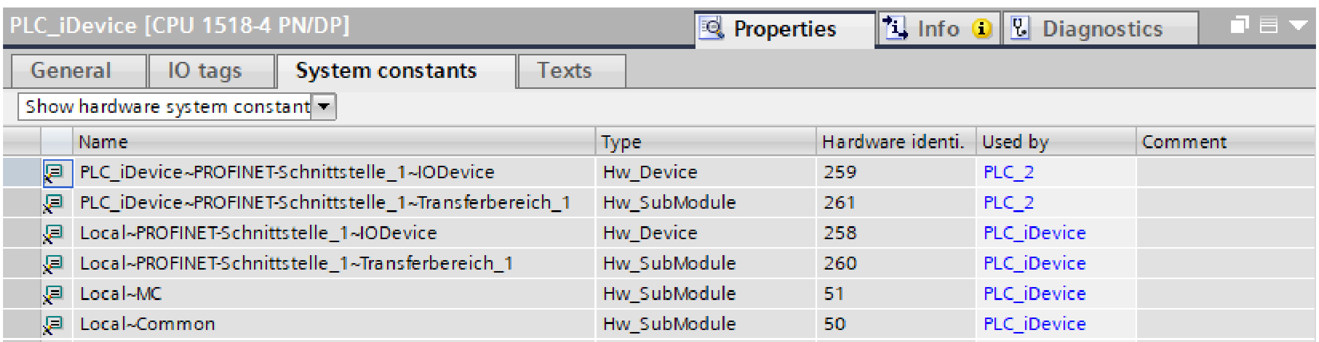
Disabling the I-device function at a PROFINET interface

NOTE

Only disable the I-device function in the I-device CPU when the I-device is not operated on a higher-level IO controller.

You disable the I-device function with the instruction "D_ACT_DP":

- Via the symbolic name or the hardware identification of the I-device of the respective PROFINET interface as LADDR parameter, in the example, HW identification 260
- Via the MODE parameter = 2



Name	Type	Hardware identi.	Used by	Comment
PLC_iDevice~PROFINET-Schnittstelle_1~IODevice	Hw_Device	259	PLC_2	
PLC_iDevice~PROFINET-Schnittstelle_1~Transferbereich_1	Hw_SubModule	261	PLC_2	
Local~PROFINET-Schnittstelle_1~IODevice	Hw_Device	258	PLC_iDevice	
Local~PROFINET-Schnittstelle_1~Transferbereich_1	Hw_SubModule	260	PLC_iDevice	
Local~MC	Hw_SubModule	51	PLC_iDevice	
Local~Common	Hw_SubModule	50	PLC_iDevice	

Figure 6-23 Example: I-device system constants

Result: The goal has been accomplished: The CPU with disabled I-device function is no longer available for the data exchange with a higher-level IO controller. The higher-level IO controller can be reached again when the I-device function was enabled in the user program of the I-device CPU.

When an I-device that has been active on an IO controller disables itself, all application relations (ARs) that were set up between the I-device and the IO controller in the network are terminated. An IO device failure is signaled at the IO controller end. In the diagnostic buffer of the I-device CPU, "IO device user disable" is displayed with information on the associated PROFINET interface or the name of the IO controller.

Enabling the I-device function at a PROFINET interface

You use the instruction "D_ACT_DP" to enable an inactive I-device function:

- Via the symbolic name or the hardware identification of the I-device of the respective PROFINET interface as LADDR parameter, in the example, HW identification 260
- Via the MODE parameter = 1

Result: After enabling, the I-device can exchange data with a higher-level IO controller.

If at least one IO controller is missing or the connection does not exist, "Hardware component removed or missing" is entered in the diagnostic buffer of the I-device CPU.

NOTE

Temporary access errors

Enabling the I-device function in the user program can result in temporary access errors "I/O data failure in hardware component", for example, because access to the I/O data of the I-device is already possible even though the data has not been declared as valid by the IO controller yet.

These messages are incoming and outgoing error events that can be neglected.

The behavior of IO controller and I-device in case of a station failure and station return still applies as described in the section Diagnostics and alarm response (Page 134).

Calling up the status information of an I-device with the instruction "D_ACT_DP"

You are requesting the status information of an I-device:

- Via the hardware identification of the I-device of the respective PROFINET interface as LADDR parameter, in the example, HW identification 260
- Via the MODE parameter = 0

The status is returned in the RET_VAL parameter:

- RET_VAL = 0x0000: "D_ACT_DP" was completed without errors (disabling an inactive I-device/enabling an active I-device)
- RET_VAL = 0x0001: I-device is enabled
- RET_VAL = 0x0002: I-device is disabled

6.3 Shared device

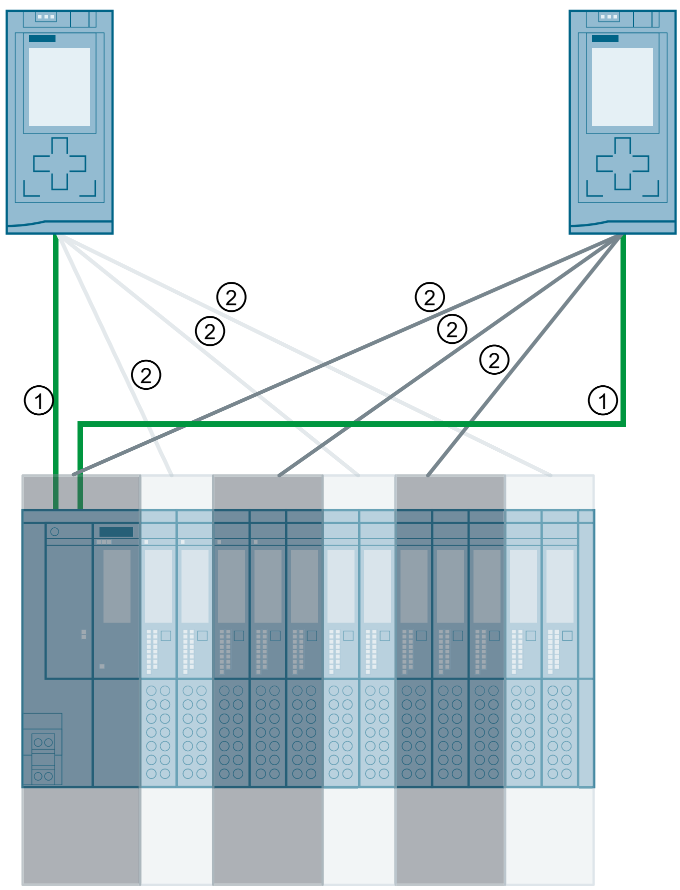
6.3.1 Useful information on shared devices

Shared device functionality

Numerous IO controllers are often used in larger or widely distributed systems.

Without the "shared device" function, each IO module of an IO device is assigned to the same IO controller. If sensors that are physically close to each other must provide data to different IO controllers, several IO devices are required.

The "Shared Device" function allows the modules or submodules of an IO device to be divided up among different IO controllers. Thus allowing flexible automation concepts. You can, for example, combine IO modules that are physically close to one other in one IO device. Drives such as SINAMICS S120 also support the shared device functionality.



① PROFINET

② Logical assignment

Figure 6-24 Example: Shared device with two IO controllers

Principle

Access to modules and submodules of the shared device is then divided up among the individual IO controllers. Each module or submodule of the shared device is assigned exclusively to one IO controller.

Configuration possibilities of shared devices

If a maximum of two IO controllers access a shared device, it is preferable to configure the shared device in a common project together with the IO controllers that access it. As a result, all components of a shared device configuration are known to the project. Therefore, configuration in a common project offers the following advantages compared to configuration in multiple projects:

- Reduction of possible sources of error: STEP 7 performs a full consistency check within a project.
- Less configuration effort: A common STEP 7 project contains all devices.
- Improved diagnostics: Complete diagnostics in one project.

To configure access to a shared device by more than two IO controllers, you need to use the shared device in multiple projects. The shared device must be configured in every project that contains IO controllers accessing it.

Using a shared device and its IO controllers in a common project:

- Since STEP 7 V18: You can configure a shared device and two CPUs accessing it (IO controllers) in a single project.
- From STEP 7 V19: You can also configure a shared I-device and two CPUs accessing it (IO controllers) in a single project.
  - Requirement: Use an S7-1500 CPU with FW version V3.1 or higher as the shared I-device

Using a shared device in multiple projects:

- Since STEP 7 V12 SP1, S7-1500 CPUs and SIMATIC distributed I/O devices support the "Shared Device" function. The shared device must be present in every project, although it is present only once in the installation. A separate project is required for each CPU accessing the shared device. To find out which components support the "Shared Device" function, see the following FAQ (https://support.industry.siemens.com/cs/us/en/view/102325771).

Setting of the real-time properties

If in addition to the CPUs in the project, other CPUs outside the project have access to the shared device, you must enter the number of project-external IO controllers for the PROFINET interface of the shared device. Only then does STEP 7 correctly calculate the communication load and thus the resulting update times.

The maximum possible number of IO controllers for the shared device depends on the device. You can find this number as follows:

- In the GSD file of the shared device.
- In the equipment manuals of the IO devices or CPUs that you intend to use as the shared device.

With a CPU used as an IO controller, you can set a very fast send clock. The send clock can be shorter than the shortest send clock supported by the shared device. In this case, the IO controller operates the shared device with a send clock it supports (send clock adaptation).

Example:

A CPU supports send clocks starting from 0.25 ms. One configured IO device also supports send clocks starting from 0.25 ms, while another IO device supports send clocks starting from 1 ms. In this case, you have the option of setting the short send clock of 0.25 ms for the CPU. The CPU operates the "slow" IO device with the send clock of 1 ms, for example.

Response in the event of fault

You can find out how PROFINET IO controllers behave in the event of an error when accessing the data of a shared device in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/109572804).

6.3.2 Shared device and assigned IO controllers in the common project

Requirements

- The shared device and the two IO controllers accessing it must be in the same subnet.
- The shared device must be assigned to each IO controller.
- There must be at least one module or submodule assigned to each connected IO controller.

If these conditions are not met, the configuration as a shared device is inconsistent. When an inconsistent configuration is compiled in STEP 7, a corresponding error message is displayed.

Restrictions

- In STEP 7 V18, only IO devices that have been installed in the hardware catalog via GSD files are permitted as shared devices in the common project.
  Starting from STEP 7 V19, you can insert and configure IO devices as shared devices in the common project directly from the STEP 7 hardware catalog without prior installation of GSD. You can find an overview of which IO devices support this usage in the following FAQ (https://support.industry.siemens.com/cs/us/en/view/102325771).
- A maximum of two IO controllers (CPUs) may access one shared device within a project.
- The shared device does not support system redundancy.
- You can find specifics on handling fail-safe IO devices in the SIMATIC Safety - Configuring and Programming V19 (https://support.industry.siemens.com/cs/us/en/view/54110126) Programming and Operating Manual.
- "TIA Portal Openness" does not support shared device.
- Loading both CPUs as a new station into the same project is not supported. Load the two CPUs that share the shared device into two different projects.

Access to modules and submodules

Each IO controller only has access to the modules and submodules assigned to it. This means:

- Data is exchanged only with the assigned modules or submodules
- Alarms and diagnostics are only received from assigned modules or submodules
- Only parameters of assigned modules or submodules are configured

Rules for the configuration

The following rules apply to the configuration of a shared device in a common project and are checked automatically by STEP 7 when the configuration is compiled:

- Only one IO controller at a time may have full access to one submodule.
- I/O addresses of a module or submodule can only be edited in the address area of the IO controller to which the shared device is assigned.
- The send clock must be identical for all IO controllers that have access to the shared device.
- The following functions are only possible with the IO controller to which the interface module of the shared device is assigned:
  - Isochronous mode (IRT)
  - Media redundancy
  - Prioritized startup
  - Parameter assignment of the port properties
- If the "Use name as extension for device names" option is selected for an IO system, this option must also be selected for the IO system of the second CPU.
- If you change the name of the shared device, then you must subsequently load all the CPUs that share this IO device.
- The maximum total of the communication relationships (ARs) for the shared device must not be exceeded. The maximum total of the communication relationships may be found, for example, in the Equipment Manual of the IO device you wish to use as the shared device.

  Example:
  An ET 200SP interface module supports up to four communication relations as a shared device. The shared device has already been assigned to two IO controllers in the project. A maximum of two more IO controllers that have been configured in other projects may access modules or submodules of this shared device.

6.3.3 Configuring a shared device in a common project

Below, you will find a description of how to configure a distributed I/O system as a shared device with STEP 7 version V18 or higher.

If your project contains a shared device that is accessed by more than two IO controllers, see section "Configuring a shared device for different projects (Page 157)". These IO controllers are configured either in another TIA Portal project or with a different engineering tool.

A "distributed" configuration with different engineering tools for different IO controller families is generally possible. However, the description of the procedure is based exclusively on STEP 7 V18 or higher.

The following description is limited to two IO controllers of the S7-1500 family that share a shared device in a common project.

Since STEP 7 V18, only one project is needed for a shared device configuration. The project contains the shared device and a maximum of two IO controllers that access it.

Requirements

- STEP 7 (TIA Portal) V18 or higher
- The IO controllers support the shared device function, e.g. CPU 1513-1 PN FW version V3.0 or higher.
- The IO device supports the shared device function, e.g. IM 155-5 PN ST interface module.
- STEP 7 V18: GSD files for the IO devices are installed and being used.
- STEP 7 as of V19: You can use IO devices from the hardware catalog of STEP 7 without prior installation of their GSD files. The desired IO device must support the shared device function when directly used from the hardware catalog.

Procedure - Creating project

To create a common project with a maximum of two IO controllers and a shared device, follow these steps:

1. Start STEP 7.
2. Create a new project, for example with the name "Shared Device".
3. In the network view, insert an IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
4. Assign a name, for example "PLC_1".
5. Insert another IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
6. Assign a name, for example "PLC_2".
7. Connect the PROFINET interface X1 of "PLC_1" and "PLC_2" to each another. The PROFINET interfaces of the CPUs are thus in the same subnet.
8. Insert an IO device (e.g. IM 155-6 PN ST) from the hardware catalog as follows:
   – STEP 7 V19 or higher: Use the default entries, for example, under "Distributed I/O > ET 200SP > interface module > PROFINET".
   – STEP 7 V18: You can find the IO devices installed from the GSD files under "Other field devices > PROFINET IO > IO > SIEMENS AG".
9. Double-click the inserted IO device. The device view opens.
10. Insert all required modules and submodules from the hardware catalog into the device overview table. If the project-internal shared device was installed and inserted as a GSD file, you can also use the modules and submodules installed from the GSD files.

11. Assign parameters for the individual IO modules.
12. Switch to the network view.
13. Assign the IO device in succession to the IO controllers "PLC_1" and "PLC_2".
14. Save the project.

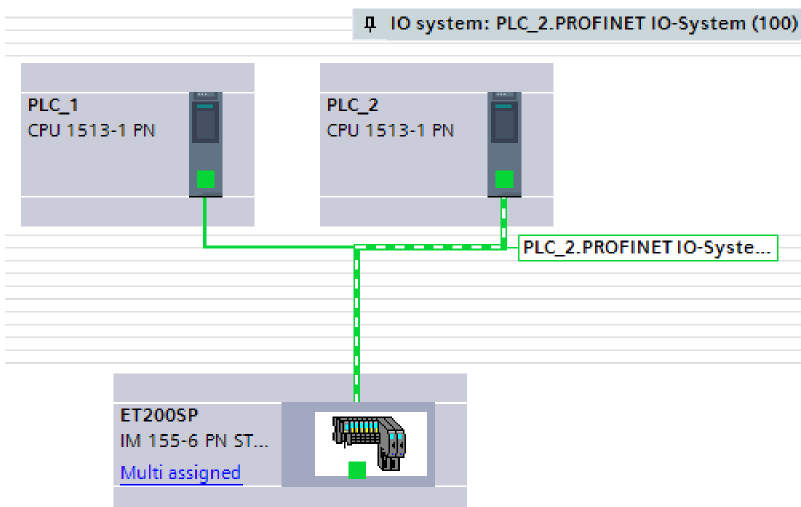


Figure 6-25 Shared device assigned in the common project

Procedure - Configuring the assignment of the modules or submodules

After the first assignment of the shared device to an IO controller, all modules or submodules are assigned to this IO controller.

Newly added modules or submodules are assigned to the IO controller to which the interface module is also assigned. If a shared device is disconnected from an IO controller, the assigned modules or submodules obtain the "unknown" access status.

To change the assignment, proceed as follows:

1. In the network view or the device view of the project, select the interface module of the shared device.
2. In the Inspector window, select the "Shared Device" tab under "Properties > General > Module parameters".
   A table shows which IO controller has access to the respective module or submodule. The default setting for all modules and submodules is the IO controller to which the shared device was first assigned.

3. Keep the "PLC_1" setting for all modules and submodules that are to remain in the address area.
   For all modules or submodules that you want to assign to the address area of the other IO controller, select the setting "PLC_2".
   Special consideration: With the "Outside of project" option, you can select IO controllers that are connected to the PROFINET IO system but are configured in another TIA project or engineering tool.

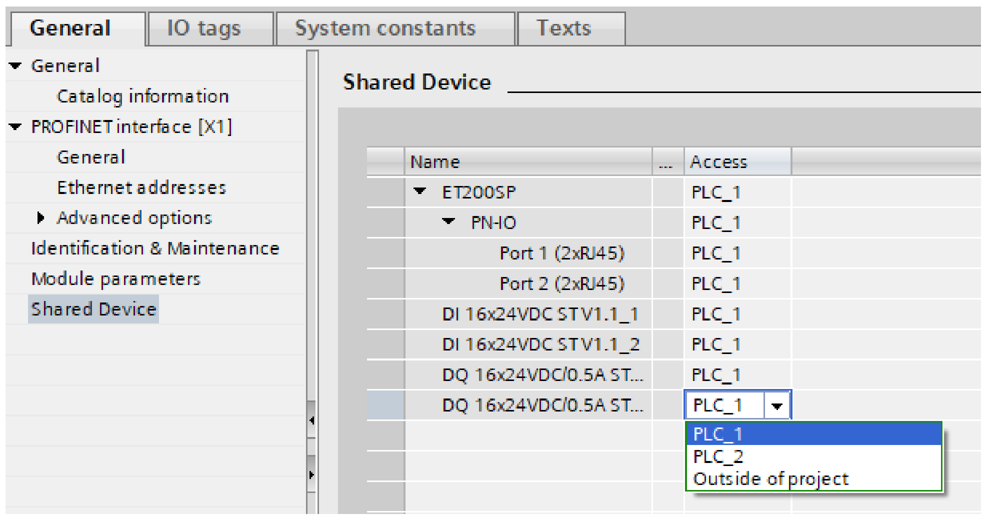


Figure 6-26 Configuring a project-internal shared device

Procedure - Adjusting the real-time settings

Adjusting and checking the settings listed below ensures the following properties:

- All IO controllers and shared devices are operated with the appropriate send clock.
- Update times are calculated correctly with respect to the communication load.

To adjust and check these settings, proceed as follows:

1. Select the interface module of the shared device in the network view.
2. In the Inspector window, navigate to the "PROFINET interface > Advanced options > Real time settings > IO cycle" area.
3. If IO controllers from other projects also access this shared device: Set the number of IO controllers outside the project in the "Shared Device" area. The maximum number depends on the IO device. You can find the maximum number as follows:
   – In the GSD file of the shared device.
   – In the Equipment Manual of the IO device you wish to use as the shared device.
4. For each IO controller that has access to modules or submodules of the shared device, you must adapt the real-time settings on a CPU-specific basis.

5. You must set the same send clock for each IO controller that has access to modules or submodules of the shared device:
   - If you configure an additional IO controller outside the project with STEP 7 (TIA Portal):
     Open the corresponding project.
     Select the PROFINET interface of the IO controller.
     Select the "Advanced options > Real time settings > IO communication" area in the Inspector window and set the same send clock.
   - If you configure an additional IO controller outside the project with a different engineering tool:
     Select the PROFINET interface of the shared device in STEP 7 (TIA Portal) and read out the send clock on the shared device ("Advanced options > Real time settings" area)
     Enter the read send clock in the engineering tool.
   - Special consideration: If you configure **all** IO controllers that have access to the shared device in a common STEP 7 project (STEP 7 V5.5 or higher), shorter send clocks can be set at the IO controller than are supported by the shared device (send clock adaptation).

Compiling and loading

When you select the shared device and click on "Compile" in the toolbar, the configuration of both IO controllers is checked for data consistency.

You must load the configurations for the various IO controllers into the IO controllers one after the other.

NOTE

Synchronizing router settings with the IO controller

The "Synchronize router settings with IO controller" option is selected by default for IO devices. If the connected IO controllers use the same router or no router, leave this setting as it is.

If the connected IO controllers use different routers, deselect the "Synchronize router settings with IO controller" option. In this case, manually configure the corresponding router address on the shared device.

6.3.4 Shared device and assigned IO controllers in different projects

Requirements

- STEP 7 V12 Service Pack 1 or higher.
- IO device supports the shared device function, e.g. IM 155-5 PN ST interface module.
- IO controller supports the shared device function, e.g. CPU 1516-3 PN/DP FW version V1.1 or higher or CPU 1215 DC/DC/DC FW version V4.1 or higher.

Configuring the access

The IO device must be present in more than one project in order for the modules or submodules of an IO device to be assigned to different IO controllers. A separate project is required for each IO controller.

Use the "Shared Device" parameter of the interface module to define the modules or submodules to which the IO controller has access:

- If the local IO controller has access to the configured module, select the name of the IO controller from the list.
- If the IO controller from a different project and not the local IO controller is to have access to the configured module, select the entry "Outside of project".

The configuration is consistent with regard to access when each module or submodule has been assigned to an IO controller in exactly one project.

Module or submodule is assigned to another IO controller

The following section is about the consequences of the "Outside of project" setting of the "Shared Device" parameter from the point of view of the local IO controller.

In this case, the local IO controller does not have access to the module configured in this way. Specifically, this means:

- No parameter assignment of the module or submodule
- No data exchange with the module or submodule
- No reception of alarms or diagnostics, which means no display of the diagnostics status in the online view

Rules for the configuration

- IO controllers that use the shared device are created in different projects. In each project, care must be taken that the shared device is configured identically in each station.
- If you have selected the "Use router" option for the IO controllers involved, you must have set the same router address in all these IO controllers. You as the user must ensure consistency.
- The shared device must have the same IP parameters and the same device name in each project.
- The S7 subnet ID of the subnet to which the shared device is connected must be identical in all projects.

- Only one IO controller at a time may have full access to one submodule. Inconsistencies in the configuration result in a failure of the shared device.
- I/O addresses of a module or submodule can only be edited if a module or submodule has been assigned to the IO controller in the same project.
- The send clock must be identical for all IO controllers that have access to the shared device.
- The following functions are only possible with the IO controller to which the interface module of the shared device is assigned:
  - Isochronous mode (IRT)
  - Media redundancy
  - Prioritized startup
  - Parameter assignment of the port properties
- If you change the name of the shared device, then you must subsequently load all the CPUs that share this IO device.
- The maximum total of the communication relationships (ARs) for the shared device must not be exceeded. The maximum total of the communication relationships may be found, for example, in the Equipment Manual of the IO device you wish to use as the shared device.
  Example:
  An ET 200SP interface module supports up to four communication relations as a shared device. The shared device has already been assigned to two IO controllers in the project. A maximum of two more IO controllers that have been configured in other projects may access modules or submodules of this shared device.

Constraints

The following constraints result because a shared device configuration is distributed across different projects:

- In the address overview of each IO controller that has access to a shared device, the addresses of modules or submodules that are not assigned to this IO controller are missing.
- The modules or submodules that are not assigned are not taken into consideration in the configuration limit calculation for the shared device during the consistency check. For this reason, you must yourself check that the maximum number of submodules or the maximum amount of cyclic IO data for the shared device is not exceeded.
  For information on the maximum configuration limits, refer to the documentation of the devices you are using.
- Configuration errors such as assignment of a module or submodule to more than one IO controller are not detected by STEP 7.
- CPUs that are loaded with a shared device configuration do not have any information on whether the IO device is a shared device. Modules or submodules that have been assigned to other IO controllers, and therefore other CPUs, are missing in the loaded configuration. These modules or submodules are therefore displayed neither in the CPU web server nor in the CPU display.

See also

Response in the event of fault
(https://support.industry.siemens.com/cs/ww/en/view/109572804)

6.3.5 Configuring a shared device for different projects

Below, you will find a description of how to configure a distributed I/O system as a shared device with STEP 7 version V12, Service Pack 1 or higher.

A "distributed" configuration with different engineering tools for different IO controller families is in principle possible. However, this description of the procedure is based exclusively on STEP 7 V12, Service Pack 1 or higher. The following description is limited to two IO controllers of the S7-1500 family that share a shared device in different projects.

Two projects are created (Shared-Device-1 and Shared-Device-2), each with one IO controller (PLC_1 and PLC_2). You must create the shared device in both projects, even though it is physically one and the same IO device.

Requirements

- STEP 7 (TIA Portal) as of V12 Service Pack 1
- The IO controllers support the shared device function, e.g. CPU 1513-1 PN, FW version V1.1 or higher.
- The IO device supports the shared device function, e.g. IM 155-5 PN ST interface module, FW version V2.0 or higher.

Procedure - Creating project 1

To create the first project with a shared device, follow these steps:

1. Start STEP 7.
2. Create a new project, for example with the name "Shared-Device-1".
3. In the network view, insert an IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
4. Assign a name, for example "PLC_1".
5. Insert an IO device (e.g. IM 155-6 PN ST) with the "Shared device" function from the hardware catalog.
6. Assign the IO device to the IO controller "PLC_1".
7. Double-click the IO device and insert all required modules and submodules from the hardware catalog into the device overview table.
8. Assign the module parameters.
9. Save the project.

Procedure - Creating project 2

To create the second project with a shared device, follow these steps:

1. Start STEP 7 once again.
   A new instance of STEP 7 opens.
2. In the new instance, create a new project, for example with the name "Shared-Device-2".
3. In the network view, insert an IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
4. Assign a name, for example "PLC_2".
5. Copy the IO device from the "Shared-Device-1" project and insert it into the network view of the "Shared-Device-2" project.
6. Assign the IO device to the IO controller "PLC_2".
7. Save the project.

Both projects now have an identically structured IO device that must be configured in the next step for the different IO controller accesses.

Procedure - Configuring the assignment of the modules or submodules

The modules and submodules you insert in the shared device are automatically assigned to the local CPU. To change the assignment, follow these steps:

1. Select the interface module in the network view or device view of the "Shared-Device-1" project.
2. In the Inspector window, select the "Shared Device" tab under "Properties > General > Module parameters".
   A table shows which CPU has access to the respective module or submodule for all configured modules. The default setting is that the local CPU has access to all modules and submodules.
3. Keep the "PLC_1" setting for all modules and submodules that are to remain in the address area of the local CPU.
   Select the "Outside of project" setting for all modules or submodules that are to be located in the address area of the CPU from the "Shared-Device-2" project ("PLC_2"). This means that an IO controller outside of the project is to have access to the module or submodule.

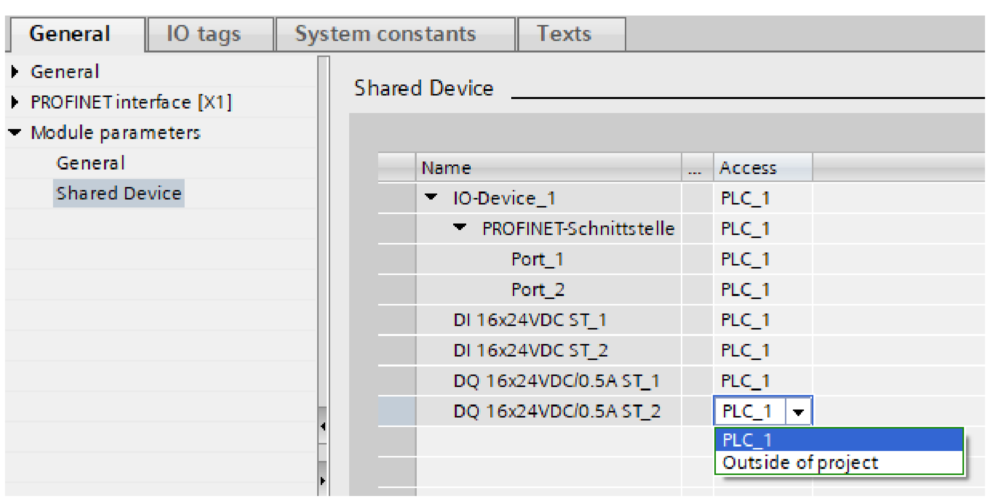


Figure 6-27 Configuring shared device

4. Select the interface module in the network view or device view of the "Shared-Device-2" project.
5. In the Inspector window, select the "Shared Device" tab under "Properties > General > Module parameters".
   A table shows which CPU has access to the respective module or submodule for all configured modules.
6. Select the "Outside of project" setting for all modules or submodules that are to be located in the address area of the CPU from the "Shared-Device-1" project ("PLC_1").
7. Finally, check whether the settings for access are "complementary" for each module or submodule in both projects. This means that if the local CPU has access in one project, the "Outside of project" option must be set in the other project and vice versa.
   Special consideration: The option "Outside of project" for the PROFINET interface and therefore for the ports makes the associated parameters read-only and not changeable. Parameters of the PROFINET interface and port parameters can only be edited in the project in which the PROFINET interface is assigned to the local CPU. The ports can be interconnected in both projects regardless of this.
8. Check whether the same IP address parameters and device name are set for the shared device in all projects.
   Check whether the same S7 subnet ID is set in all projects for the subnet to which the shared device is connected (subnet properties, "General" area in the Inspector window).

NOTE

If you make changes to the shared device: Make the same changes in each project on the shared device. Make sure that only one IO controller at a time has access to a module or submodule!

Procedure - Adjusting the real time settings

Adjusting and checking the settings listed below ensures the following properties:

- All IO controllers and shared devices are operated with the appropriate send clock.
- Update times are calculated correctly based on the communication load.

To adjust and check these settings, proceed as follows:

1. Select the project whose IO controllers have access to the PROFINET interface and the ports of the shared device.
2. Select the interface module of the shared device in the network view.
3. In the Inspector window, navigate to the "PROFINET interface > Advanced options > Real time settings > IO cycle" area.
4. In the "Shared Device" area, set the number of project-external IO controllers. The maximum number depends on the IO device (specification in GSD file).
5. For each IO controller that has access to modules or submodules of the project-internal shared device, you must adapt the real-time settings on a CPU-specific basis.

6. You must set the same send clock for each IO controller that has access to modules or submodules of the shared device:
   – If the IO controller is configured with STEP 7 (TIA Portal):
     Open the corresponding project.
     Select the PROFINET interface of the IO controller.
     Select the "Advanced options > Real time settings > IO communication" area in the Inspector window and set the same send clock.
   – If the IO controller is configured with a different engineering tool:
     Select the PROFINET interface of the shared device in STEP 7 (TIA Portal) and read out the send clock on the shared device ("Advanced options > Real time settings" area)
     Enter the read send clock in the engineering tool.
   – Special consideration: If **all** IO controllers that have access to the shared device are configured in STEP 7 (TIA Portal or STEP 7 V5.5 or higher), you can set shorter send clocks at the IO controller than are supported by the shared device (send clock adaptation).

Compiling and loading

You must compile the configurations for the different IO controllers and load them into the CPUs one after the other.

Due to the distributed configuration with separate projects, STEP 7 does not output consistency errors in the case of incorrect access parameter assignment. Example for incorrect access parameter assignment:

- Multiple IO controllers have access to the same module
- IP address parameters or send clocks are not identical

These errors do not show up until operation and are output as configuration errors, for example.

NOTE

After a configuration has been loaded in the IO controller, non-assigned modules or submodules retain their current parameterization state to ensure the independence from the parameterizations of other IO controllers.

See also

Module-internal shared input/shared output (MSI/MSO) (Page 174)

6.3.6 Shared I-device and assigned IO controllers in the common project

Requirements

- The I-device as shared device (shared I-device) and the two IO controllers accessing it must be in the same subnet.
- The shared I-device must be assigned to each IO controller.

If these conditions are not met, the configuration as a shared I-device is inconsistent. When an inconsistent configuration is compiled in STEP 7, a corresponding error message is displayed.

Restrictions

- A maximum of two IO controllers (CPUs) may access one shared I-device within a project.
- Devices from the SIMOTION product family do not support the shared I-device function.
- The shared I-device does not support system redundancy.
- "TIA Portal Openness" does not support shared I-device.
- The simultaneous loading of CPUs and shared I-device into a new project is not supported. Load each station individually into the new project.

Access to modules and submodules

Each IO controller only has access to the modules and submodules of its IO devices that are assigned to it. This means:

- Data is exchanged only with the assigned modules or submodules
- Alarms and diagnostics are only received from assigned modules or submodules
- Only parameters of assigned modules or submodules are configured

Rules for the configuration

The following rules apply to the configuration of a shared I-device in a common project and are checked automatically by STEP 7 when the configuration is compiled:

- A maximum of two IO controllers share one shared I-device.
- There must be at least one transfer area assigned to each connected IO controller.
- Only one IO controller at a time may have full access to one submodule.
- The send clock must be identical for all IO controllers that have access to the shared I-device.
- Isochronous mode (IRT) is only possible with the IO controller to which the interface module of the shared I-device is assigned.
- If the "Use name as extension for device names" option is selected for an IO system, this option must also be selected for the IO system of the second CPU.
- If you change the name of the shared I-device, then you must subsequently load all the CPUs that share this shared I-device.

6.3.7 Configuring a shared I-device in the common project

Below, you will find a description of how to configure a S7-1500 CPU as a shared I-device with STEP 7 version V19 or higher.

If your project contains a shared I-device that is accessed by more than two IO controllers, see section "Configuring a shared I-device in different projects (Page 166)". These IO controllers are configured either in another TIA Portal project or with a different engineering tool.

A "distributed" configuration with different engineering tools for different IO controller families is generally possible. However, the description of the procedure is based exclusively on STEP 7 from V19.

The following description is limited to two IO controllers of the S7-1500 family that share a shared I-device in a common project.

As of STEP 7 V19, only one project is required for a configuration with a shared I-device. The project contains the shared I-device and a maximum of two IO controllers that access it.

Requirements

- STEP 7 (TIA Portal) V19 or higher
- The IO controllers that access the shared I-device support the shared device function, e.g. CPU 1513-1 PN FW version V3.1 or higher.
- The IO controller that is used as a shared I-device supports the shared device function, e.g. CPU 1513-1 PN FW version V3.1 or higher.

Procedure - Creating project

To create a common project with a maximum of two IO controllers and a shared I-device, follow these steps:

1. Start STEP 7.
2. Create a new project, for example with the name "Shared I-device".
3. In the network view, insert an IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
4. Assign a name, for example "PLC_1".
5. Insert another IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
6. Assign a name, for example "PLC_2".
7. Connect the PROFINET interface X1 of "PLC_1" and "PLC_2" to each another. The PROFINET interfaces of the CPUs are thus in the same subnet.
8. Insert another IO controller (e.g. CPU 1513-1 PN) from the hardware catalog.
9. Assign a name, for example "PLC_I-Device".

Procedure - Configuring I-device functionality

To configure I-device functionality for the PROFINET interface X1 of the IO controller "PLC_I-Device", follow these steps:

1. Select the PROFINET interface X1 of PLC_I-Device.
2. In the Inspector window, navigate to "Properties > General > Operating mode".
3. Enable the "IO device" option. The table below is editable.
4. Click on the entry "Assign to new IO controller" in the table.
5. Select the PROFINET interface of an IO controller from the drop-down list, for example, "PLC_1.PROFINET interface_1."
   The next free device number in the higher-level IO system will be assigned by default as the device number.
6. Repeat steps 4 and 5 for a further IO controller.
7. If required, activate the "Parameter assignment of PN interface by higher-level IO controller" option.
   The drop-down list below is activated. In this case, select a higher-level IO controller from the drop-down list, for example, PLC_1.
8. If necessary, enable additional options. The "Activate DCP write protection" option is enabled by default.

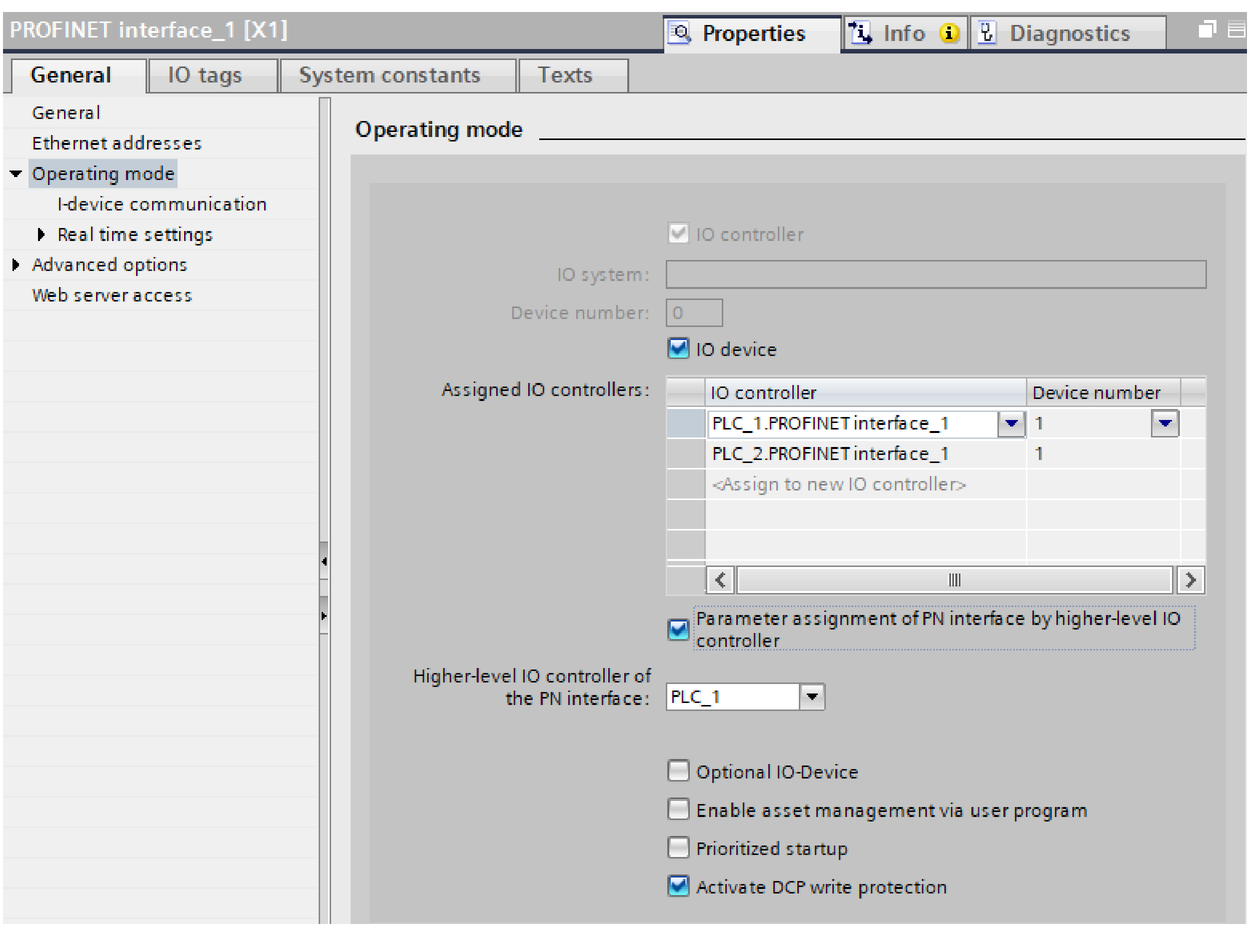


Figure 6-28 Configuring operating mode of I-device

Result: You have configured an S7-1500 CPU as the I-device and assigned it to two IO controllers.

Procedure - Creating and configuring transfer areas

Transfer areas that are assigned to IO controllers are defined as the I/O interface. At least one transfer area must be assigned to each IO controller accessing the shared I-device.

To configure a transfer area, follow these steps:

1. Select the PROFINET interface X1 of the IO controller PLC_I-device.
2. In the Inspector window, navigate to "Properties > General > Operating mode > I-device communication".
3. Double-click the table entry "Add new". A new transfer area is created.
4. Repeat steps 1 through 3 to create further transfer areas.

Intermediate result: You have created the required transfer areas. You can view the transfer areas with their basic settings in the Inspector window in the I-device communication area.

To configure a transfer area, e.g. change an input area to an output area, follow these steps:

1. Select the desired transfer area, for example, Transfer area_1, in the Inspector window under "Properties > General > Operating mode > I-device communication".
   The selected transfer area is displayed in the Inspector window.
2. Select the desired type from the drop-down list "Address type" under "Partner" or "Local." The address type at the partner will be automatically adjusted in the event of changes.

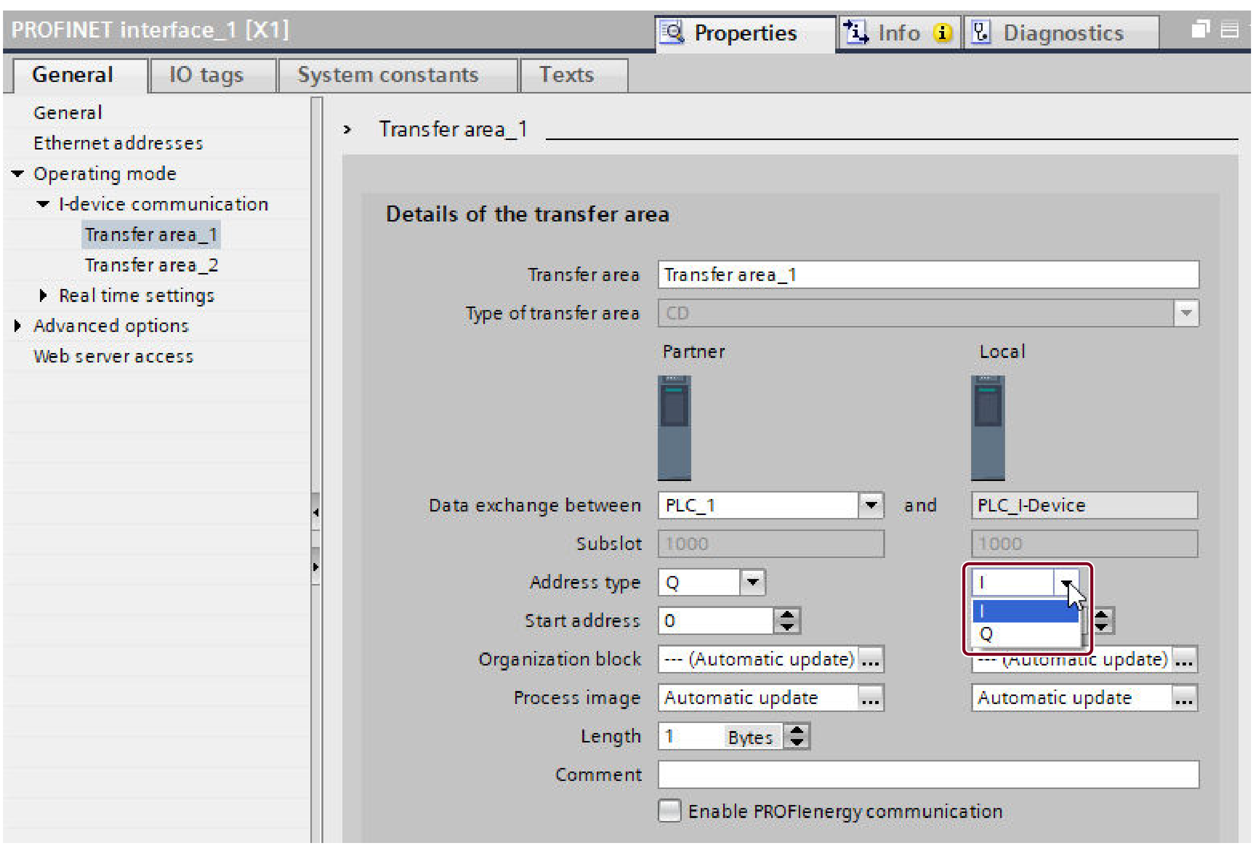


Figure 6-29 Configuring a transfer area of an I-device

Result: You have created and configured transfer areas of the I-device.

Procedure - Adapting the send clock in the real-time settings

You must set the same send clock for each IO controller that has access to modules or submodules of the shared I-device. This ensures that update times are calculated correctly with respect to the communication load.

To adjust and check these settings, proceed as follows:

1. At an assigned IO controller, select the PROFINET interface that is connected to the shared I-device.
2. In the Inspector window, navigate to the "General > Advanced options > Real-time settings > IO communication" area.
3. In the drop-down list, select a common send clock.
4. Repeat steps 1 to 3 for the other assigned IO controller.

In the common STEP 7 project, all the IO controllers that have access to the shared I-device are configured. In this case, it is possible to also set shorter send clocks on the IO controllers than supported by the shared I-device (send clock adaptation).

Compiling and loading

When you select an IO controller and click on "Compile" in the toolbar, the configuration of this IO controller and the shared I-device is checked for data consistency.
Alternatively, mark both IO controllers that access the shared I-device. When you click on "Compile" in the toolbar, the full configuration (both IO controllers and the shared I-device) is checked for data consistency.

You must compile the configurations for the different IO controllers and shared I-device and load them into the CPUs one after the other.

6.3.8 Configuring a shared I-device in different projects

Below, you can find a description of how you configure an S7-1500 as an I-device with STEP 7 Version 13 or higher and then use it in 2 projects as a shared device.

A "distributed" configuration with different engineering tools for different IO controller families is generally also possible here. The procedure described below is based on STEP 7 V13 and is limited to a configuration with 2 IO controllers of the S7-1500 family that share the transfer areas of an I-device as a shared device. The I-device itself is also an S7-1500 CPU.

Create 3 projects with one IO controller each (PLC-I-Device, PLC_1 and PLC_2).

The "PLC-I-Device" project is used to configure the I-device. The PROFINET GSD variant of "PLC-I-Device" is used in the "PLC_1" and "PLC_2" projects in order to assign the transfer areas in the respective higher-level IO controller.

Shared I-device concept

To introduce the shared I-device concept, 2 roles are introduced here:

- The role of manufacturer (e.g., machine manufacturer): The manufacturer configures and programs an I-device that performs a particular automation task. Transfer areas are defined as the I/O interface to the operator of the machine. These transfer areas can be assigned to different IO controllers. For the connection to higher-level IO controllers, the manufacturer provides a PROFINET GSD file and discloses the transfer areas via which the I-device can be accessed.
- The role of the operator: The operator uses the I-device as a PROFINET GSD variant during configuration of the PROFINET IO system and, in this process, specifies the I/O addresses under which the IO controllers access the transfer areas.

Manufacturer view

You assign the following parameters for an S7-1500 CPU as an I-device, with centralized and distributed I/O, the desired transfer areas, and the number of IO controllers having access to this I-device (always greater than 1 for a shared device!).

Special consideration: The I-device is configured without a higher-level IO controller. As a result, only the local I/O addresses of the transfer area are available (= "Address in the I-device") in order to create the user program for editing the addresses from the transfer area. The I-device that has been completely configured except for the connection to the higher-level IO controller is loaded to the S7-1500 CPU.

You export a PROFINET GSD file from the I-device configuration.

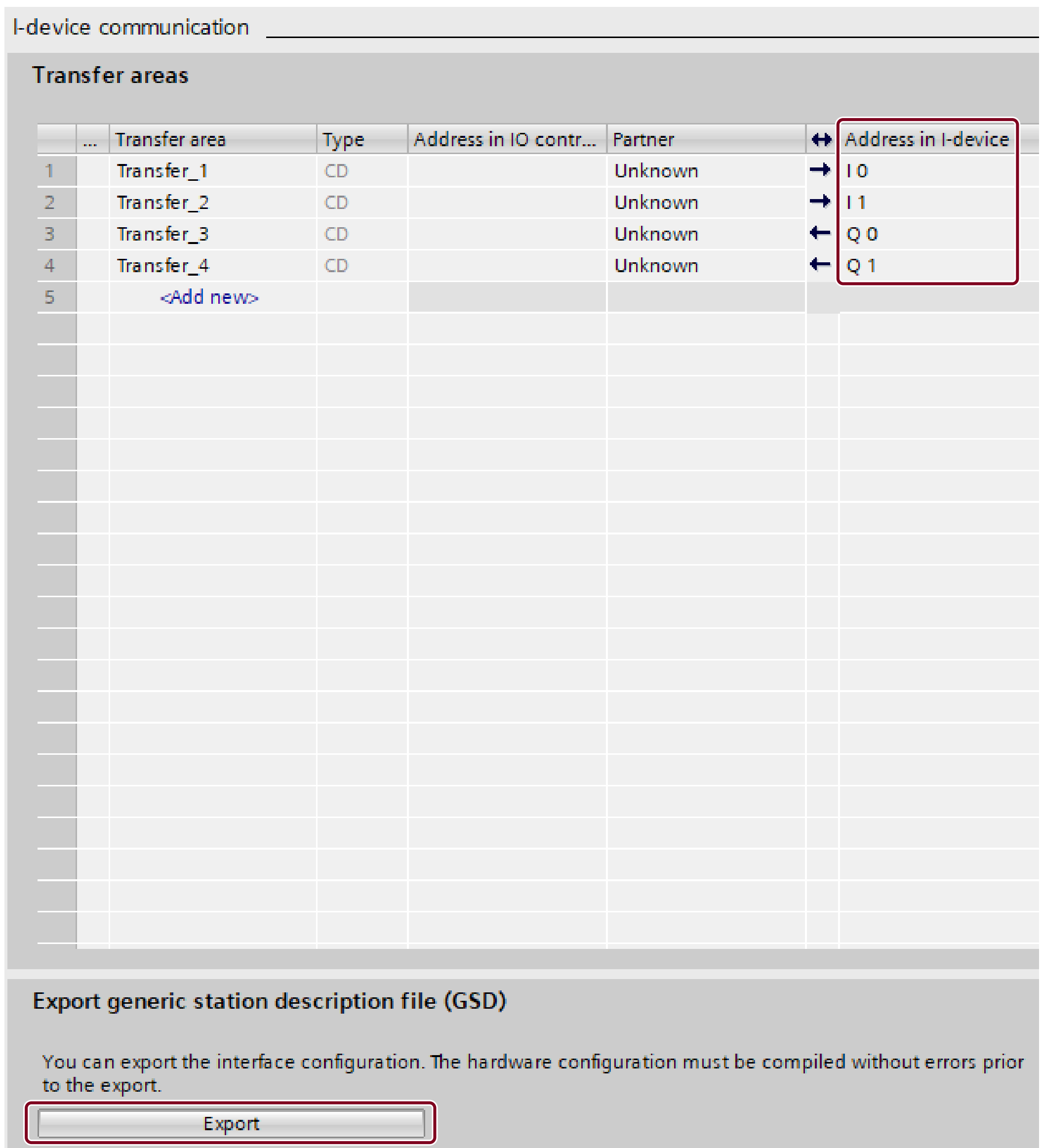


Figure 6-30 Exporting an I-device as a GSD file

Operator view

You must install the PROFINET GSD file created from the I-device configuration in all engineering systems that are involved in configuring a PROFINET IO system with this shared I-device. If all uses of this I-device will be configured with STEP 7 V13, it is sufficient to install the GSD file in STEP 7.

You configure the I-device as a GSD variant on the PROFINET IO system in the projects involved. In STEP 7 V13, this I-device can be found under "Other field devices > PROFINET IO > PLCs & CPs" following installation.

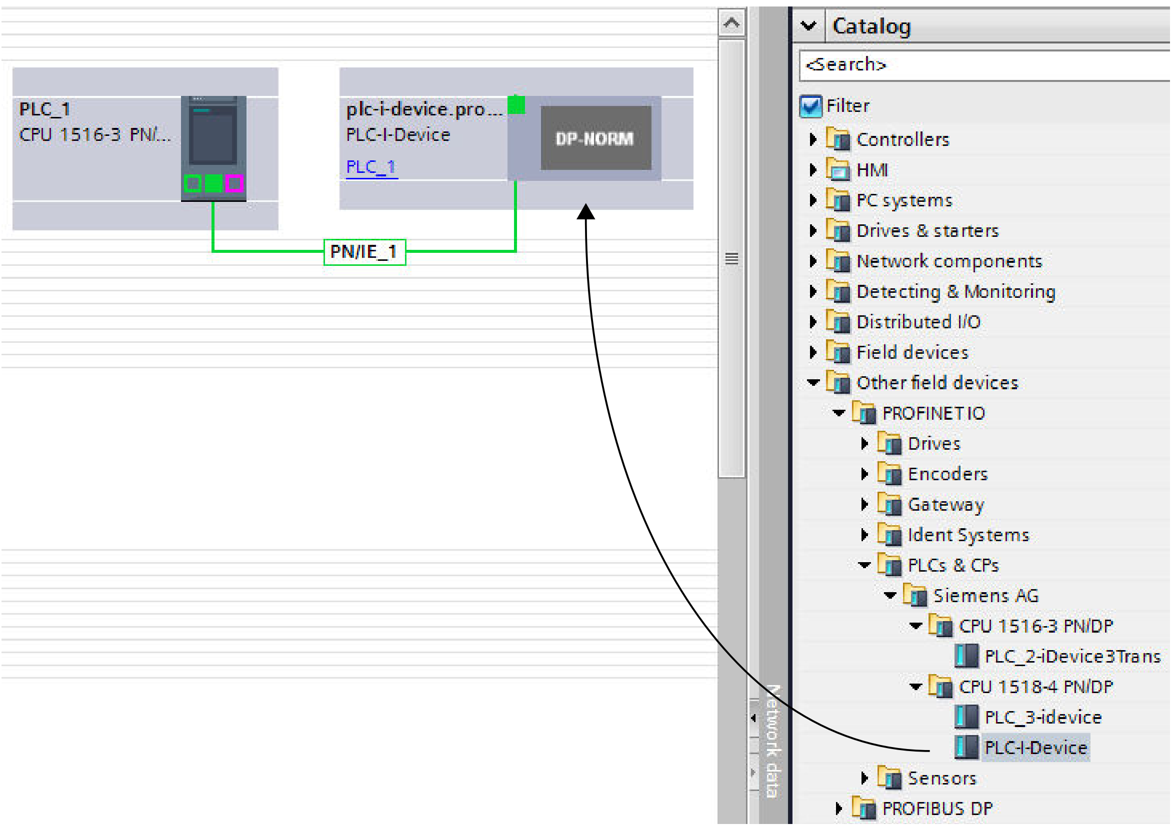


Figure 6-31 Configuring an I-device as a GSD file

In each of the projects involved, you define which transfer areas are assigned exclusively to the higher-level IO controller (default setting: all). Set the remaining transfer areas to "Outside of project". As a result of this setting, the local IO controller has no access to this transfer area and it can therefore be assigned to another IO controller in another project.

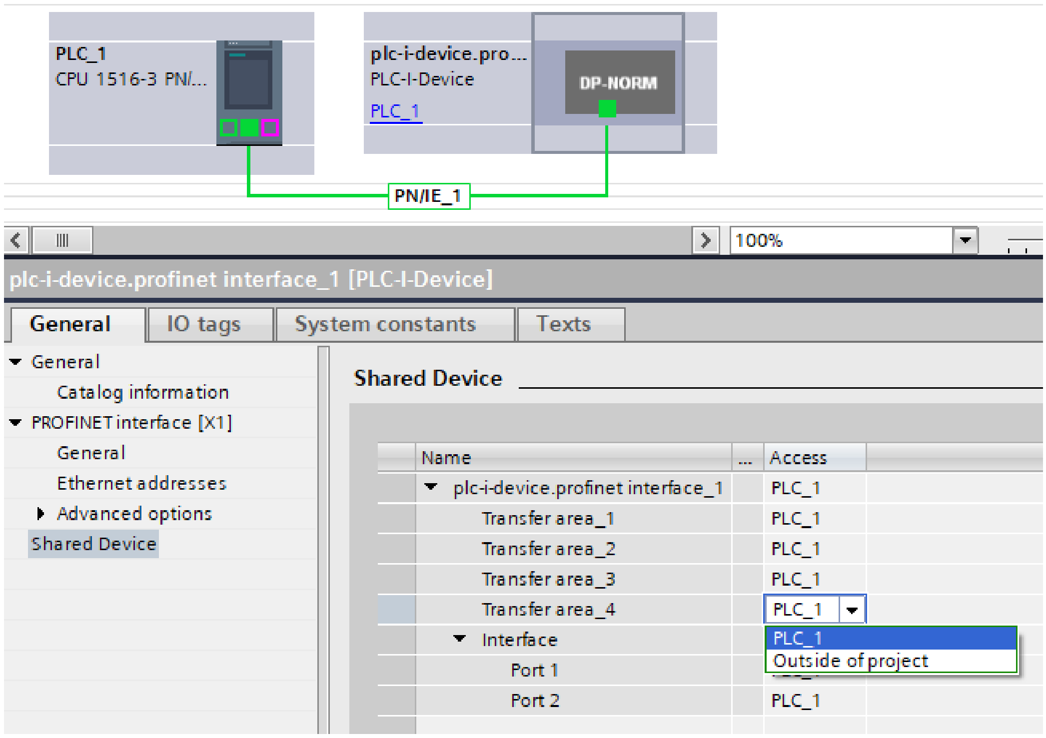


Figure 6-32 Setting the access to the shared I-device

You adapt the addresses from the view of the IO controller in the device overview. To open the device overview, double-click the I-device.

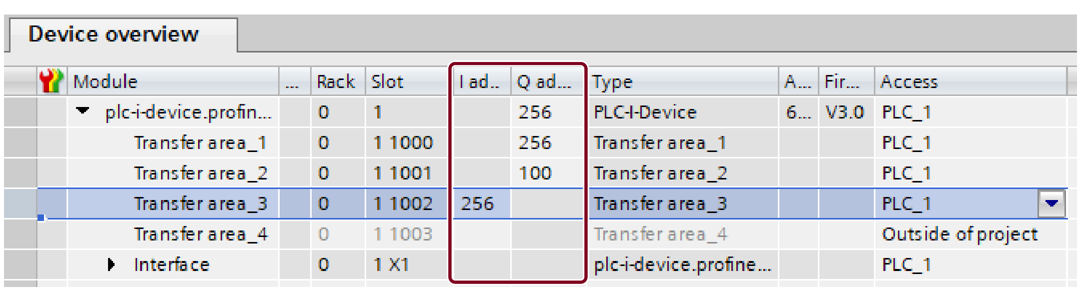


Figure 6-33 I/O addresses of the transfer areas in the device overview

Requirements

- STEP 7 as of V13

Procedure - Creating the PLC-I-device project

To create the project with a shared I-device, follow these steps:

1. Start STEP 7.
2. Create a new project with the name "PLC-I-device".
3. Insert, for example, a CPU 1518-4 PN/DP from the hardware catalog in the network view.
4. Assign a name, for example "PLC-I-Device".
5. Double-click the IO device and configure all required modules and submodules.
6. Assign the module parameters.
   In particular, the following settings for the CPU are necessary in the area of the PROFINET interface [X1]:
   – Activate the "IO device" option in the "Operating mode" area.
   – Activate the "Parameter assignment of PN interface by higher-level IO controller" option in the "Operating mode" area.
   – Configure the transfer areas in the "Operating mode > I-device communication" area. The "Address in IO controller" column remains empty because no IO controller is assigned.
   Note: To change an input area to an output area, and vice versa, you must navigate to the area of the corresponding transfer area.

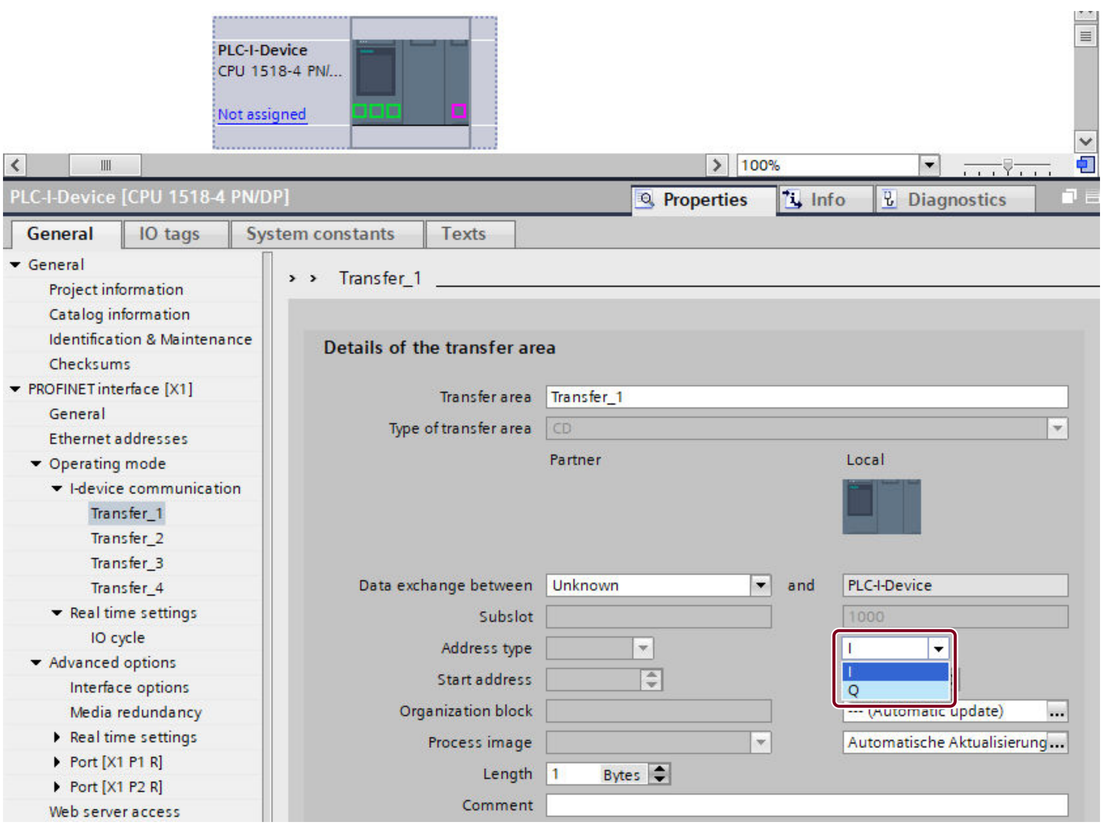


Figure 6-34 Changing the address type for the transfer area

   – Select the number of IO controllers outside the project with access to the I-device. If there is one IO controller outside the project, select "1" ("Operating mode > Real time settings", section "Shared device").

7. Click the "Compile" button in the toolbar to check your project for consistency errors.
   Inconsistent projects are not exported as GSD file by STEP 7.
8. Save the project.
9. Click the "Export" button ("Operating mode > I-device communication" area, "Export generic station description file (GSD) section").
   If you do not change the name in the Export dialog, the GSD file has a name in the form "GSDML-V2.31-#Siemens-PreConf_PLC-I-Device-20130925-123456", for example.

Procedure - Creating the PLC_1 project

To create the first project with a shared I-device, follow these steps:

1. Start STEP 7.
2. Install the PROFINET GSD file from the export of the I-device CPU (PLC-I-Device).
3. Create a new project with the name "PLC_1".
4. Insert, for example, a CPU 1516-3 PN/DP in the network view.
5. Assign a name, for example "PLC_1".
6. Insert the I-device from the hardware catalog (Hardware catalog: "Other field devices > PROFINET IO > PLCs & CPs").
7. Assign the I-device to the IO-controller "PLC_1".
8. Select the "Shared Device" area in the properties of the I-device.
   In the table, all transfer areas and the PROFINET interface are assigned to the local IO controller (PLC_1).
9. Define the transfer areas to which the "PLC_1" CPU should **not** have access. Select the "Outside of project" entry for these areas. These transfer areas are provided for the "PLC_2" CPU.
10. Save the project.

Procedure - Creating the PLC_2 project

To create the second project with a shared I-device, follow these steps:

1. Start STEP 7 once again.
   A new instance of STEP 7 opens.
2. In the new instance, create a new project with the name "PLC_2".
3. Insert, for example, a CPU 1516-3 PN/DP in the network view.
4. Assign a name, for example "PLC_2".
5. Insert the I-device from the hardware catalog (Hardware catalog: "Other field devices > PROFINET IO > PLCs & CPs").
6. Assign the I-device to the IO-controller "PLC_2".
7. Adapt the access to the transfer areas as in the "PLC_1" project. Ensure that no duplicate assignments result.
8. Adapt the parameters of the subnet and PROFINET interface. Because the shared I-device involves the same device in different projects, these data must match.
9. Save the project.

Both projects now have an identically configured shared I-device. The IO controller access and the parameters of the PROFINET interface must still be checked in the various projects during the next step.

Summary - Assigning parameters for access to the shared device

The transfer areas are automatically assigned to the local IO controller. To change the assignment, follow these steps:

1. Click the "PLC-I-Device" device in the network view of the "PLC_1" project, and select the "Shared Device" area.
2. A table shows which CPU has access to each of the configured transfer areas. The default setting is that the local CPU has access to all modules and submodules.
3. Keep the "PLC_1" setting for all transfer areas that are to remain in the address area of the local CPU

   Select the "Outside of project" setting for all transfer areas that are to be located in the address area of the "PLC_2" CPU from the "PLC_2" project. This means that an IO controller outside the project is to have access to the transfer area.
4. Follow the same procedure for the remaining projects.
5. Finally, check whether the settings for access are "complementary" for each module or submodule in both projects. This means that if the local CPU has access in one project, the "Outside of project" option must be set in the other project and vice versa.

   Special consideration: The "Outside of project" option for the PROFINET interface and therefore for the ports makes the associated parameters read-only and they cannot be changed. Parameters of the PROFINET interface and port parameters can only be edited in the project in which the PROFINET interface is assigned to the local CPU. The ports can be interconnected in both projects regardless of this.
6. Check whether the same IP address parameters and device name are set for the shared device in all projects.

   Check whether the same S7 subnet ID is set in all projects for the subnet to which the shared device is connected (subnet properties, "General" area in the Inspector window).

NOTE

If you make changes to the I-device (e.g., change the number or length of the transfer areas):

Export the I-device as a GSD file again. Re-install the GSD file in each project that uses the I-device as a shared device. Make sure that only one IO controller has access to a transfer area.

Procedure - Adapting the send clock in the real-time settings

You must set the same send clock for each IO controller that has access to modules or submodules of the shared I-device. This ensures that update times are calculated correctly with respect to the communication load.

To adjust and check these settings, use one of the following options:

- If you configure the IO controller with STEP 7 (TIA Portal):
  - Open the corresponding project.
  - At the IO controller, select the PROFINET interface that is connected to the shared I-device.
  - In the Inspector window, navigate to the "General > Advanced options > Real-time settings > IO communication" area.
  - In the drop-down list, select a common send clock.
- If you configure the IO controller with a different engineering tool:
  - Select the PROFINET interface of the shared device in STEP 7 (TIA Portal).
  - In the Inspector window, go to "General > Advanced options > Real-time settings >IO communication".
  - Read the send clock of the shared device.
  - Enter the read send clock in the other engineering tool.

Special consideration: If you configure **all** IO controllers that have access to the shared I-device in STEP 7 (TIA Portal or V5.5), you can set shorter send clocks on the IO controller than supported by the shared device (send clock adaptation).

Compiling and loading

You must compile the configurations for the different IO controllers and load them to the CPUs one after the other.

Due to the distributed configuration with separate projects, STEP 7 does not output consistency errors in the case of incorrect access parameter assignment. Example for incorrect access parameter assignment:

- Several IO controllers have access to the same module
- IP address parameters or send clocks are not identical

These errors do not show up until operation and are output as configuration errors, for example.

NOTE

After a configuration has been loaded in the IO controller, non-assigned modules or submodules in the IO device retain their current parameter assignment state. This ensures the independence to configurations of other IO controllers.

6.3.9 Module-internal shared input/shared output (MSI/MSO)

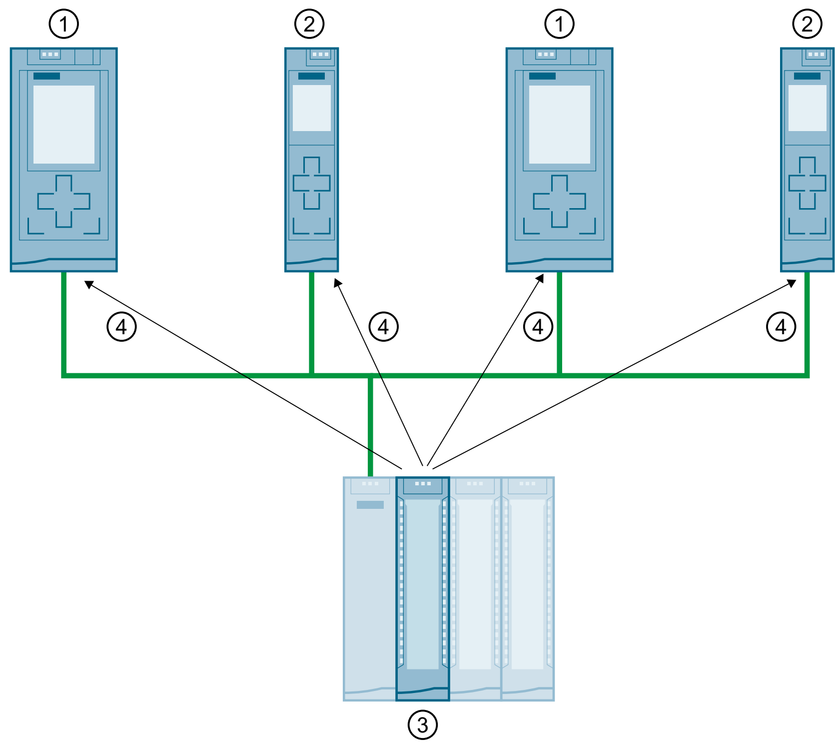
Introduction

This section describes the module-internal shared input/shared output (MSI/MSO) functionality for I/O modules that are operated on PROFINET.

Module-internal shared input/shared output functionality

The module-internal shared input (MSI) function allows an input module to make its input data available to up to 4 IO controllers. Each controller has read access to the same channels.

The following figure shows the MSI functionality.

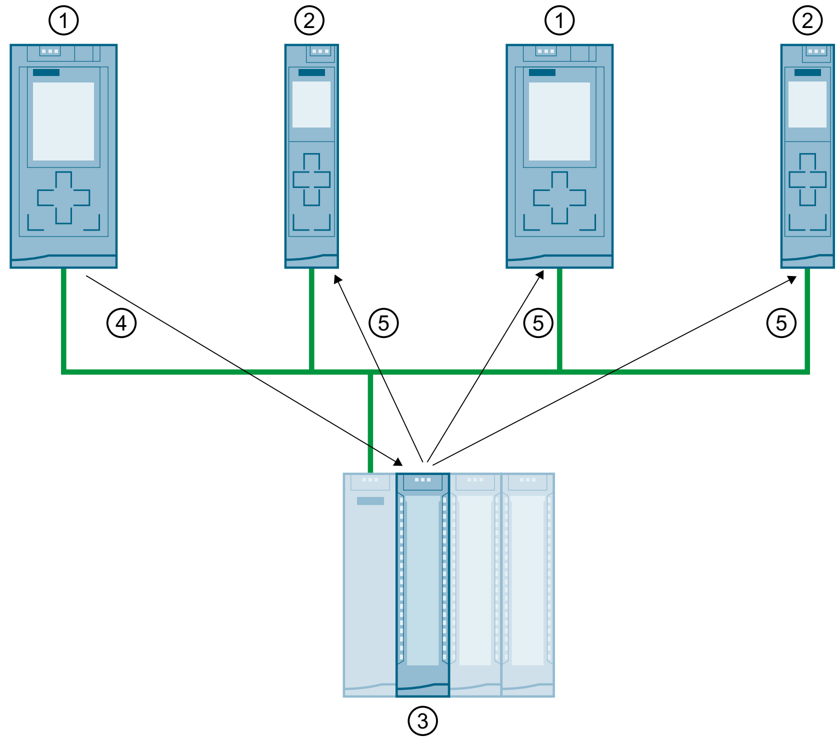


① CPU 1516-3 PN/DP as IO controller
② CPU 1511-1 PN as IO controller
③ Input module with MSI
④ Read access to the input channels of the input module
⑤ Write access to the channels of the I/O module (only with MSO)

Figure 6-35 Example configuration with MSI

The module-internal shared output (MSO) function allows an output module to make its output data available to up to 4 IO controllers. An IO controller has write access to the channels of the output module. Up to 3 IO controllers can additionally have read-access to the channels.

The following figure shows the MSO functionality.



①	CPU 1516-3 PN/DP as IO controller
②	CPU 1511-1 PN as IO controller
③	Output module with MSO
④	Write access to the output channels of the output module
⑤	Read access to the output channels of the output module

Figure 6-36 Example configuration with MSO

Advantages of MSI/MSO

Module-internal shared input/shared output (MSI/MSO) offers the following advantages:

- Real-time acquisition in multiple CPUs
- Lower costs due to saving on additional IO devices and modules
- Lower space requirements due to saving on additional IO devices and modules
- Reduced communication load because no CPU-CPU communication is needed
- No additional programming effort is needed for CPU-CPU communication

Requirements for the use of MSI/MSO

Observe the following requirements:

- MSI/MSO can only be used with PROFINET IO
- Configuration software: STEP 7 (TIA Portal) as of V12 SP1 with GSD file; the modules are integrated in the hardware catalog as of V13.
- The IM 155-5 PN ST interface module and the modules support MSI/MSO as of firmware version V2.0.0.

Constraints for using of MSI/MSO

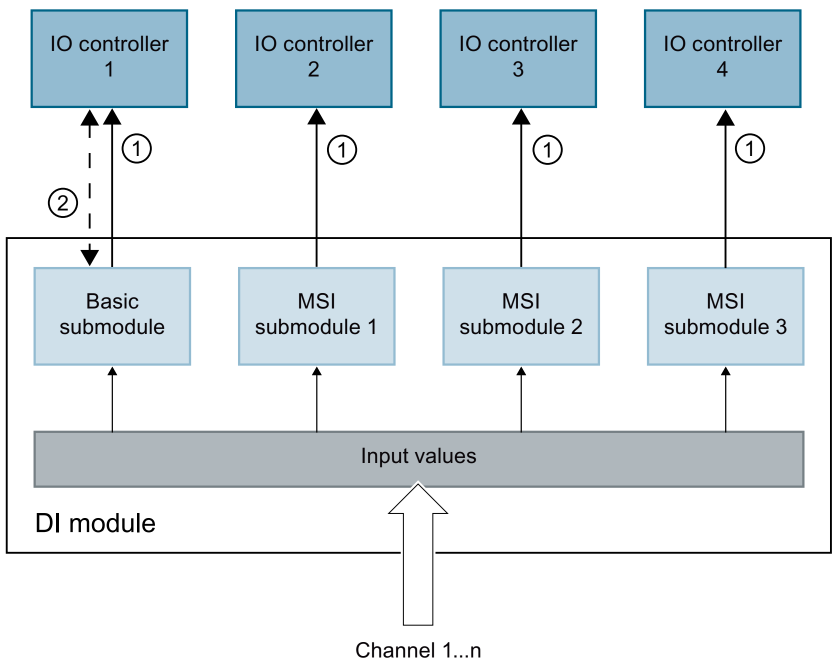
Note the following constraints:

- The use of MSI/MSO is not possible in the case of module grouping.
- Modules with MSI/MSO cannot be operated in isochronous mode.
- The maximum number of IO controllers is dependent on the interface module. You can find out how many IO controllers the interface module supports in the equipment manual for the respective interface module.

MSI submodules

The input values of all channels are copied to a basic submodule and up to 3 other MSI submodules during MSI configuration of an input module. The channels of the module are then available with identical input values in the basic submodule and the MSI submodules. The MSI submodules can be assigned to up to 3 IO controllers when the module is used in a shared device. Each IO controller has read access to the same channels.

The following figure shows a digital input module with the basic submodule and 3 MSI submodules. Each submodule is assigned to an IO controller. Diagnostics and parameter assignment of the digital input module can be performed from the IO controller 1 via the basic submodule.



① Read access

② Parameter assignment and system diagnostics

Figure 6-37 DI module with MSI submodules

Value status (Quality Information, QI)

The meaning of the value status depends on the submodule to which it pertains.

With basic submodule (= 1st submodule), the "0" value status indicates that the value is incorrect.

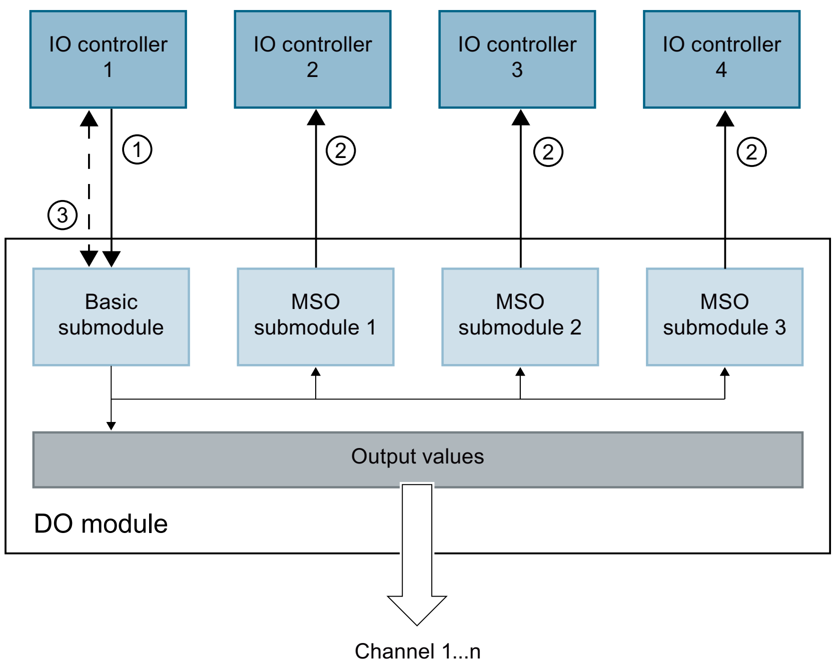
With an MSI submodule (2nd to 4th submodule), the "0" value status indicates that the value is incorrect or the basic submodule has not yet been configured (not ready).

MSO submodules

During MSO configuration of an output module, the output values of all channels of the module are copied from a basic submodule to up to 3 other MSO submodules. The channels of the module are then available with identical values in the basic submodule and the MSO submodules. The MSO submodules can be assigned to up to 3 IO controllers when the module is used in a shared device:

- The IO controller to which the basic submodule is assigned has write access to the outputs of the module. The basic submodule therefore occupies output addresses in the process image of the IO controller.
- The IO controllers to which the MSO submodules are assigned have read access to the outputs of the module. MSO submodules therefore occupy input addresses in the process image of the IO controller.

The following image shows a digital output module with the basic submodule and 3 MSI submodules. Each submodule is assigned to an IO controller. Diagnostics and parameter assignment of the digital output module can be performed from IO controller 1 via the basic submodule.



① Write access

② Read access

③ Parameter assignment and system diagnostics

Figure 6-38 DQ module with MSO submodule

Value status (Quality Information, QI)

The meaning of the value status depends on the submodule to which it pertains.

With basic submodule (= 1st submodule), the "0" value status indicates that the value is incorrect.

With MSO submodule (= 2nd to 4th submodule) the "0" value status indicates that the value is incorrect or one of the following errors has occurred:

- The basic submodule parameters have not yet been assigned (not ready for operation).
- The connection between the IO controller and the basic submodule has been interrupted.
- The IO controller from the basic submodule is in the "STOP" or "POWER OFF" state.

Configuring I/O modules with MSI/MSO submodules

Requirements

- STEP 7 as of V13
- IO device supports MSI/MSO (for example IM 155-5 PN ST as of firmware version 2.0.0)

Procedure

1. In the network view of STEP 7, insert an interface module, for example IM 155-5 PN ST as of V2.0.
2. Double-click the IO device.
   You are now in the device view.
3. Place the I/O modules from the hardware catalog in a suitable slot.
4. Add MSI/MSO submodules to the I/O modules:
   - Input modules: Select the number of MSI submodules under "Module parameters > DI Configuration" or "AI Configuration" in the area "Copy of module for Shared Device (MSI)".
   - Output modules: Select the number of MSO submodules under "Module parameters > DO configuration" or "AW configuration" in the area "Copy of module for shared device (MSO)".

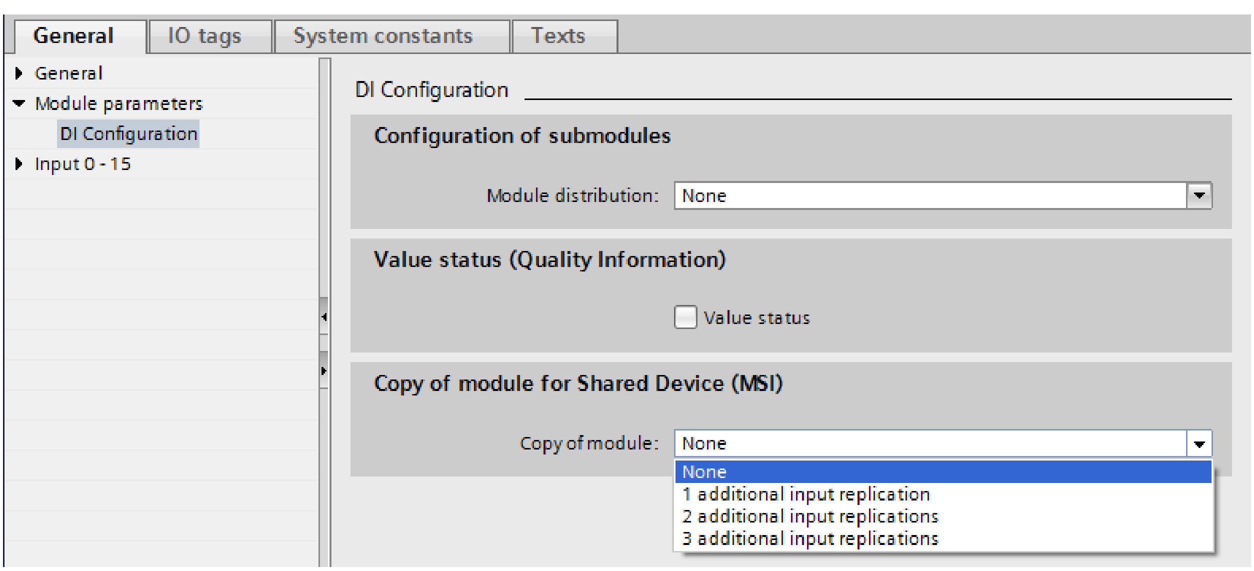


Figure 6-39 Configuring I/O modules with MSI/MSO

Assigning MSI/MSO submodules to an IO controller

You can assign the submodules in a shared device to an IO controller.

You can find additional information under "Configuring a shared device for different projects (Page 157)".

Configuring access to a shared device and the module-internal shared input / shared output (MSI /MSO) function

You can learn how to access a shared device and the MSI /MSO function in STEP 7 with this FAQ (https://support.industry.siemens.com/cs/ww/en/view/109736536).

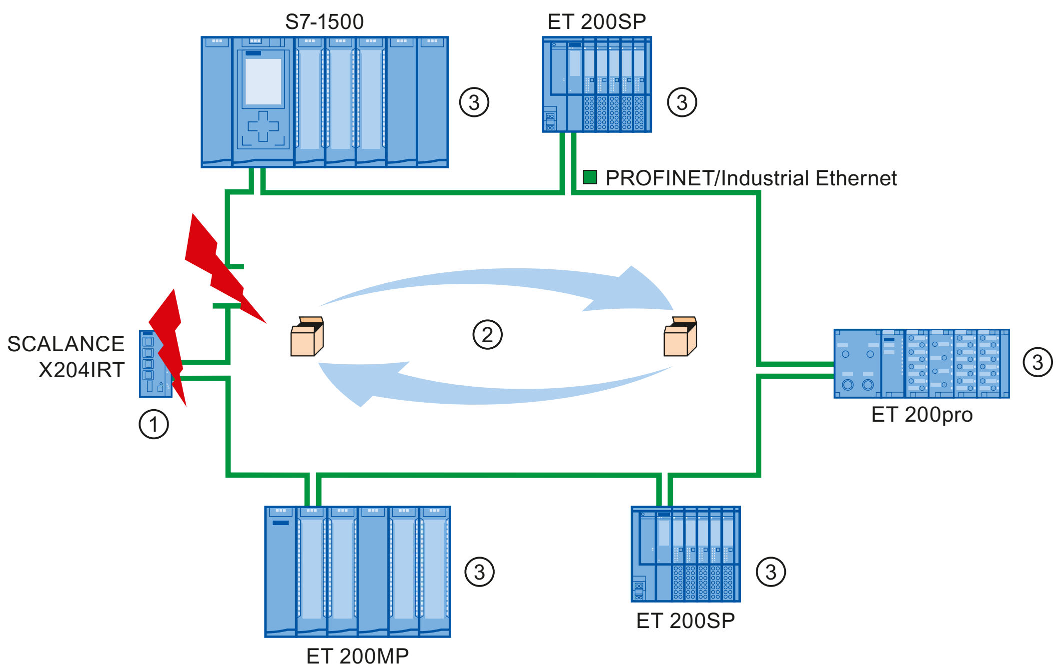
6.4 Media redundancy (ring topologies)

In order to increase the network availability of an Industrial Ethernet network with optical or electrical linear bus topologies, you can convert a linear bus topology to a ring topology by joining the ends together.

Media redundancy in ring topologies

Devices in a ring topology can be IO devices, IO controllers, external switches and/or the integrated switches of communication modules.

To set up a ring topology with media redundancy, you need to bring together the two free ends of a linear bus topology in one device. Closing the linear bus topology to form a ring is achieved with two ports (ring ports) of a device in the ring. One device of the resulting ring then takes over the role of the redundancy manager. All other devices in the ring are redundancy clients.



① Redundancy manager

② Test frames

③ Redundancy clients

Figure 6-40 Media redundancy in ring topology

The ring ports of a device are the ports that establish the connection to the two neighboring devices in the ring topology. The ring ports are selected and set in the configuration of the relevant device (is also preset, if applicable).

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. The devices are available again after reconfiguration.

In the redundancy manager, one of the two ring ports is blocked in uninterrupted network operation for normal communication so that no data frames are circulated. In terms of data transmission, the ring topology is a linear bus topology. The redundancy manager monitors the ring for interruptions. For this purpose, it sends test frames not only from ring port 1 but also from ring port 2. The test frames pass through the ring in both directions until they arrive at the other ring port of the redundancy manager.

An interruption of the ring can be caused by loss of the connection between two devices or by failure of a device in the ring.

If the test frames of the redundancy manager no longer arrive at the other ring port during an interruption of the ring, the redundancy manager connects its two ring ports. This substitute path once again restores a functioning connection between all remaining devices in the form of a linear bus topology.

The time between the ring interruption and restoration of a functional linear topology is known as the reconfiguration time.

As soon as the interruption has been eliminated, one of the two ring ports is blocked again in the redundancy manager. The redundancy clients are informed about the change and use the original paths to the other devices again.

Media redundancy method

The standard method of media redundancy in SIMATIC is MRP (Media Redundancy Protocol) with a typical reconfiguration time of 200 ms. Up to 50 devices can participate per ring.

In addition, the real-time capable media redundancy process MRPD (Media Redundancy with Planned Duplication of frames) is also available.

6.4.1 Media Redundancy Protocol (MRP)

Media Redundancy Protocol (MRP)

The "MRP" process works in conformity with the Media Redundancy Protocol (MRP) that is specified in the standard IEC 62439-2.

Requirements

- All devices in the ring support MRP.
- You have complied with the rules for topology set out below. STEP 7 monitors compliance with the rules during compilation and outputs corresponding alarms.

Topology

The following schematic shows a possible topology for devices in a ring with MRP. The devices inside the shaded oval are in the redundancy domain.

Example of a ring topology with the MRP media redundancy protocol:

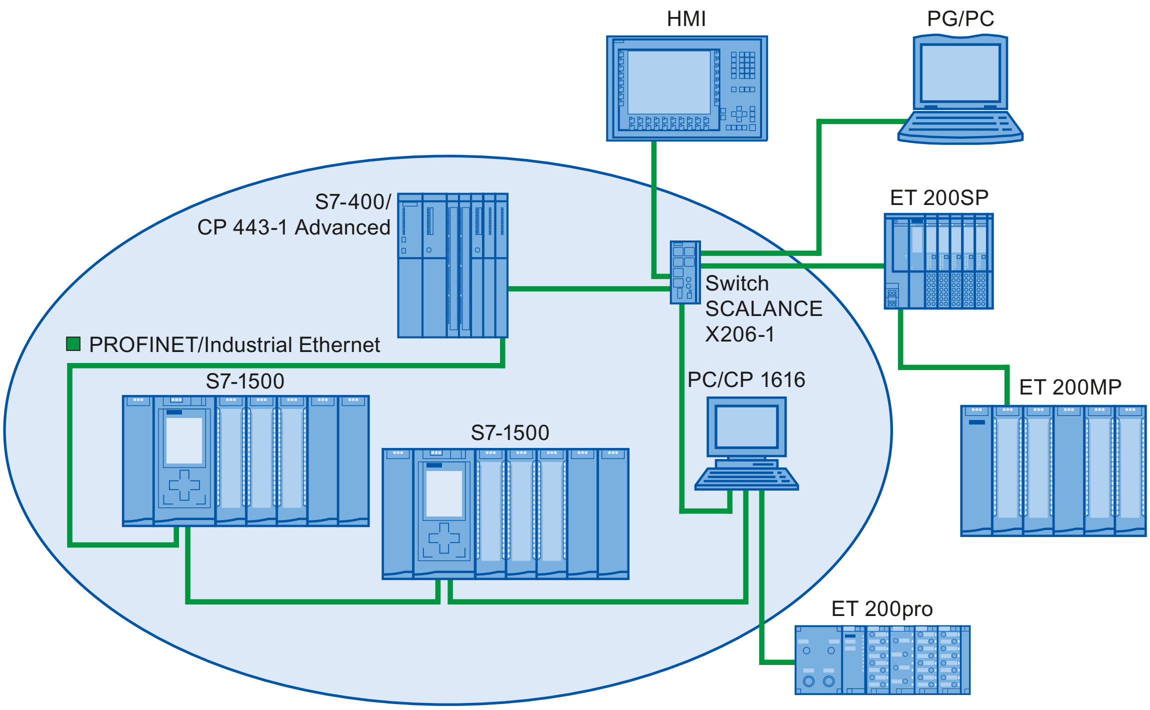


Figure 6-41 Example of a ring topology with the MRP media redundancy protocol:

The following rules apply to a ring topology with media redundancy using MRP:

- All devices must be connected to one other via their ring ports.
- All devices in the ring belong to the same redundancy domain.
- A device in the ring takes on the task of a redundancy manager.
  - One device only has the role of "Manager". No other device may have the role of "Manager". Or
  - One device or multiple devices in the ring have the role of "Manager (auto)". The devices with the role of "Manager (auto)" then negotiate between themselves which device is to take on the role of redundancy manager. In this case, no device is permitted to have the role of "Manager".
- All other devices in the ring are redundancy clients.
- You can connect up to 50 devices in a ring.

Non MRP-compliant devices can, for example, be connected to the ring via a SCALANCE X switch or via a PC with a CP 1616.

Rules for loading the devices of an MRP domain

When loading devices of an MRP domain, frame loops can occur and result in network failure if there is an invalid MRP configuration.

Example: You change the MRP roles of several devices and consecutively load the configuration into the devices involved. Configurations can arise that contradict the roles mentioned above, for example, there may be devices that simultaneously have the "Manager" and "Manager (auto)" roles in the ring.

To ensure that an invalid MRP configuration does not result in the failure of the network, open the ring before loading.

Follow these steps:

1. Open the ring.
2. Load the error-free and consistent MRP configuration from your project onto all the devices involved and ensure that the devices are in data exchange mode (i.e. the application relations (ARs) are set up).
3. Close the ring.

Boundary conditions

MRP and RT

RT operation is possible with the use of MRP.

NOTE

The RT communication is disrupted (station failure) if the reconfiguration time of the ring is greater than the selected response monitoring time of the IO device. This is why you should select a response monitoring time of the IO devices that is sufficiently large.

MRP and IRT

IRT mode is not possible together with MRP.

If you want to use media redundancy together with IRT in a ring, only use devices that support MRPD.

MRP and TCP/IP (TSEND, HTTP, ...)

The TCP/IP communication with MRP is possible, because lost data packages are resent, if applicable.

MRP and 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 you must disable MRP in the configuration (the device may also not be part of the ring).

Information on S7-1500R/H

You can find information on the media redundancy method MRP of the redundant S7-1500R/H system in section Configuring PROFINET IO on a redundant S7-1500R/H system (Page 317).

Application example: Reading the MRP status in the user program

The "LPNDR_ReadMRPState" function block from the "LPNDR" block library helps you to determine the status of the MRP ring per program. The block reads the MRP information from the PROFINET device (MRP Manager) and outputs the status.

The "LPNDR" block library and the corresponding documentation can be found in this application example (https://support.industry.siemens.com/cs/ww/de/view/109753067).

6.4.2 Configuring media redundancy

Configuring MRP

Proceed as follows to create a PROFINET IO configuration with MRP in STEP 7:

1. Generate a ring via the port interconnections in the topology view. First interconnect the devices to a line topology. Connect the unassigned port of the last device in the line with the unassigned port of the first device.
   The following example shows one CPU 1516-3 PN/DP and two interface modules IM 155-6 PN HF that are interconnected in a ring in the topology view of STEP 7.

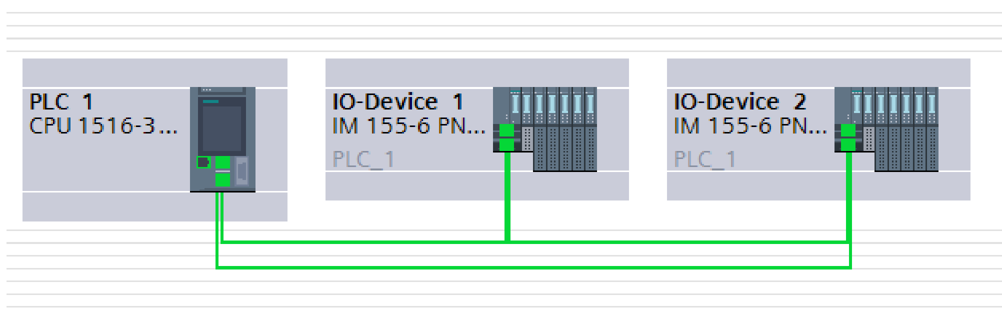


Figure 6-42 Configuring media redundancy

2. Select the PROFINET IO system in the network view.
3. In the Inspector window, navigate to "Properties" > "General" > "PROFINET" > "MRP domains" in the "Ring interconnections" field.
   This field shows you all the topological rings in the IO system with the associated MRP domains.
4. Select the ring generated above in the "Ring interconnections" field.
   The table below it shows all the PROFINET devices in the ring.

5. Set the media redundancy role for the PROFINET devices in the MRP role column.

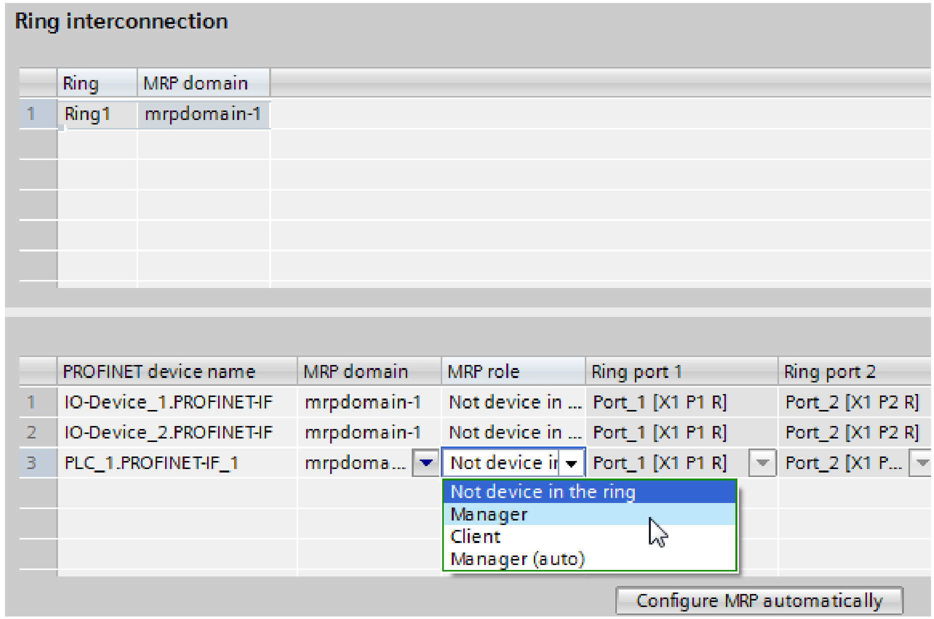


Figure 6-43 MRP domain

Automatic MRP configuration

You can also have the media redundancy roles assigned automatically for your PROFINET devices in the ring.

To have the media redundancy roles assigned automatically, click the "Configure MRP automatically" button. STEP 7 automatically assigns the media redundancy role for each device in the ring. After the automatic MRP configuration, you can make modifications to the media redundancy roles in the "MRP role" column.

"Media redundancy" setting options

Media redundancy role

Depending on the device used, the roles "Manager", "Manager (auto)", "Client" and "Not device in the ring" are available.

Rules:

- A ring must have precisely one device with the role "Manager". No additional devices with the role "Manager" or "Manager (auto)" are permissible. All the other devices may only have the "Client" role.
- If a ring has no device with the "Manager" role, the ring must at least have a device with the role "Manager (auto)". Any number of devices with the "Client" role may exist.
- Managers and clients of an MRP domain can be configured in different projects. In the project without a manager, you need to select the "Redundancy manager outside the project" option so that the configuration can be compiled. For multiple ring configurations, you need to set the option identically for each MRP domain.

Ring port 1 / Ring port 2

Select one at a time those ports you want to configure as ring port 1 or ring port 2. The drop-down list box shows the selection of possible ports for each device type. If the ports are set at the factory, then the fields are unavailable.

If you use single-stage commissioning, use the preset ring ports in STEP 7.

Diagnostic interrupts

If diagnostic interrupts to the MRP state are to be output in the local CPU, select the "Diagnostic interrupts" check box. The following diagnostic interrupts can be configured:

- Wiring or port error
  Diagnostic interrupts will be generated for the following errors in 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)
  - In case the ring is interrupted, a diagnostic interrupt "Ring open (incoming)" is generated.
  - When the ring is closed, the diagnostics "Ring open (outgoing)" is generated.

You can respond to these events in the user program by programming the appropriate response in the diagnostic error interrupt OB (OB 82).

Reference

You can find the procedure for configuring a common MRP ring and the topology with two projects in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/109741671).

6.4.3 Media redundancy with planned duplication of frames (MRPD; not for S7-1500R/H)

MRP extension "Media Redundancy with Planned Duplication of frames" (MRPD)

The MRP extension "Media Redundancy with Planned Duplication of frames" (MRPD) provides the advantage that, in the case of a failure of a device or a line in the ring, all other devices continue to be supplied with IO data without interruption and with short update times.

MRPD is based on IRT and MRP. To realize media redundancy with short update times, the PROFINET devices participating in the ring send their data in both directions. The devices receive this data at both ring ports so that there is no reconfiguration time.

Requirements for media redundancy with MRPD

- All the devices of the ring must support MRPD, for example interface module IM 155-6 PN HS as of firmware version 4.0.
- You have configured MRP for all the devices of the ring. You have assigned the MRP role "Not device in the ring" to devices that are not located in the ring.
- You have configured IRT for all the involved components.

Configuring MRPD

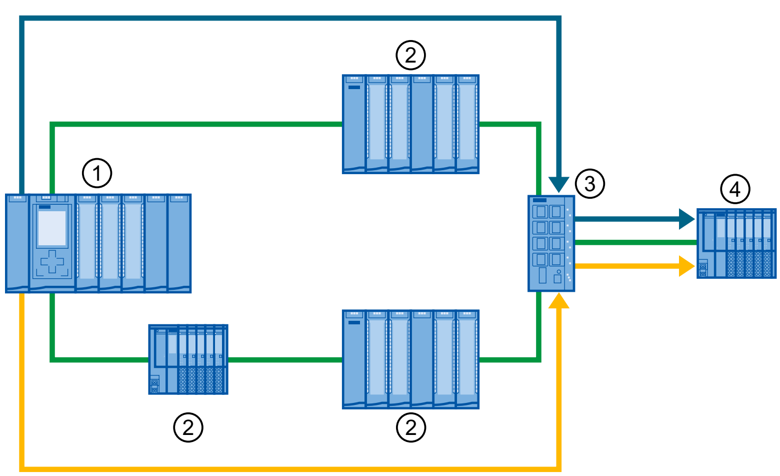
You do not have to explicitly activate MRPD in STEP 7. The function is available automatically as soon as all the requirements for MRPD are fulfilled.

Redundancy levels of IO devices with MRPD

The redundancy level of an IO device specifies how strongly the real-time communication is influenced in the case of a power interruption between an IO device and its IO controller.

- Full redundancy: No influence, because the IO controller and IO device are located in the same ring.
- Partial redundancy:
  - If the interruption takes place on a non-redundant part (line) between the IO device and IO controller, real-time communication is influenced.
  - If the interruption takes place on a redundant part (ring), real-time communication is not influenced.
- No redundancy: No redundant path between the IO device and its IO controller, communication is always influenced.

The figure below shows the redundancy levels of the IO devices for an example configuration with MRPD. The three devices in the ring and the switch have the redundancy level "Full redundancy". Device 4 has the redundancy level "Partial redundancy", because the connection between the switch and the device is not redundant.



①	IO controller in the ring
②	IO device in the ring with redundancy level "Full redundancy"
③	Switch in the ring with redundancy level "Full redundancy"
④	IO device in the spur with the redundancy level "Partial redundancy"

Figure 6-44 Example configuration with MRPD

The figure below shows the display of the redundancy levels in STEP 7 for the example configuration.

IO devices

PROFINET device name	RT class	Synchronization role	Redundancy level	DFP group
plc_1.profinet interface_1	RT,IRT	Sync master		
io-device_1	IRT	Sync slave	Full redundancy	
io-device_2	IRT	Sync slave	Full redundancy	
io-device_3	IRT	Sync slave	Full redundancy	
io-device_4	IRT	Sync slave	Partial redundancy	
switch_1	IRT	Sync slave	Full redundancy	

Figure 6-45 Display of the redundancy levels in STEP 7

6.4.4 Multiple rings

Multiple rings

Use multiple rings to achieve higher availability for PROFINET IO networks with star topology.
In a multiple ring configuration, several PROFINET lines lead from a switch (star topology). The individual PROFINET lines lead from IO device to IO device. Redundant PROFINET cables run back to the switch from the last IO device on each individual line.
The switch operates as a manager. The manager must have two ring ports for each ring. Multiple rings are possible. The SCALANCE X414 as of firmware version 3.10, for example, supports up to 4 rings.
The manager monitors all the rings individually: It checks for each particular ring (an MRP domain) whether the transmission path is intact. To do this, it uses an MRP instance in each case. An MRP instance is required for each connected ring (set up automatically by STEP 7).

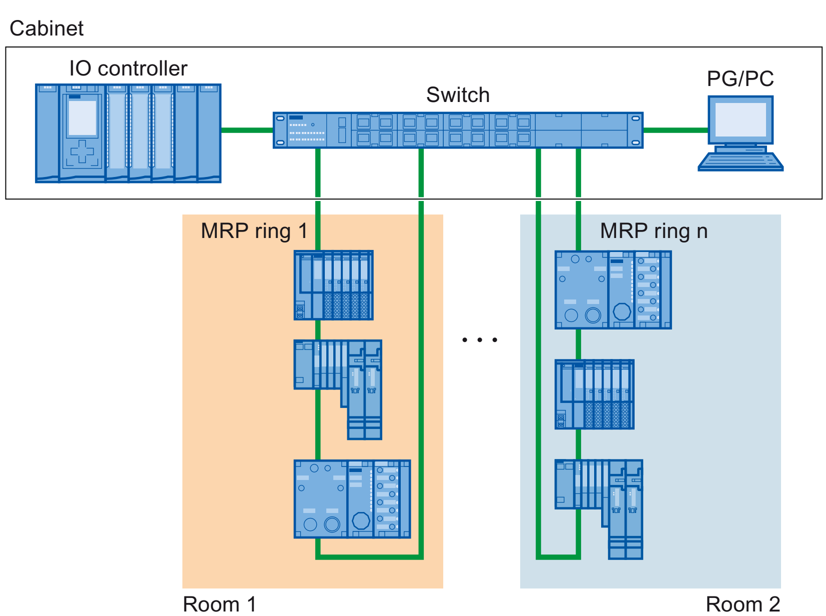


Figure 6-46 Configuration with several rings

Requirements

- SCALANCE X414 as of version 3.10
- SCALANCE X300 as of version 4.0 (configured via GSD file)

Rules for the configuration of multiple rings

- MRP role at multiple rings:
  - The device that belongs to all the rings must have the MRP role in every instance that is entered in the GSD file in the "SupportedMultipleRole" attribute.
  - The switches from the series SCALANCE X300 as of version 4.0 and the switch X414 as of version 4.10 support the MRP role "Manager" for multiple rings.
- If the device that belongs to all rings has the "Manager" role in one ring, no devices with the "Manager (Auto)" role may exist in this ring.

Configuring multiple rings

To configure an MRP configuration with multiple rings, follow these steps:

1. In the topology view, interconnect the ring ports of the devices that are intended to belong to an MRP domain to form a ring.
2. Select the PROFINET IO system in the network view.
3. In the Inspector window, navigate to "Properties" > "General" > "PROFINET" > "MRP domains" in the "Ring interconnections" field.
   This field shows you all the topological rings in the IO system with the associated MRP domains.
4. Select one of the rings generated above in the "Ring interconnections" field.
   The table below it shows all the PROFINET devices in the ring.
5. Set the media redundancy role for the PROFINET devices in the MRP role column.

Example of multiple rings

The following figure shows an example configuration for multiple rings.

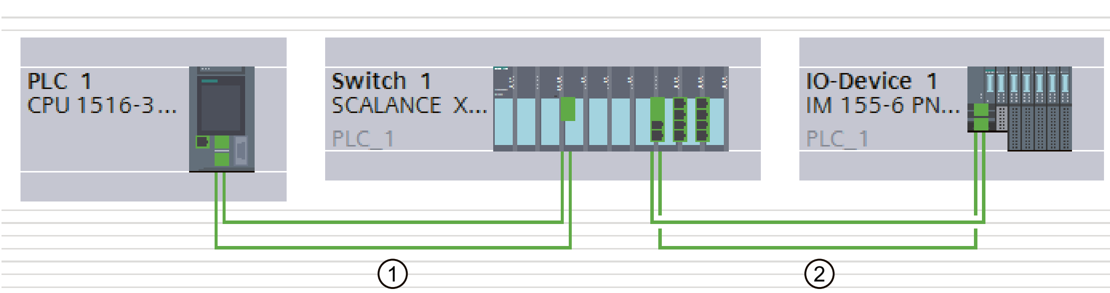


Figure 6-47 Example configuration with two rings ① and ②

In the example, Switch 1 belongs to two MRP rings. Ring 1 is formed by Switch 1 and PLC 1, Ring 2 by Switch 1 and IO device 1.
STEP 7 automatically assigns the "Manager" role to the switch. The other devices receive the "Client" role.

The manager is located at the point of intersection of the two rings 1 and 2. The manager monitors the two rings separately. To do this, it uses two MRP instances.
One MRP instance checks whether all devices in ring 1 are reachable, another instance monitors whether all devices in ring 2 are reachable (only one device in each case in the example).
You can configure each MRP instance separately.
The following figure shows the two MRP instances in the manager (PROFINET interface of the switch). Here in the example, MRP instance 1 checks whether the devices in the MRP domain "mrpdomain-1" can be reached. The MRP instance 2 is responsible for monitoring the devices of the MRP domain "mrpdomain-2".

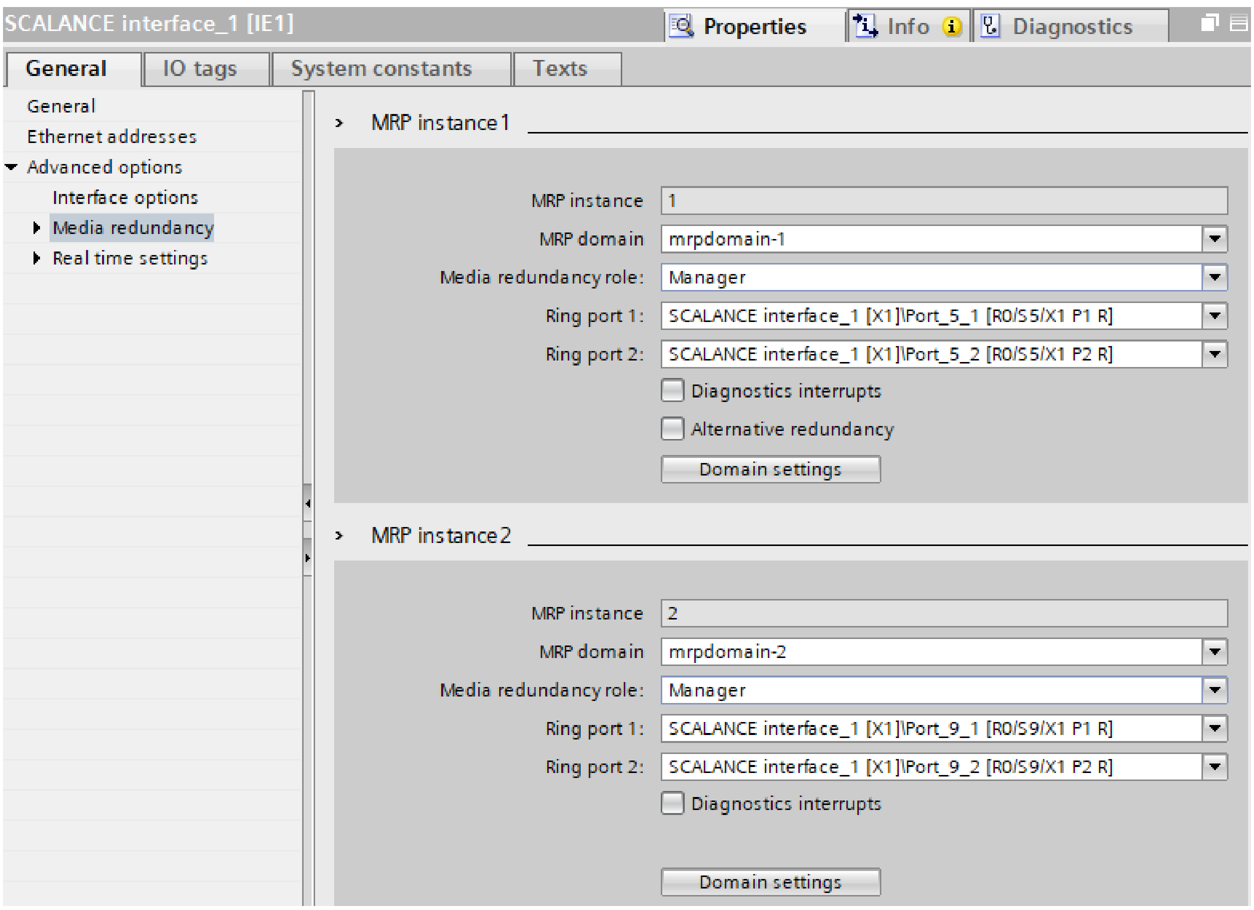


Figure 6-48 Setting media redundancy for multiple rings

The following figure shows Ring 1 (mrpdomain-1). The participants of the mrpdomain-1 are the PROFINET interface of the CPU as the "Client" and the MRP interface 1 of the PROFINET interface of the switch as the "Manager".

Ring interconnection

	Ring	MRP domain
1	Ring1	mrpdomain-1
2	Ring2	mrpdomain-2

	PROFINET device name	MRP instance	MRP domain	MRP role	Ring port 1	Ring port 2
1	PLC_1.PROFINET-IF_1		mrpdomain-1	Client	Port_1 [X1 P1 R]	Port_2 [X1 P2 R]
2	Switch_1.SCALANCE-IF_1	1	mrpdomain-1	Manager	Port_1 [R0/S5/X1 P1 R]	Port_2 [R0/S5/X1 P2 R]

Figure 6-49 Ring 1

The following figure shows Ring 2 (mrpdomain-2). The participants of the mrpdomain-2 are the PROFINET interface of the IO device as the "Client" and the MRP instance 2 of the PROFINET interface of the switch as the "Manager".

Ring interconnection

	Ring	MRP domain
1	Ring1	mrpdomain-1
2	Ring2	mrpdomain-2

	PROFINET device name	MRP instance	MRP domain	MRP role	Ring port 1	Ring port 2
1	IO-Device_1.PROFINET-IF		mrpdomain-2	Client	Port_1 [X1 P1 R]	Port_2 [X1 P2 R]
2	Switch_1.SCALANCE-IF_1	2	mrpdomain-2	Manager	Port_5 [R0/S9/X1 P1 R]	Port_6 [R0/S9/X1 P2 R]

Figure 6-50 Ring 2

6.4.5 MRP interconnection

Definition

The process MRP interconnection is an enhancement of MRP and allows the redundant coupling of two or more rings with MRP in PROFINET networks. MRP interconnection is - like MRP - specified in the standard IEC 62439-2 (Edition 3).

Advantages

MRP interconnection allows the monitoring of larger topologies with ring redundancy. With MRP interconnection, you are not limited to the maximum number of 50 devices in a ring when setting up redundant network topologies. For information on the quantity structure of MRP Interconnection, refer to the SCALANCE XM-400/XR-500 Web Based Management (WBM) (https://support.industry.siemens.com/cs/ww/en/view/109760840) configuration manual.

Requirements

- The media redundancy procedure MRP is used in the participating rings.
- Each ring has its own MRP domain with an MRP manager and MRP clients.
- As MRP managers in the rings, the PROFINET devices support MRP interconnection (see technical specifications of the devices).
  S7-1500 CPUs as of firmware version V2.9 support MRP interconnection.
- If you use PROFINET devices with more than 2 ports as MRP clients in the ring, then MRP interconnection is binding for these devices. For a device without MRP interconnection, telegrams leave the ring. This results in an additional load on the network.
- If you use PROFINET devices with only 2 ports in the ring as MRP clients, then MRP interconnection is recommended for all devices in the ring.

4 devices for MRP interconnection connections

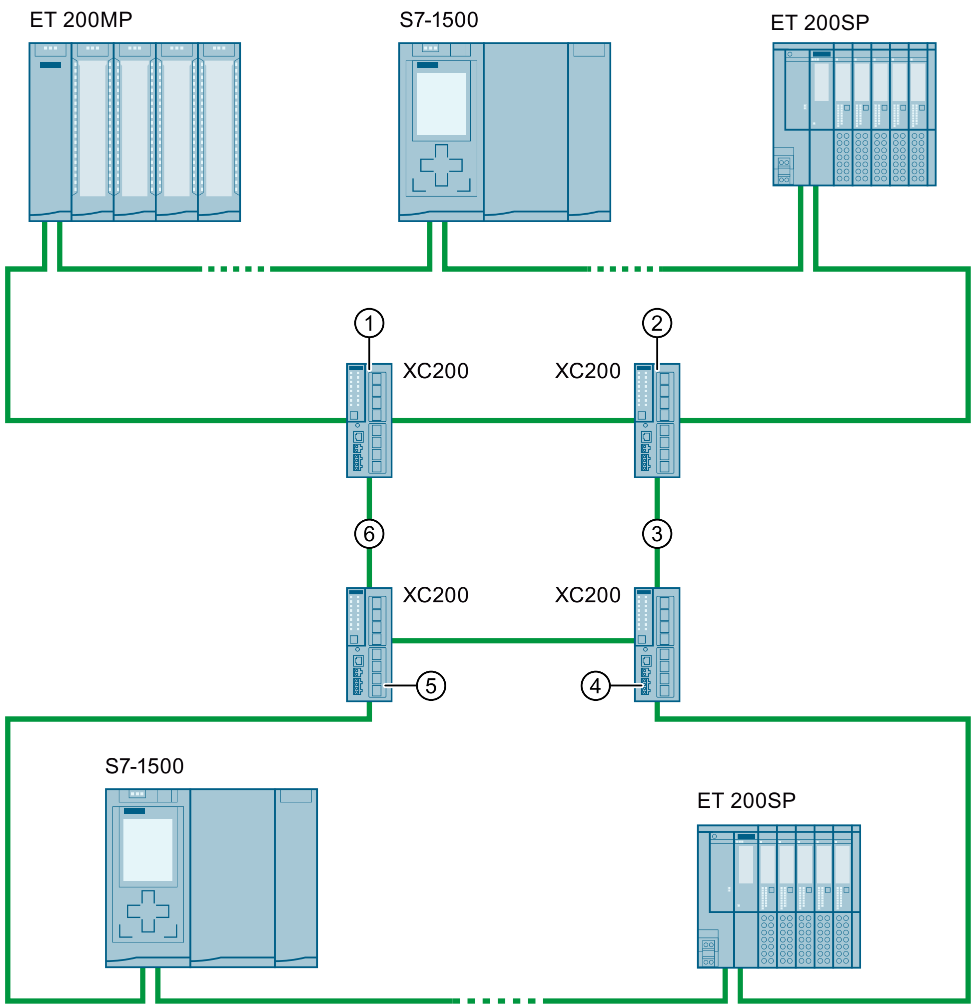
2 MRP-interconnection connections provide redundant coupling between 2 MRP rings. Four devices are required for the 2 connections:

- A Media Redundancy Interconnection Manager (MIM)
- 3 media-redundancy interconnection clients (MIC):
  – Primary MIC
  – Primary Coupled MIC
  – Secondary Coupled MIC

Since each of the four devices is part of an MRP ring, each device also assumes a media redundancy role.
Each of the 4 devices can take over the MRP manager role or MRP client role.

Topology

The following screen shows the redundant coupling of 2 MRP rings.
Redundancy: If the primary connection (primary link) between the two rings is interrupted, the secondary connection (secondary link) takes over the data exchange between the rings after a short reconfiguration time.



① Primary MIC
② Media Redundancy Interconnection Manager (MIM)
③ Secondary link
④ Secondary Coupled MIC
⑤ Primary Coupled MIC
⑥ Primary link

Figure 6-51 Example: Redundant connection of 2 rings with MRP interconnection

Function of the interconnection ports

Interconnection ports are appropriately configured PROFINET ports of the 4 devices for MRP interconnection. The primary or secondary connection of the rings exists between the interconnection ports.

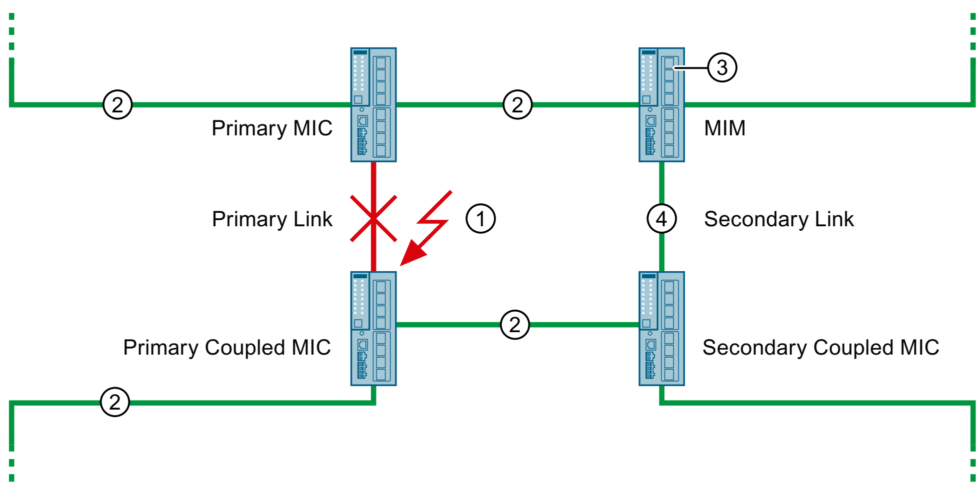
Via the interconnection ports; the MIM is always informed about the connection status between the Primary MIC and the Primary Coupled MIC ("primary connection") as well as its own connection to the Secondary Coupled MIC ("secondary connection").

Depending on the connection status of the interconnection ports, each MIC sends a message to the MIM when its status changes (Link-Up or Link-Down).

In regular operation the data between the two rings is exchanged over the primary connection (primary link) and the MIM blocks its interconnection port.

Link-Down of the primary connection

In the figure below, the primary connection was interrupted (Link-Down).



① Link-Down

② Status message: Link-Down primary connection

③ Blocking of the interconnection port is released

④ Data exchange via secondary connection

Figure 6-52 Link-Down of the primary connection

With a Link-Down of the primary connection ①, Primary MIC and Primary Coupled MIC send the status message "Link-Down" via all lines ②.

The MIM then unblocks its interconnection port at ③ and informs the MRP managers in both rings to trigger the reconfiguration of the rings.

After the reconfiguration, the data exchange between the two rings takes place via the secondary connection ④.

As soon as Primary MIC and Primary Coupled "Link-Up" for the primary connection, the MIM blocks its interconnection port and informs the MRP managers in both rings to trigger the reconfiguration of the rings.

After the reconfiguration, the data exchange is again carried out via the primary connection.

Configuring and assigning MRP interconnection parameters

NOTE

MRP interconnection is not integrated into the TIA Portal. This means you cannot configure, parameterize and diagnose MRP interconnection in STEP 7.

You configure and assign MRP interconnection parameters via the integrated web pages (Web Based Management) of the SCALANCE switches used as MIM and MIC (e.g. SCALANCE XC200, XM-400, XR-500).

For the configuration, follow the step-by-step instruction in the configuration manual SCALANCE XM-400/XR-500 Web Based Management (WBM) (https://support.industry.siemens.com/cs/ww/en/view/109760840).

Reconfiguration and watchdog timer

The reconfiguration time of the rings is typically less than 200 milliseconds.

In order for the data exchange to work after a reconfiguration, the watchdog timers of the IO devices must be set sufficiently large. Therefore, when using MRP interconnection, set the watchdog timer of the IO devices to 256 milliseconds or more.

6.5 Real-time communication

6.5.1 Introduction

Properties

PROFINET IO is a scalable real-time communication system based on the Layer 2 protocol for Fast Ethernet. With the RT transmission method, two real-time-support performance levels are available for time-critical process data and IRT for high-accuracy and also isochronous processes .

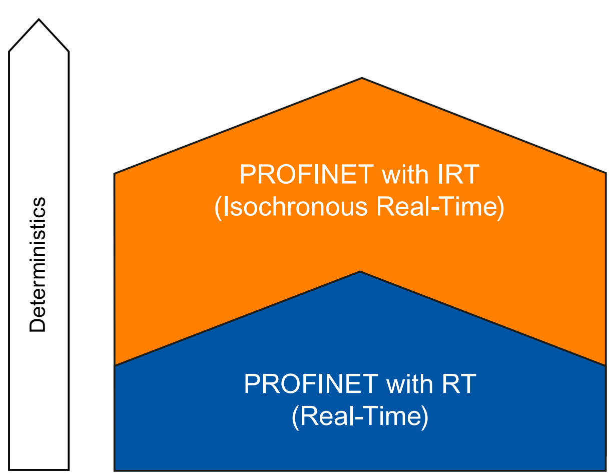


Figure 6-53 Real-time communication performance

Real-time communication and IT communication

Industrial communication, especially in the factory automation and process automation, requires a punctual and deterministic data transmission. This is why, for the cyclic exchange of time-critical IO user data, PROFINET IO does not use TCP/IP, it uses real-time communication (RT) or isochronous real-time communication (IRT) for the synchronized data exchange in reserved time intervals.

Fields of application of PROFINET with RT

PROFINET with RT is suited for:

- Time-critical applications in factory automation
- The implementation of large quantity structures in line process plants

Fields of application of PROFINET with IRT

PROFINET with IRT is particularly suitable for:

- Considerable deterministics with large quantity structures concerning the I/O user data communication (productive data)
- Considerable performance also with many devices concerning the I/O user data communication (productive data)
- Parallel transfer of productive and TCP/IP data via a cable, even with considerable data traffic whilst ensuring the forwarding of productive data by reserving the transmission bandwidth.

International standard IEC 61158

The communication procedures RT and IRT are standardized in the international standard IEC 61158.

6.5.2 RT

PROFINET IO with real-time communication (RT) is the ideal solution for integrating IO systems. This is a solution that also uses standard Ethernet in the devices and commercially available industrial switches as infrastructure components. A special hardware support is not required.

If you want to use PROFINET functions that provide an additional value, such as topology detection, diagnostics, device exchange without exchangeable medium/PG, you have to use switches that support the PROFINET standard according to the IEC 61158 standard. In the integrated switches of PROFINET devices and PROFINET switches (e.g., the SCALANCE product family), PROFINET functions in accordance with the PROFINET standard are implemented and can be used without restriction for integration in the PROFINET IO system (see also section Active network components (Page 37).

Real-time communication (RT)

PROFINET IO frames have priority over standard frames in accordance with IEEE802.1Q. This ensures the required determinism in the automation technology. The data is transferred via prioritized Ethernet frames. With RT, you can achieve update times ≥ 250 µs with RT.

Switching mechanisms

Switches in SIMATIC meet the real-time requirements with two mechanisms on PROFINET: "Store and forward" and "Cut through".

Store and forward

In the Store and forward method, the switch stores the frames completely and places them in a queue. If the switch supports the international standard IEEE 802.1Q, then the data is sorted according to its priority in the queue. The frames are then forwarded selectively to the specific port that can access the addressed node (Store and forward).

Cut through

In the Cut through process the entire data package is not stored temporarily in a buffer, it is passed directly onto the target port as soon as the target address and the target port have been determined.

The times required by the data package to pass through the switch are then minimal and independent of the frame length. The data are only stored temporarily in accordance with the Store and forward process as per priority when the section between the target segment and the port of the next switch is occupied.

6.5.3 IRT

Isochronous real-time communication (IRT)

IRT is a synchronized communication protocol for cyclic exchange of IRT data between PROFINET devices. A reserved bandwidth is available in the send cycle for IRT data. The reserved bandwidth ensures that the IRT data can be transferred at reserved synchronized intervals, without being influenced also by higher other network loads (such as TCP/IP communication, or additional real-time communication).

Advantages

PROFINET with IRT is synchronized communication in reserved time intervals. With IRT you have the option to be able to control time-critical applications such as motion control via PROFINET. With IRT you use further advantages:

- High-precision deterministics allow highest control quality and thus, for example, exact positioning of axes
- Simple and flexible integration of PROFINET devices for real-time applications in existing company networks
- Shortest response times and highest deterministics by means of bandwidth reservation and thus also for applications that have to satisfy highest performance requirements (e.g., printing control/position control for presses, print-mark recognition with packaging machines)
- Secured standard communication parallel to real-time communication via the same transmission medium
- You can continue to use standard components for your PROFINET IO system outside of the sync domain

Properties of isochronous real-time

Topology configuration is a prerequisite for IRT.

In addition to the reserved bandwidth, the frames from defined transmission paths are exchanged for the further optimization of data transfers. For this, the topological information from the configuration is used for planning the communication. Transmission and reception points of every individual data frame at every communication node are thus guaranteed. This allows you to achieve optimal usage of the bandwidth and reach the best possible performance in the PROFINET IO system.

Use of IRT allows you to achieve update times with very high deterministics ≥ 125 µs and a jitter accuracy of the send clock < 1 µs. Isochronous applications are possible with IRT (see section Isochronous mode (Page 219)).

* The minimum update time depends on the CPU used. The update times supported by a CPU are listed in the technical specifications of the CPU manual.

NOTE

IWLAN and IRT

PROFINET devices that are connected to PROFINET IO via access points, do not support IRT.

Synchronization

The prerequisite for the IRTcommunication is a synchronization cycle for all PROFINET devices in a sync domain for distributing a mutual time basis. A synchronism of the transmission cycle of the PROFINET devices within a sync domain is achieved with this basic synchronization.

A sync master specifies the clock to which sync slaves are synchronized. An IO controller or an IO device can have the role of sync master. If the sync master fails, all IO devices with IRT will fail.

Sync master and sync slaves are always devices in a sync domain. Bandwidth is reserved within the sync domain for IRT communication. Real-time and non-real-time communication (TCP/IP communication) is possible outside of the reserved bandwidth.

Default sync domain

If you have created a PROFINET subnet, a special sync domain is automatically created, the default sync domain. All PROFINET devices configured for the PROFINET subnet automatically belong to this sync domain.

For additional information on creating sync domains, refer to the STEP 7 online help.

Time ranges of the communication cycle

The communication cycle is divided into three time ranges, which are shown in the following chart:

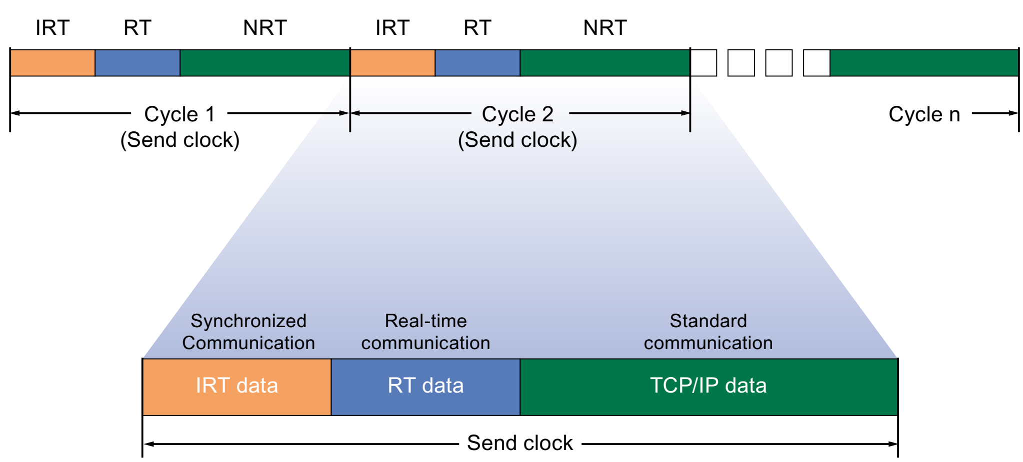


Figure 6-54 Reserving bandwidth

- IRT data (synchronized communication)
  This time range can be reserved in specific steps, depending on the send clock. Only IRT data is transmitted during this time range.
- RT data (real-time communication)
  The cyclic RT data is transferred in this time range. RT data is prioritized over "normal" TCP/IP data. TCP/IP data or Ethernet frames can have a priority of between 1 and 7. RT data have the Priority 6.
- TCP/IP data (standard communication)
  Standard communication (e.g., TCP/IP) is transmitted in the remaining interval of the communication cycle.

In STEP 7, you can set the ratio of the bandwidth usage of cyclic IO data to standard communication. For information on how to set the bandwidth usage, refer to the section Setting the bandwidth usage for the send clock (Page 204)

6.5.4 Comparison of RT and IRT

The most important differences between RT and IRT

Table 6-2 Comparison between RT and IRT

Property	RT	IRT
Transmission method	Prioritizing the RT frames through Ethernet priority (VLAN tag)	Path-based switching on the basis of a communication path plan; no transmission of TCP/IP frames in the time range with IRT communication.
Determinism	Variance of the duration of transmission by the shared use of the transmission bandwidth with other protocols (e.g., TCP/IP)	Exact, planned transmission, transmission and reception times are guaranteed for all topologies.
Hardware support through special Ethernet controllers is required	Not required	Required
Isochronous application	-	Yes
Starting time of the isochronous application	-	The point in time for the reception of the data is planned exactly. Application can be started synchronized to the cycle.

6.5.5 Configuring PROFINET IO with IRT

Introduction

If you want to configure a PROFINET IO system with IRT, you have to configure the PROFINET IO devices. These PROFINET devices must support IRT. Set which device acts as the sync master and synchronizes the other devices. This is done by configuring a sync domain with a sync master and at least one sync slave.

Requirement

- There is an IO system with an IO controller and at least one IO device.
- You have configured the topology for the IO system.
- The devices support IRT.

Procedure

Proceed as follows to enable an existing IO system for the use of IRT:

1. Select the PROFINET interface of the IO controller.
2. In the Inspector window, navigate to "Advanced options > Real time settings > Synchronization".

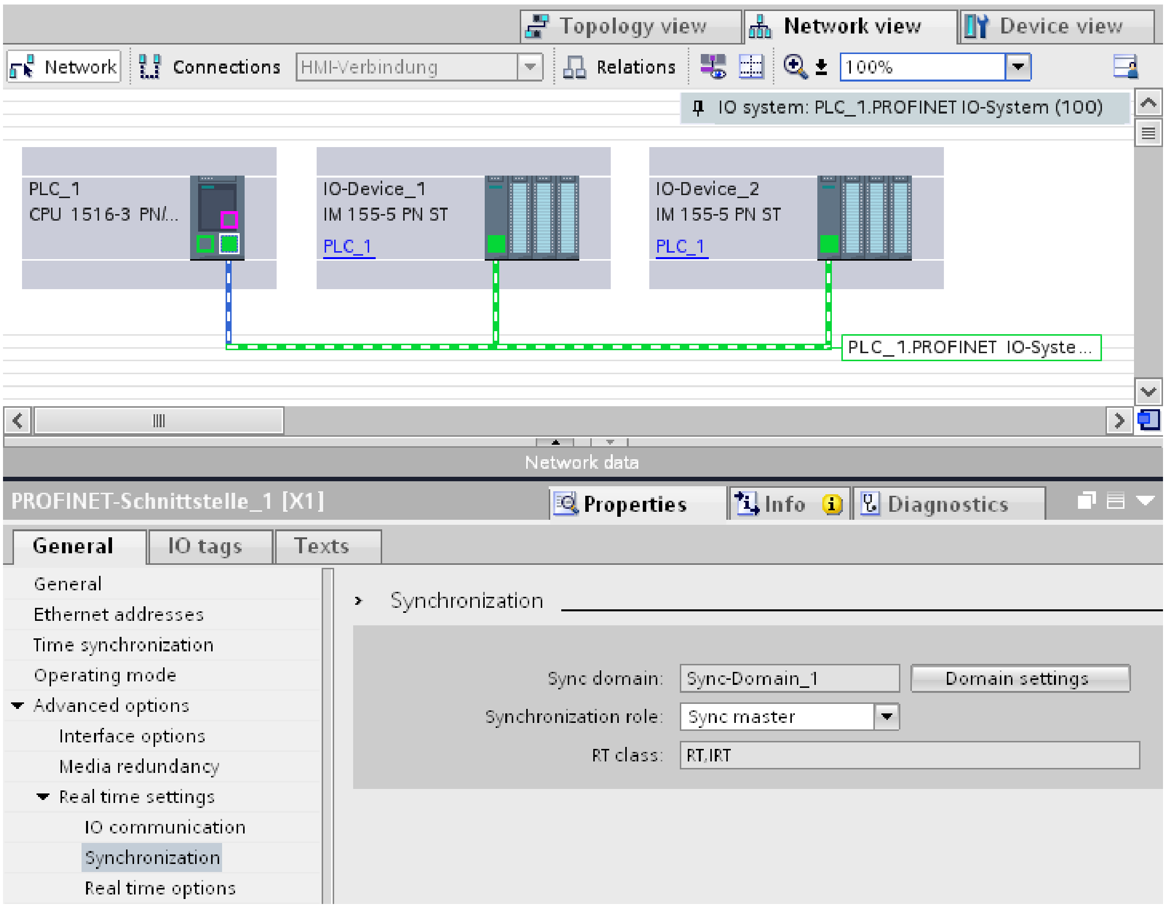


Figure 6-55 IRT configuring sync master

3. Assign the IO controller the role of sync master under "Synchronization role".
4. Select the PROFINET interfaces of an associated IO device.

5. In the Inspector window, navigate to "Advanced options > Real time settings > Synchronization".

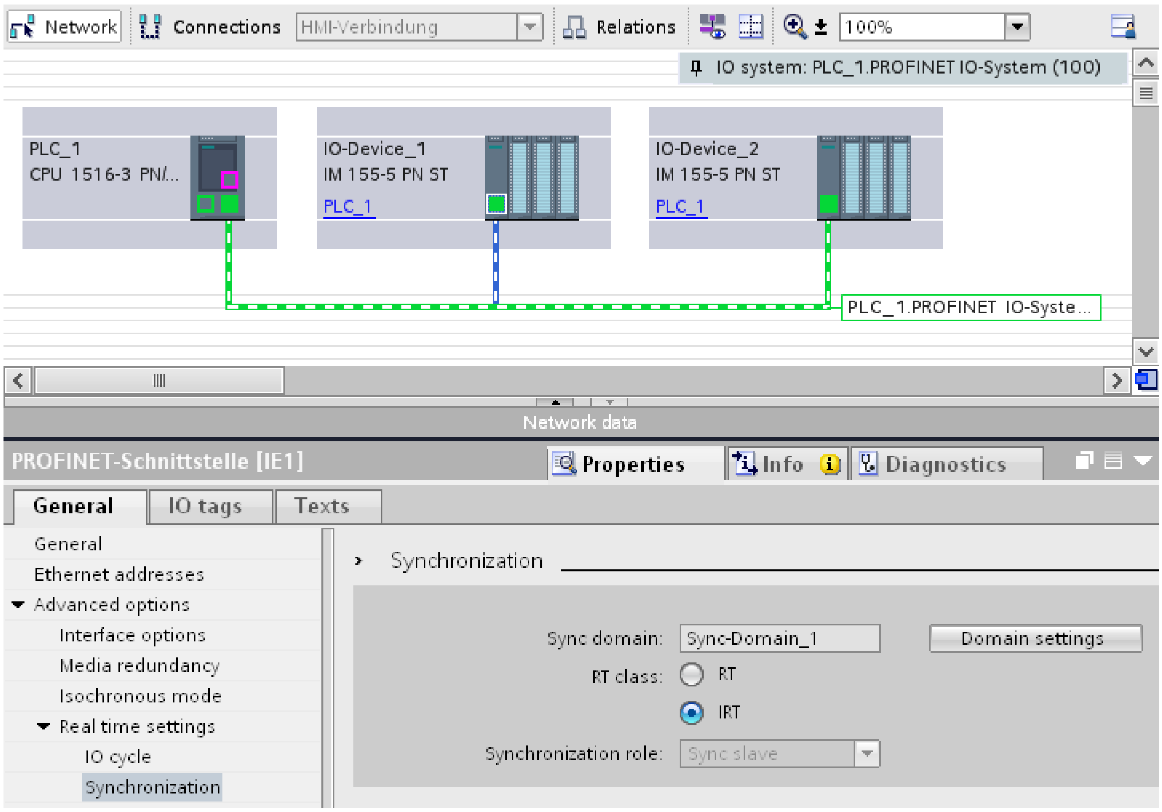


Figure 6-56 IRT configuring sync slave

6. Activate the RT class "IRT". The IO device will then automatically be assigned the synchronization role "sync slave".
7. You can check and correct your settings at any time using the "Domain settings" button.

Or

1. Highlight the PROFINET IO system in the network view.
2. Click on the PROFINET IO system.

3. Navigate to the device of the required sync domain in the Inspector window.

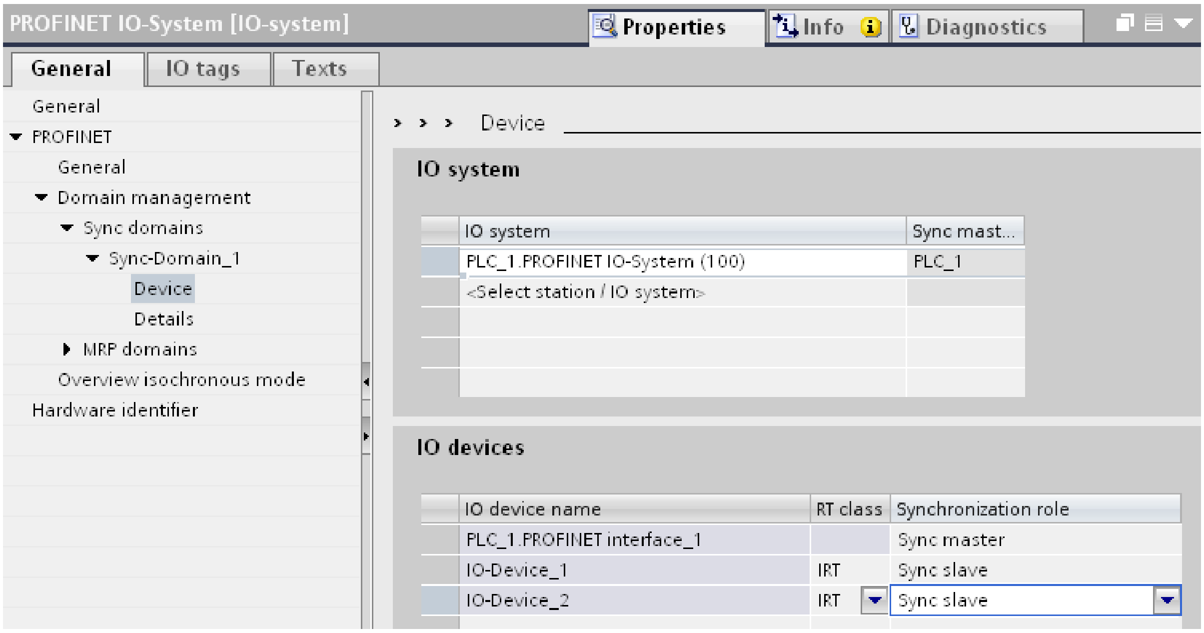


Figure 6-57 IRT configuring sync domain

4. Enter all necessary settings in the tables:
   – Select an IO system.
   – Set the synchronization role "sync master" for the IO controller.
   – Set the RT class for the IO devices to "IRT". The IO devices will then automatically be assigned the synchronization role "sync slave".

You can now load the configuration to the relevant devices with PROFINET IRT.

6.5.6 Setting the bandwidth usage for the send clock

Bandwidth level

For PROFINET IO with IRT, you can specify the maximum portion of the send cycle that you want to use for cyclic IO communication.

You have the following setting options for the bandwidth usage in STEP 7.

- Maximum 25% cyclic IO data. Focus on non-cyclic data.
- Maximum 37.5% cyclic IO data. Focus on non-cyclic data.
- Maximum 50% cyclic IO data. Balanced proportion.
- Maximum 90% cyclic IO data. Focus on cyclic IO data. (Requirement: The "Make 'high performance' possible" option is activated)

Setting the bandwidth usage

To set the bandwidth usage for a PROFINET IO system in STEP 7, follow these steps:

1. Select your IO system in the network view of STEP 7.
2. In the Inspector window, go to "Properties" > "General" > "PROFINET" > "Domain management" > "Sync domains" > "Name of the sync domain" > "Details".
3. Select the desired level of bandwidth usage from the "Bandwidth use" drop-drop down list box.

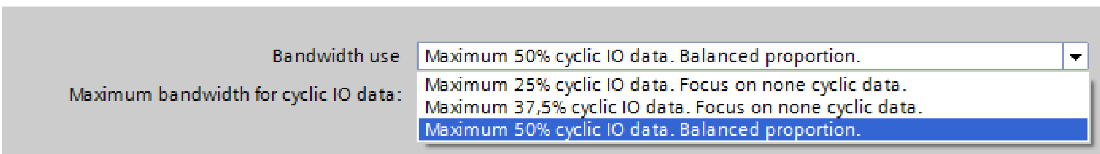


Figure 6-58 Setting the bandwidth usage

For information on how to set the bandwidth usage for IRT with maximum performance, refer to the section Configuration of IRT with high performance (Page 215)

6.5.7 Setup recommendations for optimizing PROFINET

Optimizing PROFINET with RT

PROFINET provides you with high-performance communication on all levels.
The following figures shows an example of an optimized PROFINET topology.

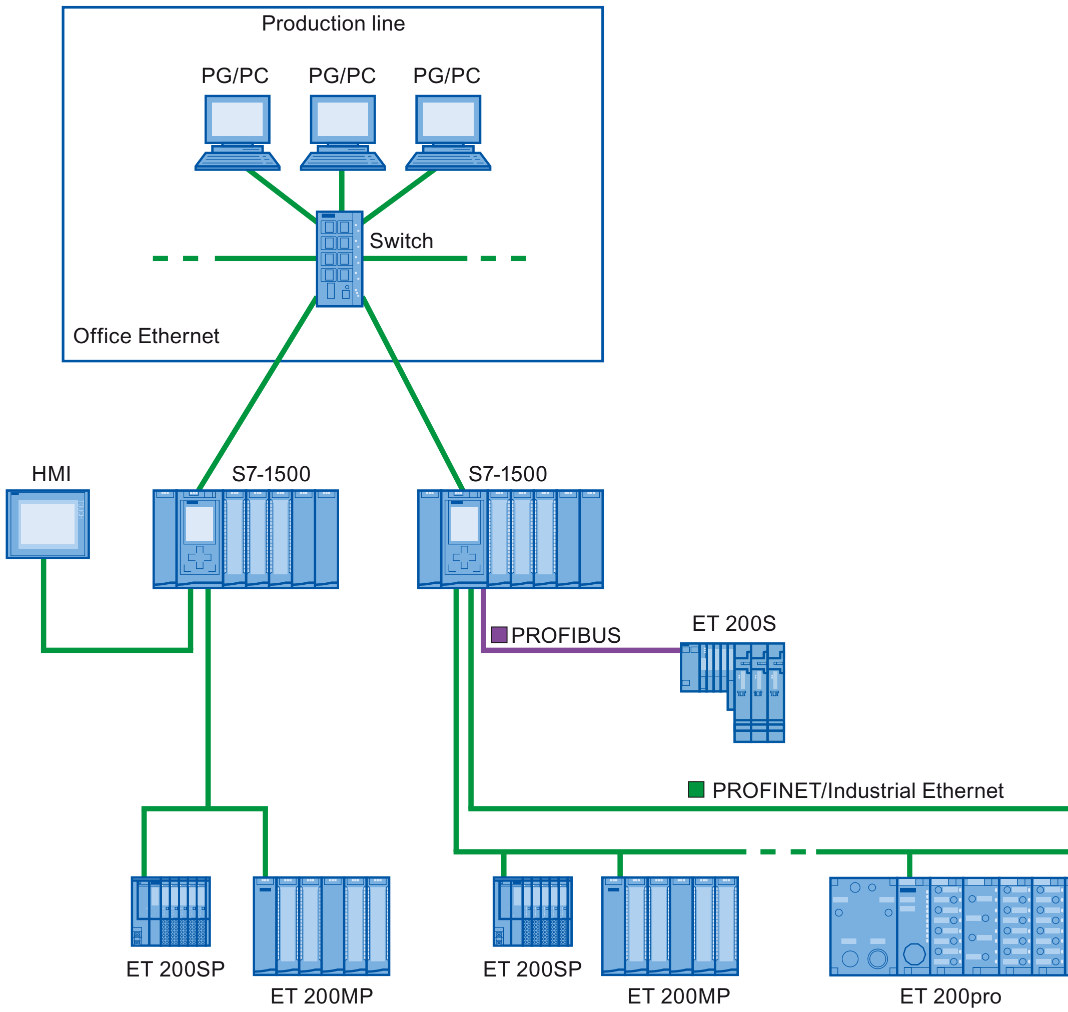


Figure 6-59 Optimized PROFINET topology

When setting up your PN network topology, take care that the various automation applications are distributed among separate network branches so that a sufficient bandwidth reserve will be available for future expansions.

- If you integrate standard Ethernet devices into the network topology or use standard Ethernet communication, take into account the network load caused by standard Ethernet and adapt the network topology as appropriate (max. bandwidth 100 Mbps).
- For communication with higher-level networks with a high data volume, use the most direct paths possible to the higher-level network infrastructure.

Also observe the Installation Guideline (http://www.profibus.com/nc/download/installation-guide/downloads/profinet-installation-guide/display/) of the PROFIBUS User Organization.

Setting up PROFINET with IRT

Keep in mind the following rules for setting up and operating a PROFINET IO system in IRT mode. These rules will ensure best possible operation of your PROFINET IO system.

- When using IRT, you must configure the topology. This will enable exact calculation of the update time, bandwidth, and other parameters.
- If you would like to use multiple sync domains, configure a sync boundary for the port which is currently connected to the PROFINET device of another sync domain.
- In a sync domain, you can only configure one sync master at a time.
- A PROFINET IO system may only be part of a single sync domain.
- If you configure PROFINET devices in a sync domain and want to synchronize with IRT, the PROFINET devices concerned must support IRT communication.
- If possible, use the same PROFINET device as the PROFINET IO controller and sync master.
- If only some of the PROFINET devices in a PROFINET IO system are synchronized, please keep in mind the following: Assign PROFINET devices that are not participating in the IRT communication to the RT class "RT" and the synchronization role "unsynchronized" in the sync domain.

Applications for CPU with multiple PROFINET IO interfaces

- Connecting machines: Your configuration contains machines located on separate IO lines. You can perform real-time communication between the CPUs over PROFINET IO interface X2. Use the I-device or shared I-device function for this purpose.
  The figure below shows an example configuration for two machines connected over the X2 interface with an I-device relationship.

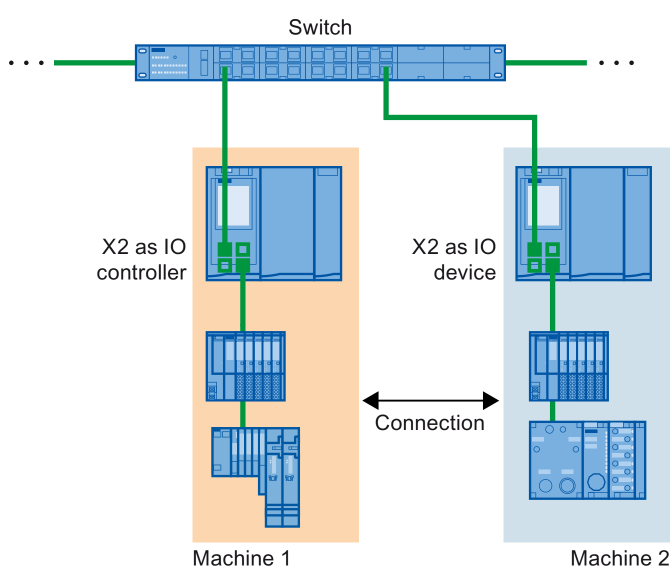


Figure 6-60 I-device connection over 2nd PNIO interface

- Distribution by automation tasks:
  – For automation tasks with high performance and deterministic requirements, use PROFINET with IRT over the X1 interface.
  – For other tasks that you can implement with RT, use the X2 interface.

If you use interface X2 as the PROFINET IO interface for one of the following CPUs, this can affect performance:

- CPU 1515(F)-2 PN
- CPU 1515T(F)-2 PN
- CPU 1516(F)-3 PN/DP
- CPU 1516T(F)-3 PN/DP
- CPU 1516(F)pro-2 PN

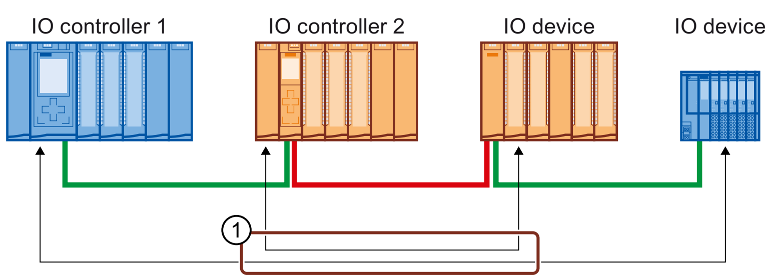
You can find additional information in the section PROFINET interface (Page 30) and in the Cycle and Response Times (https://support.industry.siemens.com/cs/ww/en/view/59193558) function manual.

Topological overlap of IO systems in multi-controller applications

In a configuration with multiple IO controllers, shared paths are subject to the combined network loads of all connected PROFINET IO systems.

To avoid high communication loads in multi-controller applications, observe the following recommendations:

- Avoid paths that are shared by multiple IO systems.
  The figure below shows a configuration with two PROFINET IO systems that use the same paths.



① Shared path

Figure 6-61 Multi-controller application with shared path

In the figure below, the two PROFINET IO systems do not use shared paths.

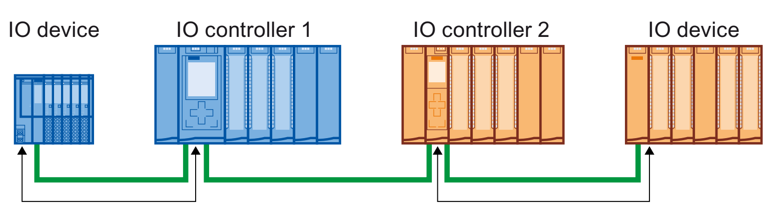


Figure 6-62 Multi-controller application with separate paths

- If separation is not possible: Increase the update time for the affected IO devices.

Installation guidelines of the PROFIBUS user organization

The installation guidelines can be found on the Internet (http://www.profibus.com/nc/download/installation-guide/downloads/profinet-installation-guide/display/).

6.5.8 Limitation of the data infeed into the network

Limiting the data infeed into the network on PROFINET interfaces

The "Limit data infeed into the network" function limits the network load of standard Ethernet communication which is fed into the network by the interface to a maximum value. This does not apply to cyclic real-time communication (RT/IRT).

In a PROFINET IO system, critical network loads can occur quickly with standard Ethernet communication, especially in line topologies. Devices which feed a lot of standard Ethernet communication into the PROFINET IO system, should support the function "Limit data infeed into the network". If necessary, you should optimize your topology by avoiding e.g. standard Ethernet devices in line topologies. You can find additional information on this in section Setup recommendations for optimizing PROFINET (Page 206).

Depending on the performance capability of the PROFINET interface, you can set in its properties whether or not the data infeed into the network should be limited. Requirements for adjustability:

- The interface offers sufficient performance.
- No IO data transfer is configured

If the conditions are not met, the data infeed is always limited.

Depending on the interface, you can activate or deactivate the "Limit data infeed into the network" function. If you use the X1 interface of an S7-1500 CPU as the IO controller or I-device, the "Limit data infeed into the network" function is always enabled. Example: If you are not using the X1 interface of an S7-1500 CPU as IO controller or I-device, you can enable or disable the function.

Uses of the limitation of the data infeed to the network

- Division of the bandwidth for standard Ethernet communication between devices:
  In PROFINET networks cyclic real-time communication and standard Ethernet communication share the same network. This means that only a limited bandwidth remains for standard Ethernet communication. The limitation of the data infeed ensures that the remaining bandwidth for standard Ethernet communication is not used just by one device, but is divided between several devices.
- Smoothing peaks in the data infeed:
  The limitation of the data infeed smooths peak loads of standard Ethernet communication (for example, from Open User Communication, access by the Web server).
- Prevent excessive data at the source:
  If applications in a device generate too much data, this data is not forwarded to the PROFINET network. Negative effects (for example data loss, communication interruption) remain limited to the device feeding in the data and its communications partner. Other nodes are not disturbed.

Setting limitation of the data infeed into the network for a CPU

To set the limitation of the data infeed into the network, follow these steps:

1. In the network view of STEP 7, select the interface of the CPU.
2. In the Inspector window, go to "Properties" > "General" > "Advanced options" > "Interface options".
3. Select or clear the "Limit data infeed into the network" check box.

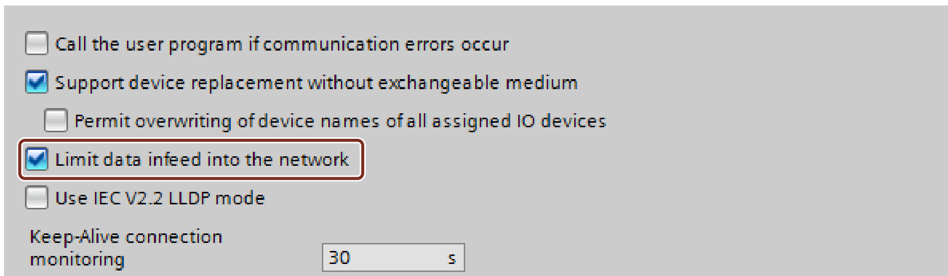


Figure 6-63 Limit data infeed into the network

6.6 PROFINET with performance upgrade

Performance upgrade

The performance upgrade implements the application class "High Performance" of the PROFINET specification V2.3.

The performance upgrade provides a series of measures that lead to the following improvements for PROFINET with IRT:

- Reduction of runtime delays in the IO devices
- Increase in the bandwidth for cyclic IO data
- Reduction of the bandwidth used for PROFINET frames
- Reduction of the send clocks

The improvements achieved with the performance upgrade mean that you can operate your PROFINET IO system with more devices with the same send clock or the same number of devices with a shorter send clock.

With PROFINET, it was previously possible to reach a send clock of 250 μs.

With the performance upgrade, it is now possible to achieve isochronous send clocks of up to 125 μs with the fast forwarding, dynamic frame packing and fragmentation procedures, depending on the performance capability of the CPU (see Technical Specifications in the device manuals). With short send clocks, standard communication remains possible.

The performance upgrade will help you to implement applications with high speed and send clock requirements.

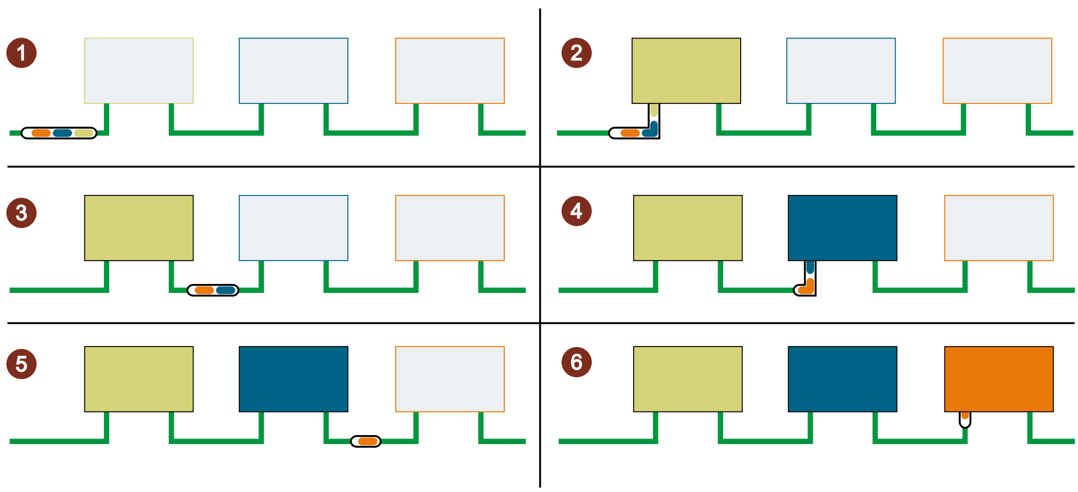
6.6.1 Dynamic frame packing

Dynamic frame packing

Previously, individual PROFINET IO frames were sent for every PROFINET IO device.

The performance upgrade uses the dynamic frame packing procedure that is based on the summation frame method. With the summation frame method, a frame contains the user data for neighboring devices on a line. With Dynamic Frame Packing, every IO device takes its data from the frame and forwards the rest. The frame is shortened from IO device to IO device. Dynamic frame packing improves the use of the bandwidth in a line topology.

The following figure shows how Dynamic Frame Packing works based on the example of a frame containing the user data for 3 IO devices.



① The PROFINET IO frame contains the user data for all 3 IO devices (green, blue and orange).

② The PROFINET IO frame reaches the first IO device. The IO device takes its user data (green) from the frame and forwards the remaining frame.

③ The PROFINET IO frame contains the user data for two IO devices (blue and orange).

④ The PROFINET IO frame reaches the second IO device. The IO device takes its user data (blue) from the frame and forwards the remaining frame.

⑤ The PROFINET IO frame contains the user data for one IO device (orange).

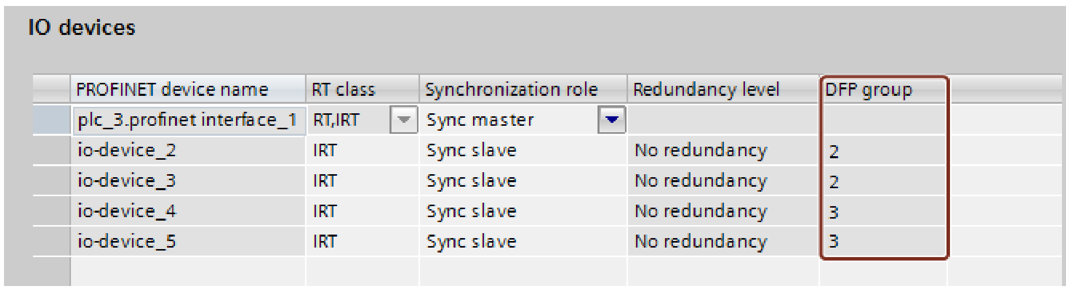
⑥ The PROFINET IO frame reaches the last IO device. The IO device saves the entire frame including user data (orange).

Figure 6-64 Dynamic Frame Packing (DFP)

DFP groups

Dynamic frame packing automatically groups IO devices that support the performance upgrade into DFP groups. To be grouped together in a DFP group. the IO devices must be located one after the other in a line and must have the same update time and watchdog time. As soon as a maximum frame size for the DFP group is exceeded or a maximum number of members for a DFP group is reached, Dynamic Frame Packing automatically opens a new DFP group.

STEP 7 shows the DFP groups in "Domain management" > "Sync domains" > "Name of the sync domain" > "Device" in the "IO devices" box.


IO devices

PROFINET device name	RT class	Synchronization role	Redundancy level	DFP group
plc_3.profinet interface_1	RT,IRT	Sync master		
io-device_2	IRT	Sync slave	No redundancy	2
io-device_3	IRT	Sync slave	No redundancy	2
io-device_4	IRT	Sync slave	No redundancy	3
io-device_5	IRT	Sync slave	No redundancy	3

Figure 6-65 Display of the DFP groups in STEP 7

NOTE

Dynamic frame packing with high watchdog times

If you set the "Accepted update cycles without IO data" parameter higher than 31 for an IO device, Dynamic Frame Packing does not place this IO device in a DFP group.

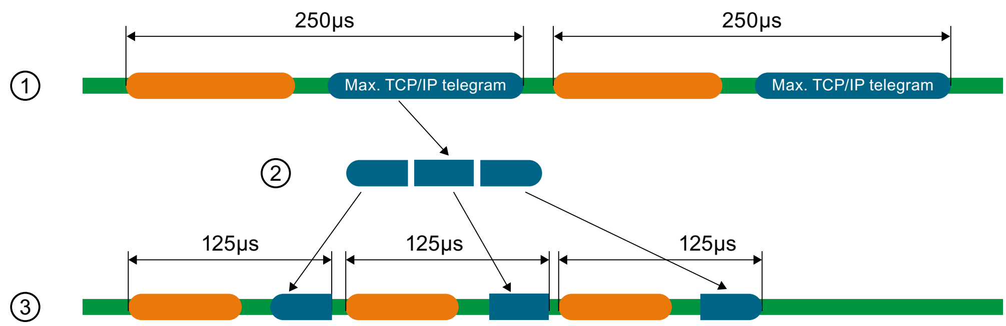
6.6.2 Fragmentation

The transfer of a complete standard Ethernet frame with TCP/IP data takes up to 125 µs. This means that the cycle time for PROFINET IO data cannot be reduced by any desired amount.

The performance upgrade uses the fragmentation procedure, which breaks down TCP/IP frames into sub-frames. These frame segments are transferred to the target device over multiple send clocks; there they are reassembled to the original TCP/IP frame.

Fragmentation is a requirement for cycle times lower than 250 µs. In these short send clocks, you can use more bandwidth for cyclic IO data, since the fragments of the standard Ethernet frames use significantly less bandwidth than a complete standard Ethernet frame.

The following figure shows how fragmentation works.



① A standard Ethernet frame with TCP/IP data is up to 125 µs.

② During fragmentation, the standard Ethernet frame is divided into frame segments.

③ The frame segments are divided into multiple short send clocks.

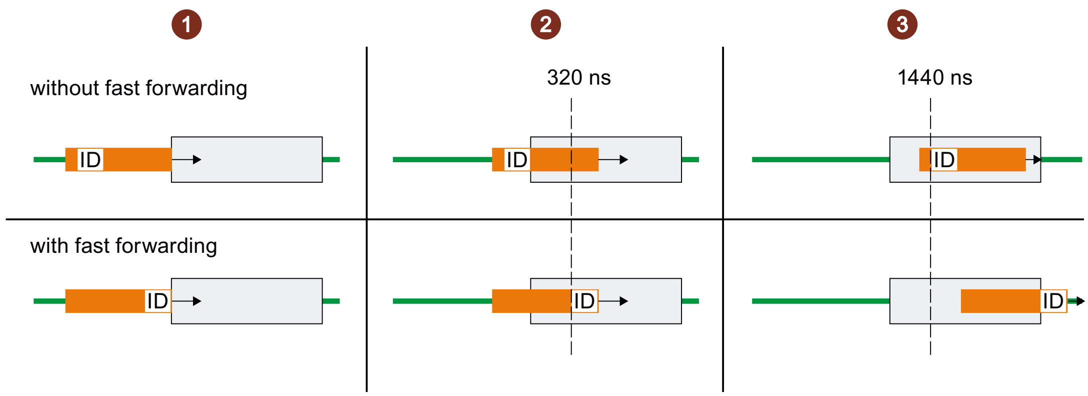
Figure 6-66 Fragmentation

NOTE

The PROFINET IO interface of the IO controller supports fragmentation if all the ports except one are deactivated.

6.6.3 Fast forwarding

To be able to decide whether a frame should be forwarded or used, a PROFINET IO device requires the frame ID. It previously took 1440 ns until the frame ID was present in the IO device.
The performance upgrade uses the fast forwarding procedure in which the frame ID is located nearer the front of the frame. The throughput time in the device is reduced to 320 ns.
With fast forwarding, the throughput time of the frame is reduced in your PROFINET IO system. This results in decisive performance advantages, particularly in line, ring and tree topologies.
The figure below compares the throughput of a PROFINET IO frame in an IO device with and without fast forwarding.



① Start:
- Both PROFINET frames reach the IO device.
  The IO devices start to check the frames for the frame ID

② After 320 ns:
- Without fast forwarding: The IO device is still checking the PROFINET frame for the frame ID.
- With fast forwarding: The IO device receives the frame ID from the PROFINET frame and forwards the frame.

③ After 1440 ns:
- Without fast forwarding: The IO device receives the frame ID from the PROFINET frame and forwards the frame.
- The PROFINET frame with fast forwarding is "ahead" of the PROFINET frame without fast forwarding.

Figure 6-67 Fast forwarding

6.6.4 Configuration of IRT with high performance

High-end applications with IO communication require excellent performance in IO processing, for example in the control of wind turbines (converter control).
To use IRT communication with the highest performance in your PROFINET IO system, enable the option "Make 'high performance' possible".
When you enable the "Make 'high performance' possible" option, this has the following effects:

- You can set send clocks of 187.5 µs and 125 µs (for example CPU 1518-4 PN/DP) or 187.5 µs (for example CPU 1517-3 PN/DP), see Technical specifications in the CPU device manuals.
- You can set more bandwidth use for cyclic IO data.
- You can use the option "Allows the use of 'fast forwarding'".

Requirements

- S7-1500 CPU as of firmware version V2.0

Enable the "Make 'high performance' possible" option.

Follow these steps to activate the "Make 'high performance' possible" option:

1. Select the PROFINET IO system in the network view of STEP 7.
2. In the Inspector window, go to "Properties" > "General" > "PROFINET" > "Domain management" > "Sync domains" > "Name of the sync domain".
3. Enable the "Make 'high performance' possible" option.

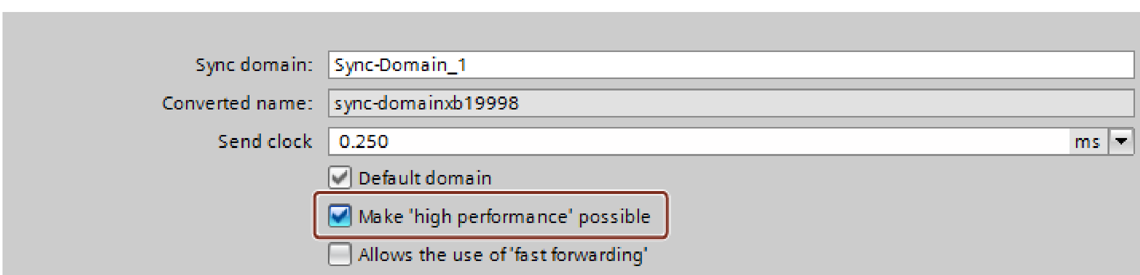


Figure 6-68 Activating high performance

Using more bandwidth for cyclic IO data

Requirement: The "Make 'high performance' possible" option is enabled.

To set more bandwidth for cyclic IO data for your PROFINET IO system, follow these steps:

1. Select your IO system in the network view of STEP 7.
2. In the Inspector window, go to "Properties" > "General" > "PROFINET" > "Domain management" > "Sync domains" > "Name of the sync domain" > "Details".
3. In the drop-down list, select "Maximum 90% cyclic IO data. Focus on cyclic IO data".

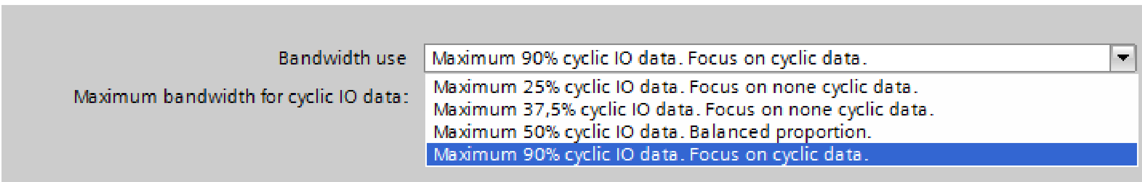


Figure 6-69 Using more bandwidth for cyclic IO data

NOTE

Bandwidth usage in isochronous mode

If you operate your PROFINET IO system in isochronous mode, avoid using the setting for the bandwidth usage "Maximum 90% cyclic IO data. Focus on cyclic IO data."

Setting low send clocks (example CPU 1518-4 PN/DP)

Requirement: The "Make 'high performance' possible" option is enabled.

1. Select the PROFINET IO system in the network view of STEP 7.
2. In the Inspector window, go to "Properties" > "General" > "PROFINET" > "Domain management" > "Sync domains" > "Name of the sync domain".
3. Select the send clock for "Send clock" in the drop-down list.

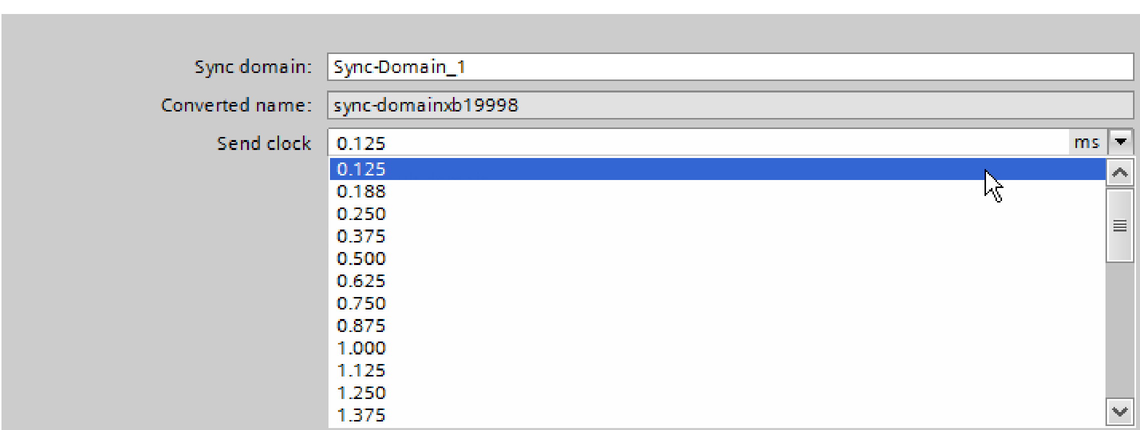


Figure 6-70 Setting low send clocks

Requirements for the fragmentation process (example CPU 1518-4 PN/DP)

If you use the following combinations for send clock and bandwidth settings, the devices in the IO system use the fragmentation process.

- Send clock cycle 125 μs: Always fragmentation irrespective of the bandwidth setting.
- Send clock cycle 187.5 μs: Fragmentation with the bandwidth settings "Maximum 50% cyclic IO data. Balanced proportion." and "Maximum 90% cyclic IO data. Focus on cyclic IO data."

The PROFINET IO interface of the controller supports fragmentation if all the ports except one are deactivated.

Optimizing port settings for low send clocks

You can further optimize the bandwidth use in your PROFINET IO system by using cables with a short cable length (< 20 m) or a short signal delay (max. 0.12 μs) between the devices.

To configure cables with a shorter length or signal delay in STEP 7, follow these steps:

1. Select the port in the topology view of STEP 7.
2. Navigate in the Inspector window to "Port interconnection" in the "Partner port" box.
3. Select the option "Cable length" or "Signal delay:".
4. Select the cable length or enter the signal delay.

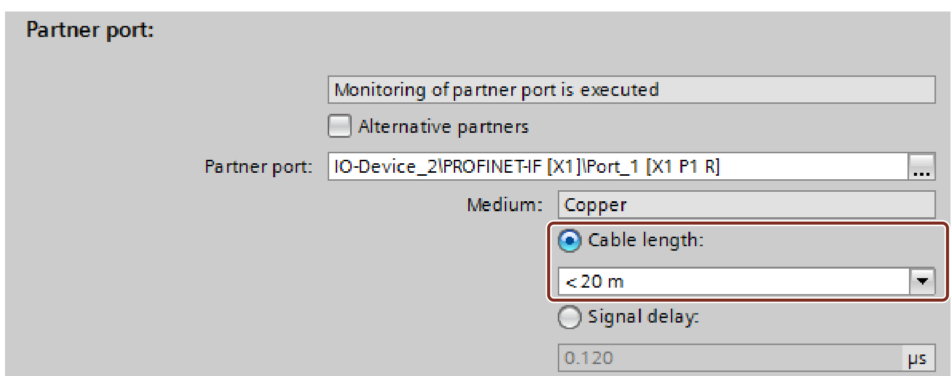


Figure 6-71 Optimizing port settings for low send clocks

Allowing fast forwarding

Requirements:

- The PROFINET IO device has to support the fast forwarding process so that the process can be used by the device.
- The "Make 'high performance' possible" option is enabled.
- The PROFINET IO interface supports fast forwarding if all the ports except one are deactivated.

Follow these steps to allow fast forwarding:

1. Select the PROFINET IO system in the network view of STEP 7.
2. In the Inspector window, go to "Properties" > "General" > "PROFINET" > "Domain management" > "Sync domains" > "Name of the sync domain".

3. Enable the "Allows the use of ‘fast forwarding'" option.

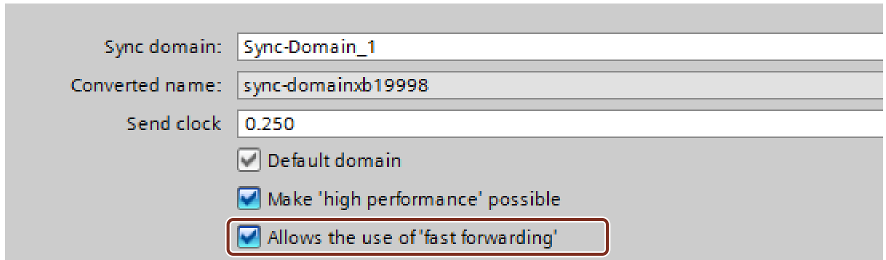


Figure 6-72 Allowing fast forwarding

NOTE

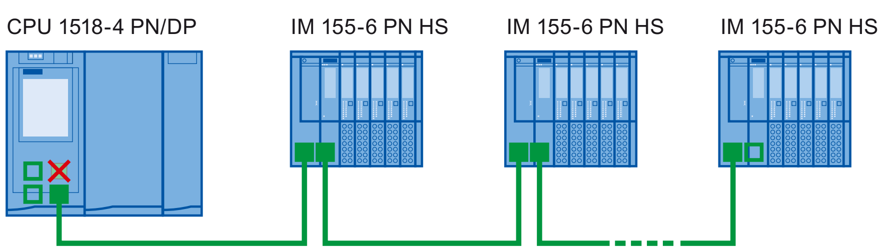
Fast forwarding and IPv6

The operation of fast forwarding in combination with IPv6 is not supported.

As soon as an IO device in the subnet uses an IPv6 address, you must not activate "fast forwarding".

6.6.5 Sample configuration for IRT with high performance

The figure below shows an example configuration with which you can achieve maximum performance.



Port 1 of the X1 interface of the CPU is deactivated.

Port 2 of the X1 interface and the ports of the interface modules on the bus use the following setting: Medium copper, cable length < 20 m or signal delay max. 0.12µs

A programming device connection to the CPU can be established via the interfaces X2 and X3 of the CPU or via the free port at the end of the line.

Figure 6-73 Example configuration for IRT with high performance

Use the following settings for the Sync domain:

- Enable the "Make 'high performance' possible" option.
- Set the send clock to 125 µs.
- Enable the "Allows the use of ‘fast forwarding'" option.

Standard Ethernet communication for IRT with high performance

Standard Ethernet communication is still possible even in a PROFINET IO system with high performance. Keep in mind that you first arrange the IRT nodes, as seen from the IO controller, and arrange the standard Ethernet nodes at the end of the line.

With a large volume of data through standard Ethernet communication, it makes sense to reduce the load on your network by separating standard Ethernet communication and cyclic real-time communication. Example: Use the interface X1 for the PROFINET IO communication and a different interface for the standard Ethernet communication.

6.7 Isochronous mode

6.7.1 What is isochronous mode?

Objectives of isochronous operation

The advantages of the isochronous mode function in automation engineering can be seen in an example from everyday life.

The transmission of data corresponds to the transport of people on public transport. Assuming public transport were to operate at maximum speed while reducing stop times at the passenger terminals to absolute minimum, the last thing many potential passengers would notice of the departing contraption are its red tail lights. The overall travel time is, however, decided by the train, bus or underground-railway clock, because well adjusted timing is essential to a good service. This also applies in automation engineering. Not only fast cycles but also the adaptation and synchronization of the individual cycles result in optimum throughput.

Just-In-Time

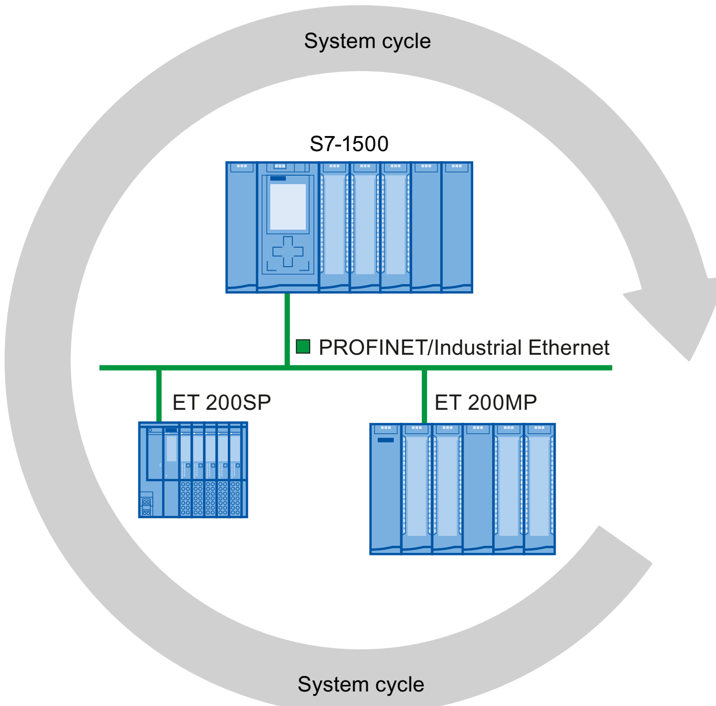


Figure 6-74 System cycle

The fast and reliable response time of a system operating in isochronous mode is due to the fact that all data is provided just-in-time. The basis for this is an equidistant cycle.

The isochronous mode function guarantees synchronization of the following at constant time intervals:

- Signal acquisition and output by the distributed I/O
- Signal transmission via PROFINET IO
- Program execution in the CPU in time with the equidistant PROFINET IO

The result is a system that acquires its input signals, processes them and outputs the output signals at constant time intervals. Isochronous mode guarantees precisely reproducible and defined process response times as well as equidistant and synchronous signal processing for distributed I/O.

Advantages of isochronous mode

The use of isochronous mode allows high-precision control.

- Optimized control loops through constant, calculable dead times
- Determinism, reliable reproducibility of response times
- Consistent (simultaneous) reading in of input data
- Consistent (simultaneous) output of output data

6.7.2 Use of isochronous mode

An isochronous system acquires measured values and process data within a fixed system cycle, processes the signals and outputs them synchronously to the process. Isochronous mode contributes to a high control quality and increased production accuracy. With isochronous mode, the possible fluctuations of process response times are drastically reduced. You make use of the time-assured processing to improve machine cycle times. Even fast processes can be reliably controlled thanks to the precise time reproducibility of all sequences. Shorter cycle times increase the processing speed and help to lower production costs.

In principle, isochronous mode is worthwhile whenever measured values must be acquired synchronously, movements must be coordinated, and process responses must be defined and simultaneously executed, as in the following example. Isochronous mode can thus be used in a wide variety of applications.

Example: Measuring at multiple measurement points with isochronous mode

Automation task

A camshaft production process requires precise measurement of the camshafts for quality assurance purposes.

Feature

For this purpose, a component is needed that can synchronously measure the positions and displacements of the cam during a rotation of the camshaft.

Solution

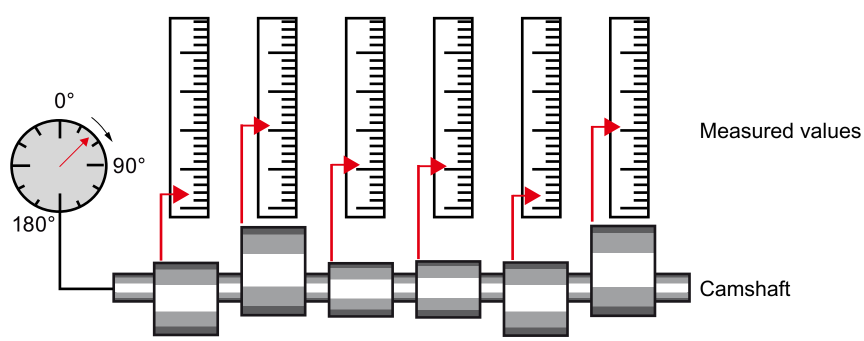


Figure 6-75 Measuring of camshafts

With use of isochronous mode, the measured values are simultaneously acquired at the various measuring points at fixed times. This results in the following workflow:

- Continuously rotate the camshaft
- During continuous rotation, synchronously measure the positions and cam displacements
- Process the next cam shaft

Thus, during a single rotation of the camshaft, all the positions of the camshaft and the associated measured values (red) are measured synchronously. The machine cycle time improves with the same or better measuring accuracy.

Advantage and benefits

The time required for measurement is reduced.

6.7.3 Time sequence of synchronization on PROFINET IO

Introduction

On PROFINET IO, you can isochronously operate I/O modules in distributed I/O systems on a CPU, e.g. in distributed I/O systems ET 200SP, ET 200MP.
Like the I/O modules, the interface modules of the I/O systems must support isochronous mode.

From reading in input data to outputting output data

The basic time sequence of all components involved in synchronization is explained below:

① Measured value acquisition in the process

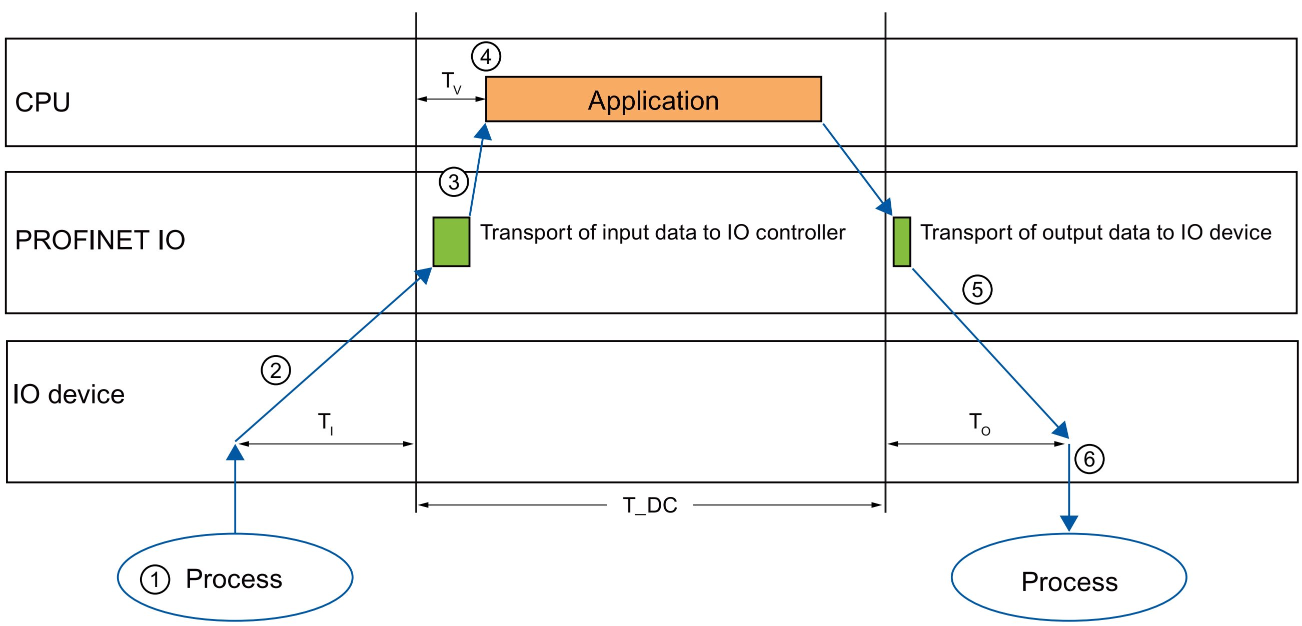
② Isochronous read-in of input data

③ Transport of input data to the IO controller (CPU) via the subnet

④ Further processing in the isochronous application of the CPU

⑤ Transport of output data to the outputting IO device via the subnet

⑥ Isochronous output of output data



T_DC Data cycle (Time_DataCycle)

T_I Time for reading in the input data

T_O Time for outputting the output data

T_V Configured delay time

Figure 6-76 Time sequence of synchronization on PROFINET IO

To ensure that all input data is ready for transport via the subnet at the next start of the PROFINET IO cycle, the start of the I/O read-in cycle is advanced by the amount of lead time T_I. T_I is the "flashbulb" for the inputs; at this instant, all synchronized inputs are read in. T_I is necessary in order to compensate for analog-to-digital conversion, backplane bus times, and the like. The lead time T_I can be configured by STEP 7 or by you.

Let the lead time TI be assigned automatically by STEP 7. With the default setting, STEP 7 ensures that a common, minimum T_I is set.
The subnet transports the input data to the IO controller/DP master. The application is started synchronized to the cycle. That is, the isochronous mode interrupt OB is called after a configurable delay time T_V. The user program in the isochronous mode interrupt OB defines the process response and provides the output data in time for the start of the next data cycle. The length of the data cycle (send clock/DP cycle time) is always configured by you.
T_O is the time for the compensation arising from the backplane bus and the digital-to-analog conversion within the IO device/DP slave. T_O is the "flashbulb" for the outputs. The synchronized outputs are output at this time. Time T_O can be configured by STEP 7 or by you. Let time T_O be assigned automatically by STEP 7. STEP 7 automatically calculates a common, minimum T_O.

6.7.4 Configuring isochronous mode

6.7.4.1 Introduction

Configuring at a glance

Setting parameters for isochronous operation of the I/O module

You use the properties of the I/O addresses of the corresponding I/O module to:

- Set isochronous mode for the module.
- Assign the inputs and outputs of the module to a process image partition and an isochronous mode interrupt OB.
  The data of the process image partition is updated synchronously to the assigned OB. Isochronous mode interrupts give you the option of starting programs isochronously with the PROFINET send clock. Isochronous mode interrupts are processed with high priority.

Setting the send clock

The send clock is the shortest possible transmission interval for the data exchange. In isochronous mode, the send clock corresponds to data cycle T_DC.

You set the send clock in the properties of the PROFINET interface of the CPU or in the sync domain.

Setting the application cycle

The application cycle is a multiple of data cycle T_DC. If the runtime of the isochronous mode interrupt OB is short, the application cycle can be identical to the data cycle (= send clock).

You can reduce the application cycle of the isochronous mode interrupt OB relative to the send clock of an isochronous PROFINET IO system. Set an integer multiple of the send clock as the reducing factor.

You use the factor to reduce the CPU utilization by executing the isochronous mode interrupt OB less frequently. You set the application cycle in the properties of the isochronous mode interrupt OB.

Setting the delay time

The delay time is the time between the start of the send clock and the start of the isochronous mode interrupt OB. During this time, the IO controller performs cyclic data exchange with the IO devices.

STEP 7 sets the default delay time in such a way that the isochronous update of the process image partition automatically falls within the execution window of the application cycle.

You set the delay time in the properties of the isochronous mode interrupt OB. A shorter delay time enables you to increase the processing time for your user program in the isochronous mode interrupt OB.

Additional configurations for isochronous mode on PROFINET IO:

- **Setting IRT as the RT class for the interconnected PROFINET interfaces**
  A precondition for isochronous operation on PROFINET IO is IRT communication (Isochronous Real Time Communication). IRT means synchronized data exchange at reserved time intervals.
- **Configuring the topology of the configuration**
  A precondition for IRT communication is the topology configuration. In addition to the reserved bandwidth, the exchange of frames on defined transmission paths is used for further optimization of data communication. For this, the topological information of the configuration is used for planning the communication.
- **Using a sync domain**, you assign the IO devices (sync slaves) to an IO controller (sync master) for the isochronous data exchange.
  A precondition for IRT communication is a synchronization cycle for all PROFINET devices in a sync domain, for distribution of a common time base. With this basic synchronization, synchronous operation of the transmission cycle of the PROFINET devices within a sync domain is achieved.

6.7.4.2 Configuring isochronous mode on PROFINET IO

Introduction

The configuring of isochronous mode for a module is described as IO device in the following based on the ET 200MP distributed I/O system. The procedure described also applies to other distributed I/O systems (e.g., ET 200S or ET 200SP).
The IO controller is an S7-1500 CPU.

Requirements

- The STEP 7 network view is open.
- A S7-1500 CPU has been placed (e.g., CPU 1516-3 PN/DP).
- An IM 155-5 PN HF interface module (ET 200MP) has been placed and networked with the CPU via PROFINET IO.
- All requirements for an IRT configuration are met:
  - The ports of the networked PROFINET interfaces of the CPU and interface module are interconnected (topology configuration).
  - The RT class of the PROFINET interface of the interface module is set to "IRT" (area "Advanced options > Real time settings > Synchronization").
  - The "sync master" and "sync slave" roles are assigned for the PROFINET interfaces of the CPU and interface module (in the properties of a PROFINET interface: "Advanced options > Real time settings > Synchronization > Domain settings" area).

Procedure

To create an isochronous connection between the I/O and user program, follow these steps:

1. Select the IM 155-5 PN HF in the network view of STEP 7. Change to the device view.
2. Insert an I/O module that can be operated isochronously (e.g. DI 16 x 24VDC HF).
3. Go to the "I/O addresses" area in the Inspector window of the selected I/O module.

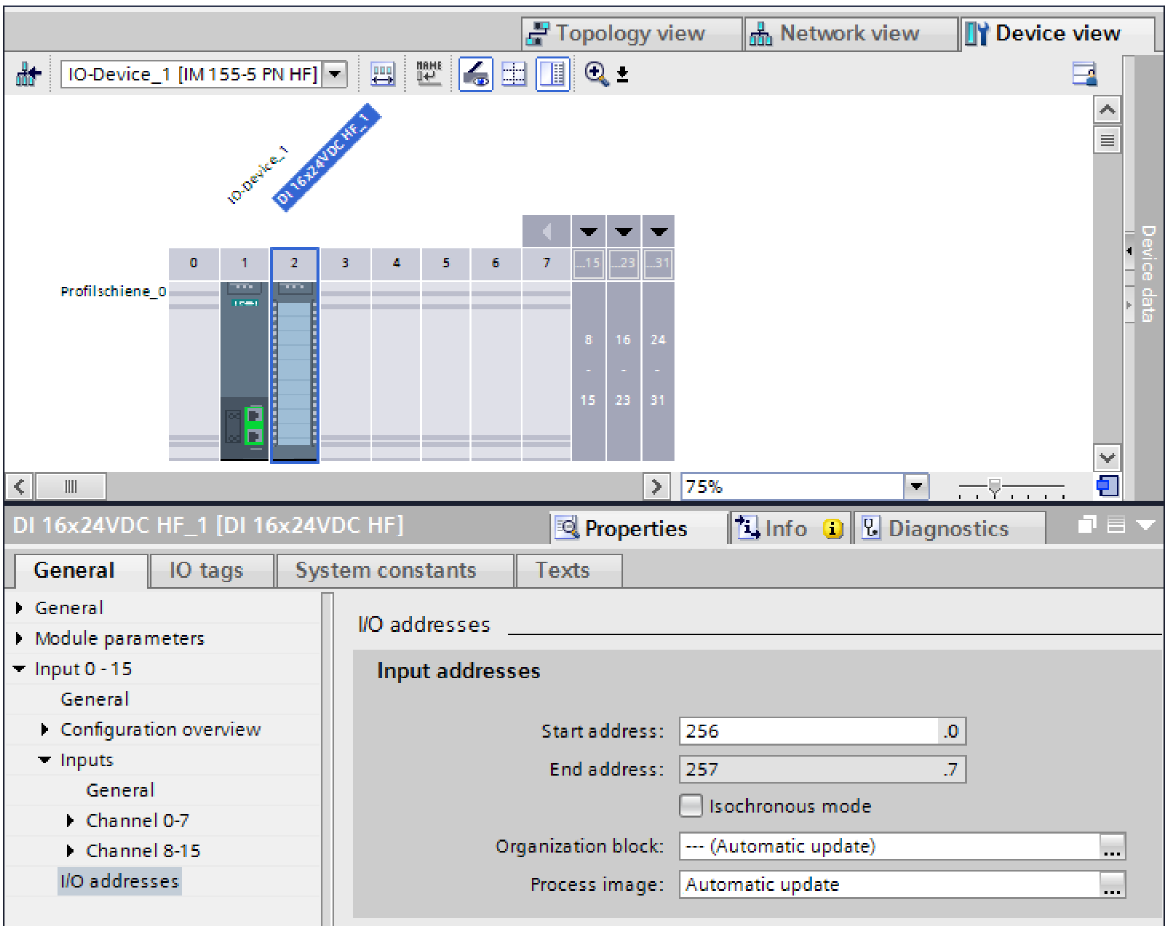


Figure 6-77 Configuring isochronous mode on PROFINET IO

4. Make the following settings in the I/O addresses area:
   - Select the "Isochronous mode" option.
   - Select a process image partition, e.g., process image partition 1.
   - Click the "Organization block" drop-down list. Click the "Add" button or select an existing OB. A dialog box for selecting organization blocks opens.
   - Select the "Synchronous Cycle" OB. Confirm the selection with "OK".

     In the case of automatic number assignment, OB 61 will be generated and opened.

     In the Inspector window, you can continue directly with the setting of the application cycle and delay time (Page 226) in the "Isochronous mode" area and start the programming of the OB in the instruction section.

5. If required, insert additional IO devices in the network view. Adapt the configuration and the settings for the isochronous mode.
6. You want to retrieve information about calculated bandwidths or on adapting the send clock. Select the sync domain in the network view and navigate to the corresponding area of the domain management in the Inspector window.

Reference

You can find examples of parameter assignment and possible settings of isochronous mode for distributed I/O and drives in STEP 7 in this FAQ on the Internet (https://support.industry.siemens.com/cs/ww/en/view/109480489).

6.7.4.3 Setting the application cycle and delay time

Requirements

- You have created an isochronous mode configuration in STEP 7.
- You have created an isochronous mode interrupt OB Synchronous Cycle (OB 6x).
- The isochronous mode interrupt OB is open.

Setting the application cycle

The application cycle is a multiple of data cycle T_DC (send clock). You use the application cycle setting to reduce the CPU utilization caused by execution of the isochronous mode interrupt OB. In the following example, the OB is called only after every 2nd data cycle T_DC in the CPU.

To set the application cycle for your isochronous mode application, follow these steps:

1. Open the "Properties" dialog of the isochronous mode interrupt OB under consideration.
2. In the area navigation, click the "Isochronous mode" group .
3. Set the application cycle in "Application cycle (ms)". Open the drop-down list box and select the application cycle. The drop-down list offers multiples of data cycle T_DC as possible values for the application cycle. Data cycle T_DC is set to 2 ms in the following figure.

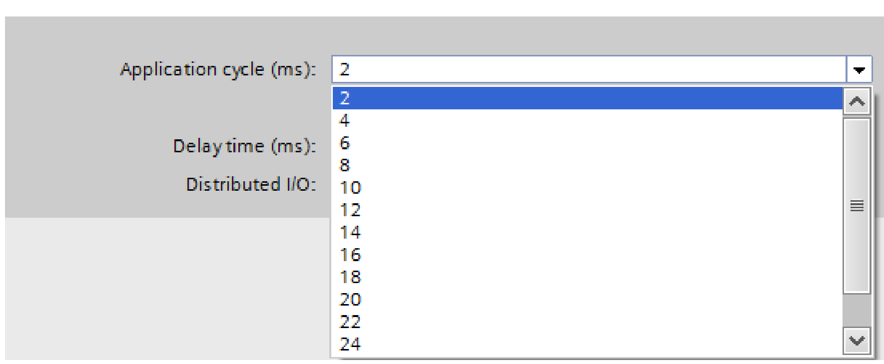


Figure 6-78 Setting the application cycle

Setting the delay time

The delay time is the time between the start of the send clock and the start of the isochronous mode interrupt OB. STEP 7 sets the delay time automatically to the start of the execution window by default. As a result, the isochronous mode update of the process image partition automatically falls within the execution window of the application cycle.

You can also set the delay time manually. A shorter delay time enables you to increase the processing time for your user program in the isochronous mode interrupt OB. If a manual setting is made, note that you must call the "SYNC_PI" and "SYNC_PO" instructions in the execution window of the application cycle.

To set the delay time for your isochronous mode application, follow these steps:

1. Open the "Properties" dialog of the isochronous mode interrupt OB under consideration.
2. In the area navigation, click the "Isochronous mode" group .
3. Clear the "Automatic setting" check box.
4. Enter your desired delay time in "Delay time (ms)".

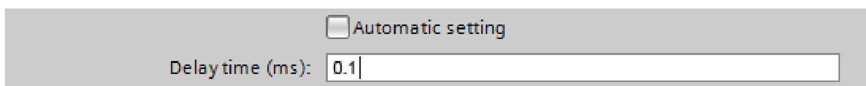


Figure 6-79 Setting the delay time

6.7.5 Programming isochronous mode

6.7.5.1 Basics of Programming

Programming in the isochronous mode interrupt OBs

You program the isochronous section of the program exclusively in the isochronous mode interrupt OBs Synchronous Cycle (OB 6x).

Because the isochronous mode interrupts are processed with high priority, only the time-critical sections of the program should be processed in the isochronous mode interrupt OB. The isochronous mode interrupt OB is called with a configured delay time.

Access to isochronous I/O through call of instructions

You access the isochronous I/O via a process image partition. That is, the addresses of the isochronous modules must be within one process image partition.

You program access to the isochronous IO in isochronous mode interrupt OB Synchronous Cycle (OB 6x) with the SYNC_PI and SYNC_PO instructions.

The isochronous I/O is updated by the call of the "SYNC_PI" and "SYNC_PO" instructions and therefore in the corresponding process image partition.

NOTE

Recommendation: To prevent inconsistent data from being returned to OB 6x, do not use the "DPRD_DAT" and "DPWR_DAT" instructions (direct data access) in the isochronous mode interrupt OB.

The "SYNC_PI" and "SYNC_PO" instructions update the process image partition only within the permitted execution window. The execution window stretches from the end of the cyclic data exchange to the point in time before the end of T_DC at which the outputs can still be copied in time. The data exchange must be started within this time window. If the execution window is violated by the processing of the "SYNC_PI" and "SYNC_PO" instructions, the instructions indicate a corresponding error message.

Program execution models

Depending on the order of the "SYNC_PI" and "SYNC_PO" instruction calls in OB 6x, there are two basic models for the program execution:

- **IPO** model (read **I**nputs - **P**rocessing - write **O**utputs)
- **OIP** model (write **O**utputs - read **I**nputs - **P**rocessing)

6.7.5.2 Program execution according to the IPO model

If the execution time of the isochronous mode interrupt OB is significantly shorter than one data cycle T_DC, use the IPO model. In the IPO model, you do not reduce the data cycle. That is, the application cycle of the isochronous mode interrupt OB is equal to data cycle T_DC. The IPO model enables the shortest response times.

Programming according to the IPO model in the isochronous mode interrupt OB

For programming according to the IPO model:

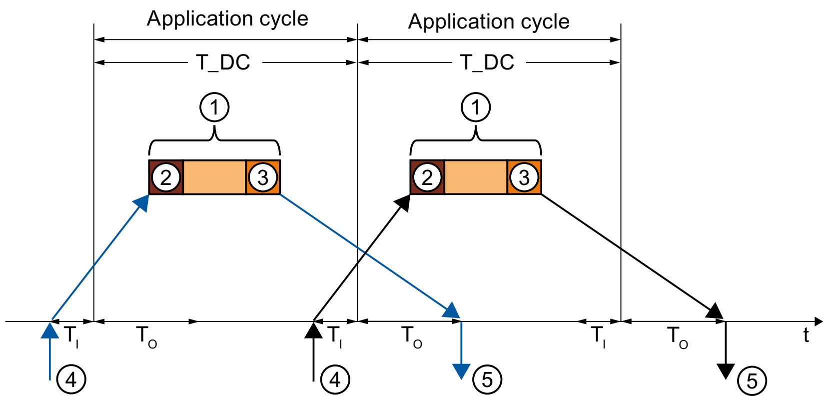
1. Call the SYNC_PI instruction at the start of the isochronous mode interrupt OB.
2. Then call the actual user program.
3. Call the SYNC_PO instruction at the end of the isochronous mode interrupt OB.

Table 6-3 Programming according to the IPO model

Step	Action	Explanation
1	Read in (I)	The SYNC_PI instruction reads in the inputs of the process image partition and provides them to the isochronous mode interrupt OB.
2	Process (P)	You program the actual user program of the isochronous mode interrupt OB.
3	Output (O)	The SYNC_PO instruction outputs the data changed by the user program via the process image partition.

Signal sequence in the IPO model

The following figure shows the signal sequence in the IPO model from the acquisition and the processing in the CPU to the output of the process values:



① Execution of isochronous mode interrupt OB

② "SYNC_PI" instruction

③ "SYNC_PO" instruction

④ Isochronous read-in of process values on I/O module at time T_I

⑤ Isochronous output of process values on I/O module at time T_O

Figure 6-80 Signal sequence in the IPO model

At time T_I the process values are read in isochronously on the I/O. The processing of the data in the IPO model is completed within one data cycle T_DC. The output data is always available on the I/O in the next data cycle T_DC at time T_O.
With the IPO model, there is a constant execution time from the "input terminal" to the "output terminal" of T_I + T_DC + T_O.
T_I + 2×T_DC + T_O can be guaranteed for the process response time.

6.7.5.3 Program execution according to the OIP model

Use the OIP model in the case of execution cycles of the isochronous mode interrupt OB of different lengths, if the application cycle is greater than data cycle T_DC.
The data exchange with the process is always deterministic in the OIP model, i.e. takes place at a precisely specified time.

Programming according to the OIP model in the isochronous mode interrupt OB

For programming according to the OIP model:

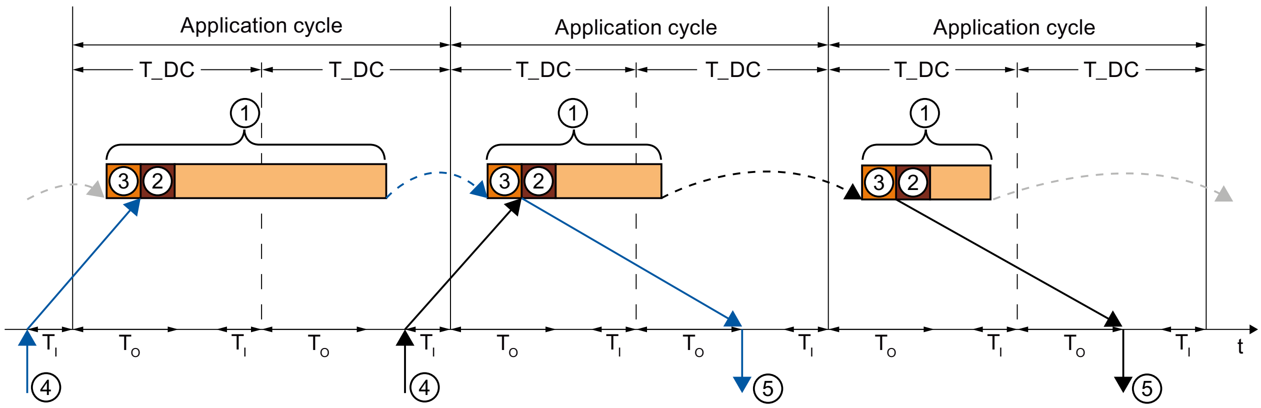
1. Call the SYNC_PO instruction at the start of the isochronous mode interrupt OB.
2. Then call the SYNC_PI instruction.
3. Then call the actual user program.

Table 6-4 Programming according to the OIP model

Step	Action	Explanation
1	Output (O)	The SYNC_PO instruction outputs the data that was changed by the user program in the previous cycle via the outputs of the process image partition.
2	Read in (I)	The SYNC_PI instruction reads in the inputs of the process image partition of the current cycle and provides the inputs to the isochronous mode interrupt OB.
3	Process (P)	You program the actual user program of the isochronous mode interrupt OB.

Signal sequence in the OIP model

The following figure shows the signal sequence in the OIP model from the acquisition of process values and the processing in the CPU to the output of the process values. The application cycle is twice as long as data cycle T_DC in this example.



① Execution of isochronous mode interrupt OB

② "SYNC_PI" instruction

③ "SYNC_PO" instruction

④ Isochronous read-in of process values on I/O module at time T_I

⑤ Isochronous output of process values on I/O module at time T_O

Figure 6-81 Signal sequence in the OIP model

At time T_I the process values are read in isochronously on the I/O. The data is processed over 2 application cycles in the OIP model. The output data is always available on the I/O in the following application cycle at time T_DC + T_O.
With the IPO model, there is a constant execution time from the "input terminal" to the "output terminal" of T_I + application cycle + T_DC + T_O.
T_I + 2 x application cycle + T_DC + T_O can be ensured as the process response time.

6.8 Direct data exchange

6.8.1 Introduction

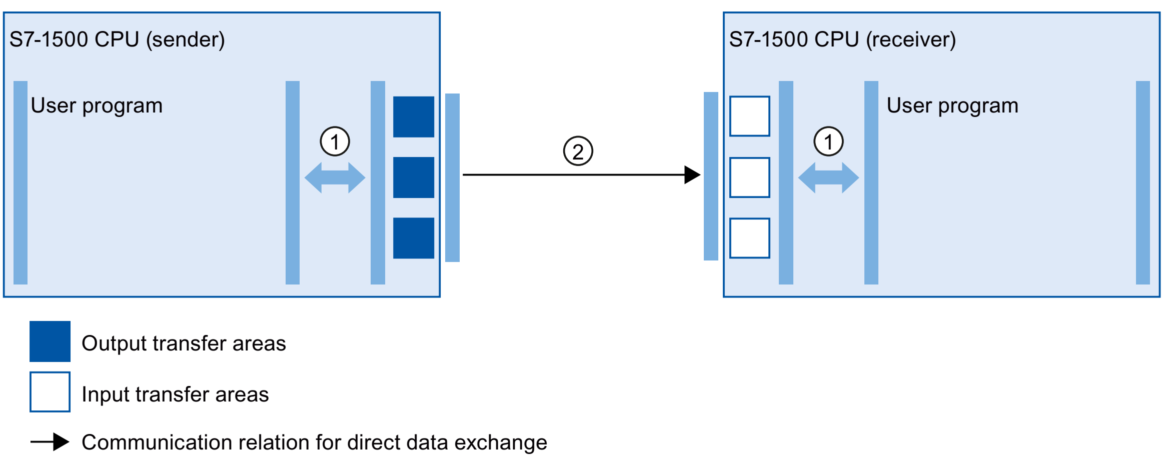
This section describes the direct data exchange function.

Principle of operation

Starting with firmware version V2.8, the S7-1500 CPU supports direct data exchange (cross data traffic) with other S7-1500 CPUs.
In the case of direct data exchange, an S7-1500 CPU provides cyclic user data from the I/O area to one or more partners. The direct data exchange is based on PROFINET with IRT and isochronous mode.
The data exchange takes place via transfer areas.

Direct data exchange between two S7-1500 CPUs (1:1)

The figure below shows the direct data exchange between two S7-1500 CPUs. The output transfer areas of the sending S7-1500 CPU correspond to the input transfer areas of the receiving S7-1500 CPU.



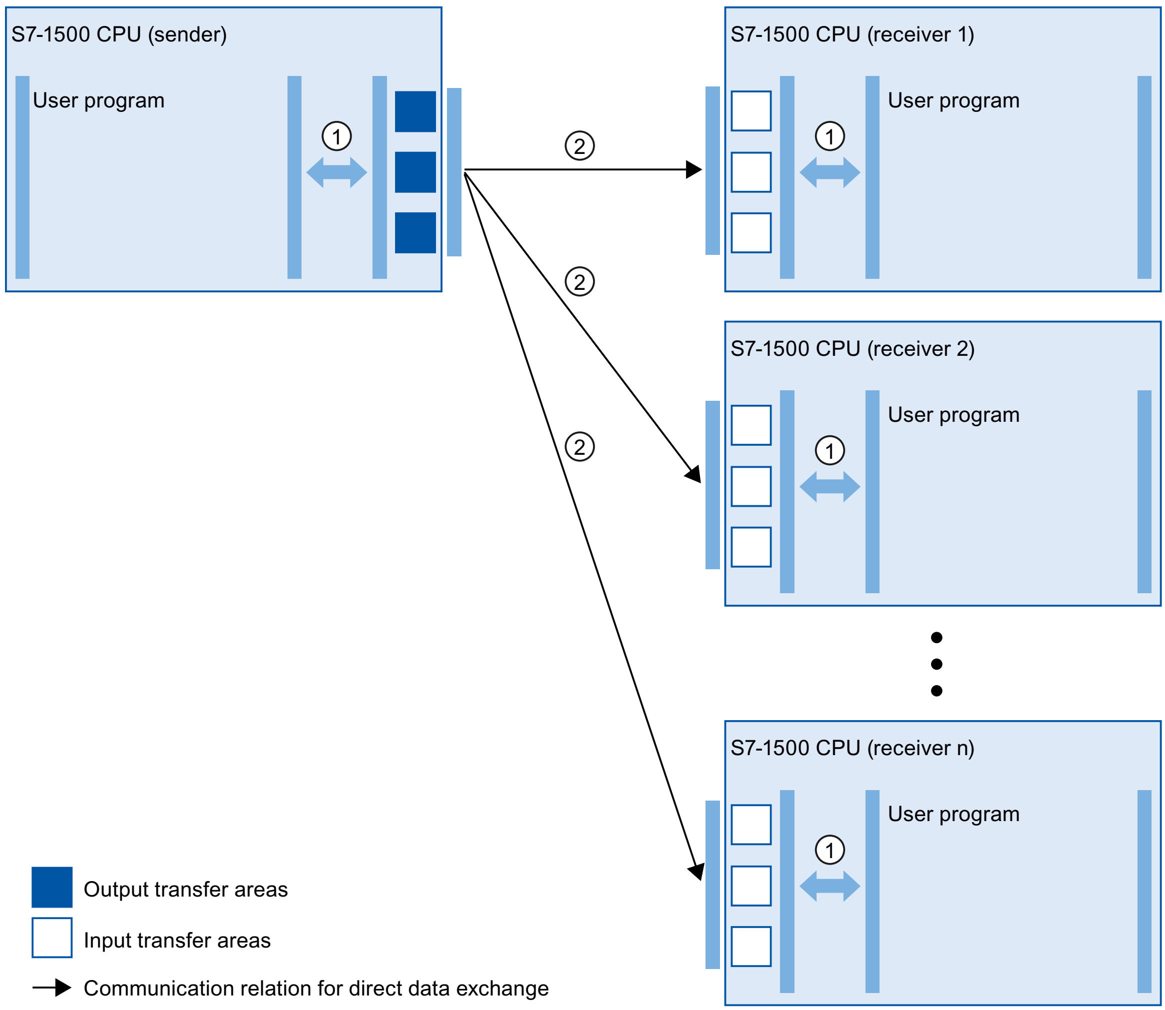
① Transfer relation between the user program and the transfer area. The user program and the transfer areas exchange input and output data via this path.

② Communication relation for direct data exchange You configure the communication relations in STEP 7. You create transfer areas for direct data exchange in the properties of the communication relations.

Figure 6-82 Direct data exchange between two S7-1500 CPUs (1:1)

Direct data exchange with multiple receivers (1:n)

The following figure shows the direct data exchange with multiple S7-1500 CPUs. In this case, the sending S7-1500 CPU provides the data of its output transfer areas to multiple S7-1500 CPUs. Each receiving S7-1500 CPU has its own input transfer areas.



① Transfer relation between the user program and the transfer area. The user program and the transfer areas exchange input and output data via this path.

② Communication relation for direct data exchange You configure the communication relations in STEP 7. You create transfer areas for direct data exchange in the properties of the communication relations.

Figure 6-83 Direct data exchange with multiple receivers (1:n)

Applications

- Deterministic, isochronous I/O communication between multiple S7-1500 CPUs

Properties of direct data exchange

- Always isochronous
- Support of MRPD for MRP configuration
- No acyclic data exchange via PROFINET data record mechanisms
- No PROFINET alarms
- Configuration limits:
  - Maximum data length for direct data exchange 3075 bytes including user data qualifier
  - Maximum data length for a transfer area: 1024 bytes without user data qualifier
  - Maximum number of transfer areas of sender: 128
  - Maximum number of transfer areas of receiver: 512, distributed among a maximum of 64 receivable PROFINET frames and thus up to 64 sender CPUs

Diagnostics options of the receiver

Operating state change of sender:

- When the sender goes from RUN to STOP, the receiver behaves as follows:
  - The "SYNC_PI" and "SYNC_PO" instructions return an error message in parameter RET_VAL during synchronization of the process image.
  - With direct I/O access to the input transfer areas of the direct data exchange, OB 122 "I/O access error" is called, if present.
  - Incoming diagnostic message "I/O data failure in hardware component"
- When the sender goes from STOP to RUN, the receiver behaves as follows:
  - Call of OB 83 "Pull/plug interrupt" for input transfer areas of the direct data exchange
  - Up until the call of OB 83, OB 122 is called if present.
  - Outgoing diagnostic message "User data failure of hardware component"

Station failure/station recovery of the sender:

- When the sender fails, e.g. due to a bus interruption, the receiver behaves as follows:
  - Call of OB86 "Rack failure"
  - The "SYNC_PI" and "SYNC_PO" instructions return an error message in parameter RET_VAL during synchronization of the process image.
  - With direct I/O access to the input transfer areas of the direct data exchange, OB 122 "I/O access error" is called, if present.
- When the sender recovers after a station failure, e.g. because the bus connection is re-established, the receiver behaves as follows:
  - Call of OB86 "Rack failure"

6.8.2 Configuring direct data exchange between two S7-1500 CPUs

The procedure for configuring direct data exchange between two IO controllers is described below.
First, you create the communication relation for direct data exchange. You then configure transfer areas for the connection.

Requirements

- STEP 7 V16 or higher
- Two S7-1500 CPUs firmware version V2.8 or higher
- IRT is configured:
  - Both CPUs are in one sync domain.
  - One CPU is the sync master, and the other CPU is the sync slave.
  - Ports are interconnected.

Setting up the communication relation for direct data exchange

To set up the communication relation for direct data exchange between two S7-1500 CPUs, follow these steps:

1. Select the PROFINET interface X1 of the sending S7-1500 CPU.
2. Change to the table view of the network view, tab "I/O communication".
   The PROFINET interface X1 of the CPU is shown in the "Partner 1" column.
3. In the "Partner 2" column at "<Drop or select the device here>", select the PROFINET interface of the communication partner from the drop-down list as the connection partner.
   Note the communication direction:
   - ←: Communication partner is sender
   - →: Communication partner is receiver

The communication relation for direct data exchange between the two S7-1500 CPUs is set up.

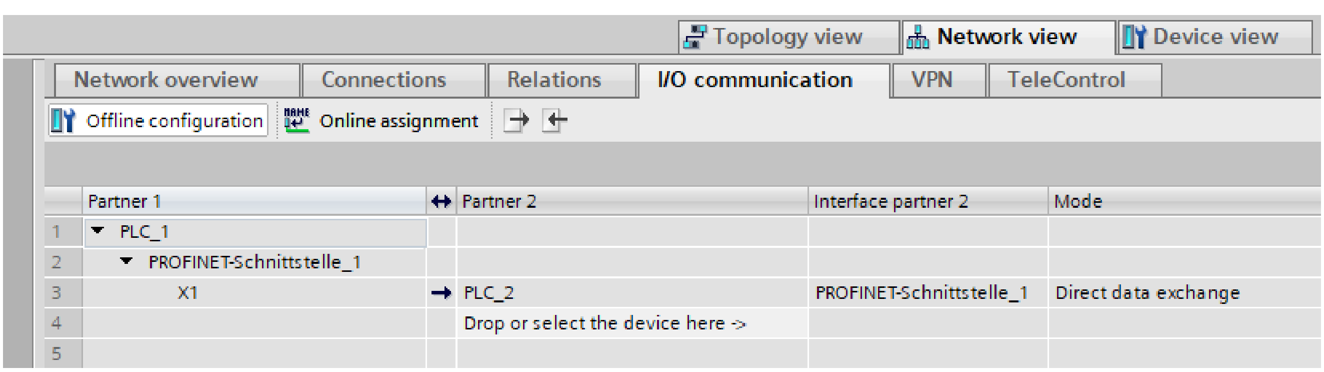


Figure 6-84 Communication relation for direct data exchange

Configuring transfer areas for direct data exchange

To configure a transfer area for direct data exchange, follow these steps:

1. Select the communication relation for direct data exchange.

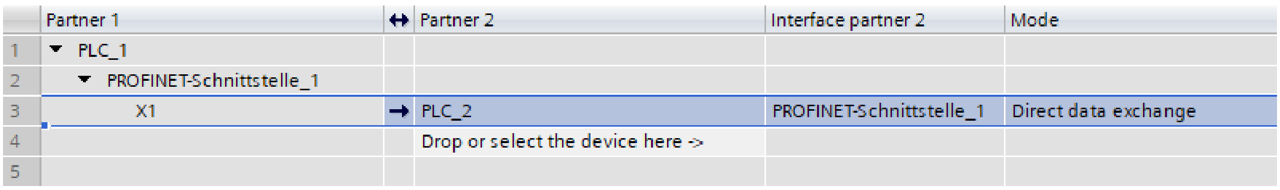


Figure 6-85 Communication relation for direct data exchange

2. Navigate to the properties of the communication relation to "General" > "Direct data exchange" > "Transfer areas".
3. Create a new transfer area by double-clicking on "<Add new>". Assign a meaningful name for the transfer area.

A transfer area for direct data exchange is created.

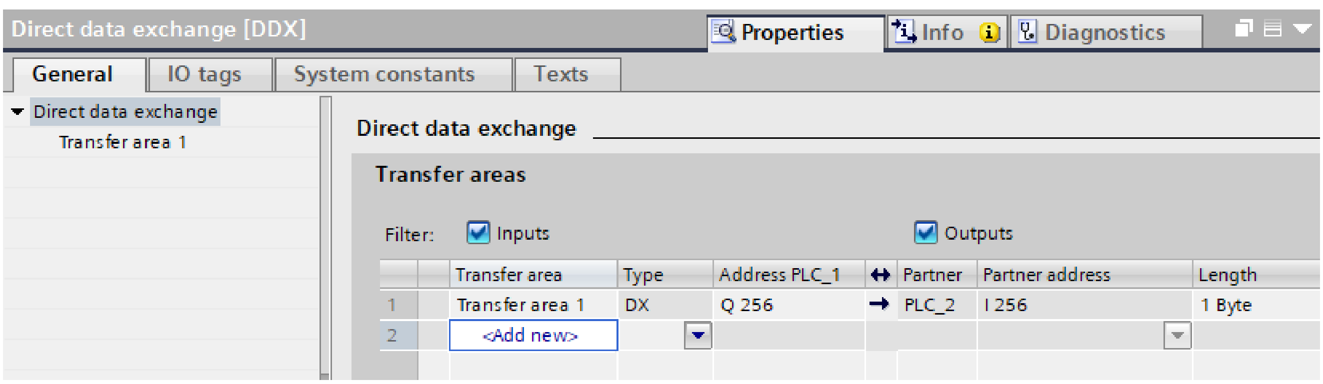


Figure 6-86 Transfer area for direct data exchange

The communication direction of the transfer area is specified by the communication relation. You cannot change the communication direction of the transfer area.

Editing the transfer area

Set the properties of the transfer area under "General" > "Direct data exchange" > "Name of transfer area" > "Detail of the transfer area".

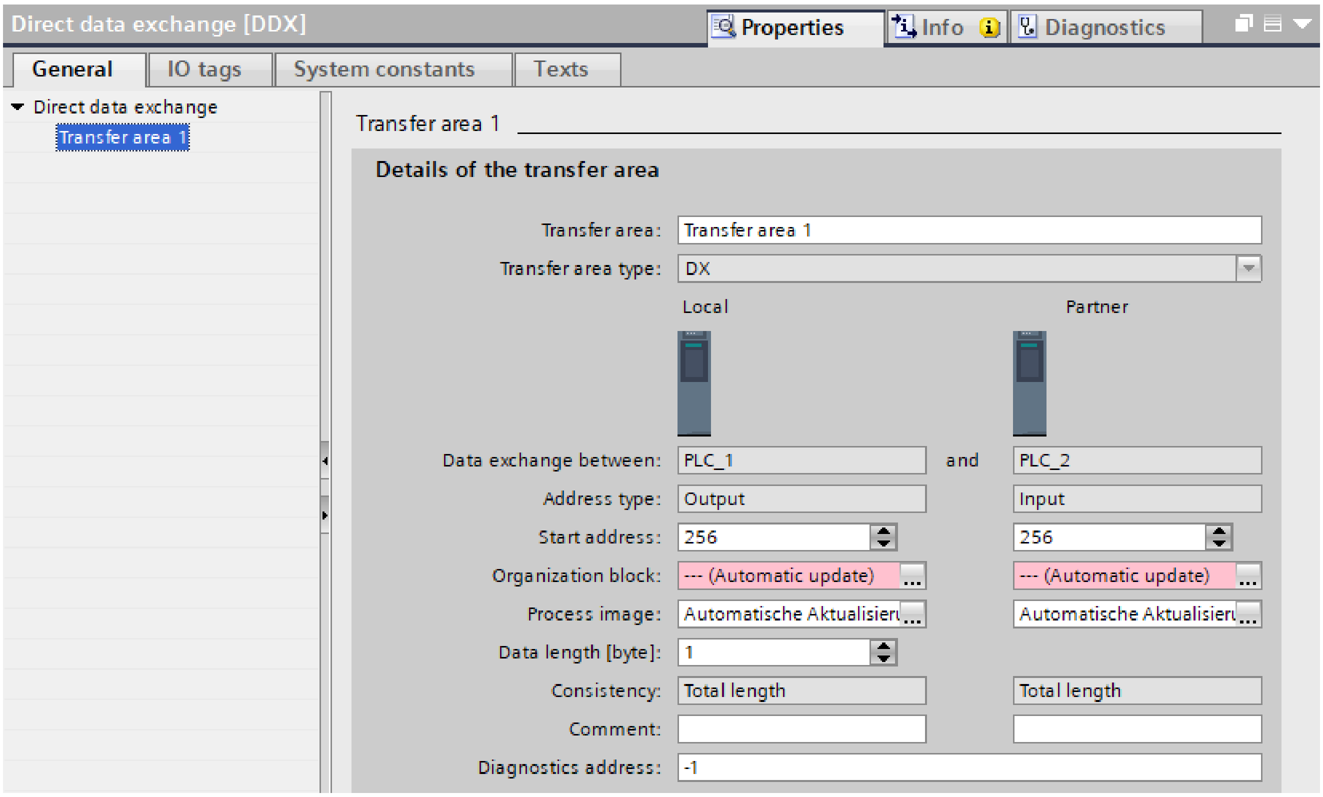


Figure 6-87 Properties of the transfer area

Table 6-5 Overview of the settings of the transfer area

Parameter	Local	Partner
Start address	Set the start address of the input or output transfer areas in the local CPU.	Set the start address of the input or output transfer areas in the partner CPU.
Organization block	Assign the transfer area to an isochronous mode interrupt OB or the "MC-Servo" OB.	Assign the transfer area to an isochronous mode interrupt OB or the "MC-Servo" OB.
Process image	Select a process image partition, e.g. PIP 1. If you have assigned "MC-Servo" as the organization block, STEP 7 automatically sets "PIP OB Servo" as the process image.	Select a process image partition, e.g. PIP 1. If you have assigned "MC-Servo" as the organization block, STEP 7 automatically sets "PIP OB Servo" as the process image.
Data length [bytes]	Set the size of the transfer area.	-

Downloading the configuration to devices

Rules:

- Download the configuration to all CPUs involved.
- If you make changes to the configuration of the direct data exchange, download these changes to all CPUs involved.

6.8.3 Configuring direct data exchange between multiple IO controllers

The procedure for configuring direct data exchange between multiple S7-1500 CPUs is described below.
First, you set up the communication relations for direct data exchange. You then configure transfer areas for the communication relations.

Requirements

- STEP 7 V16 or higher
- S7-1500 CPUs firmware version V2.8 or higher
- IRT is configured:
  - All CPUs are in one sync domain.
  - One CPU is the sync master, and the other CPUs are sync slaves.
  - Ports are interconnected.

Setting up the communication relations for direct data exchange

To set up the connection for direct data exchange between multiple S7-1500 CPUs, follow these steps:

1. Select the PROFINET interface X1 of the sending CPU.
2. Change to the table view of the network view, "I/O communication" tab.
   The PROFINET interface X1 of the CPU is shown in the "Partner 1" column.
3. In the "Partner 2" column at "<Drop or select the device here>", select the PROFINET interface of the communication partner from the drop-down list as the connection partner.
   Note the transfer direction:
   - ←: Connection partner is sender
   - →: Connection partner is receiver

   The connection for direct data exchange between the two S7-1500 CPUs is set up.
4. Repeat step 3 for every other receiving IO controller.

The connections for direct data exchange between the sending CPU and the receiving CPUs are set up.

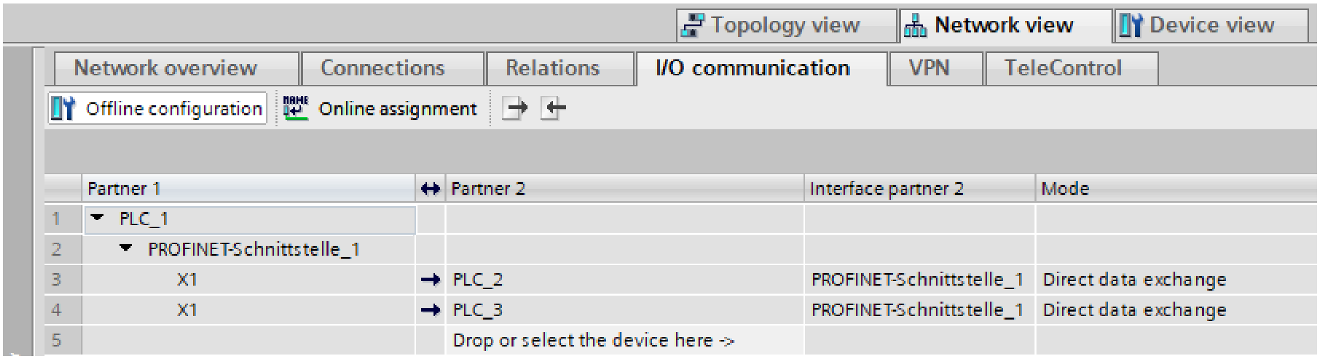
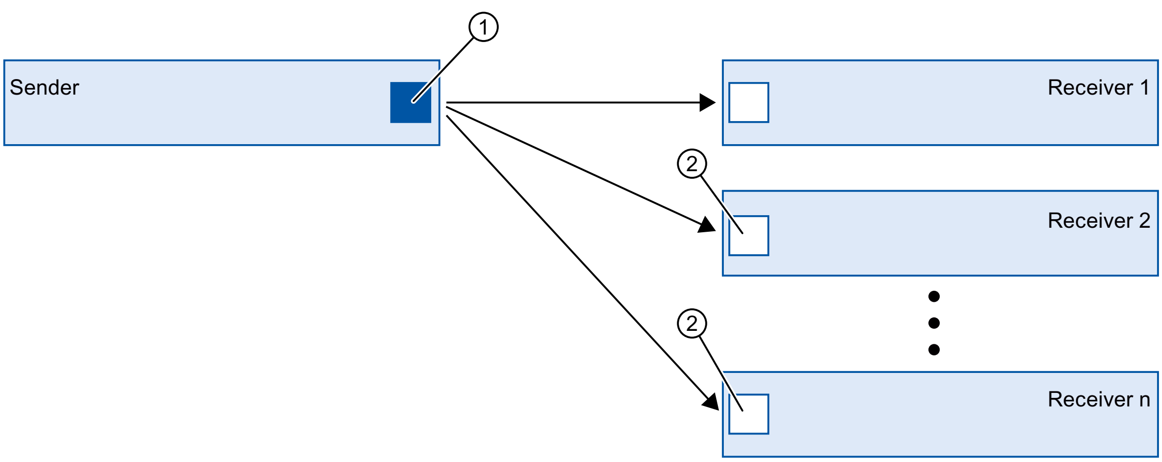


Figure 6-88 Connection for direct data exchange with multiple S7-1500 CPUs

Configuring transfer areas for direct data exchange

The following graphic shows the order for configuration of the transfer areas.



① First, you configure the transfer area between the sender and one receiver. You configure this transfer area at the PROFINET interface of the sender.

② You then configure the transfer areas between the sender and the other receivers. You configure these transfer areas at the PROFINET interfaces of the receivers.

Figure 6-89 Order of configuration of transfer areas in the case of multiple receivers

To configure a transfer area for direct data exchange, follow these steps:

1. Select the PROFINET interface X1 of the sending CPU and change to the table view, "I/O communication" tab.
2. In the table view of the network view, select a communication relation for direct data exchange between the sender and receiver 1.
3. Navigate to the properties of the I/O connection to "General" > "Direct data exchange" > "Transfer areas".

4. Create a new transfer area by double-clicking on "<Add new>". Assign a meaningful name for the transfer area.
   A transfer area for direct data exchange between the sender and receiver 1 is configured.

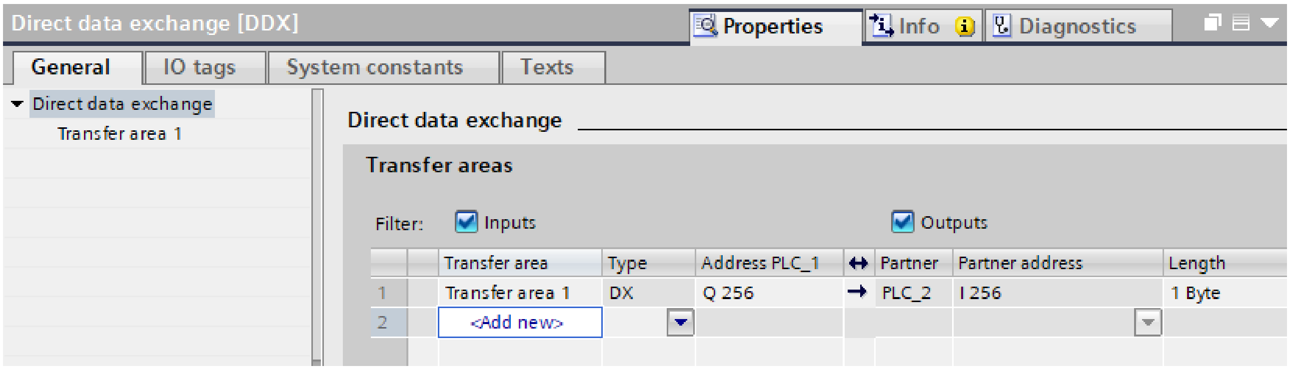


Figure 6-90 Transfer area for direct data exchange

5. Now, select the PROFINET interface of a receiver for which a transfer area is not yet set up, e.g. receiver 2.
6. Change to the table view of the network view to "I/O communication"
   The communication relation for direct data exchange with the sender is displayed.

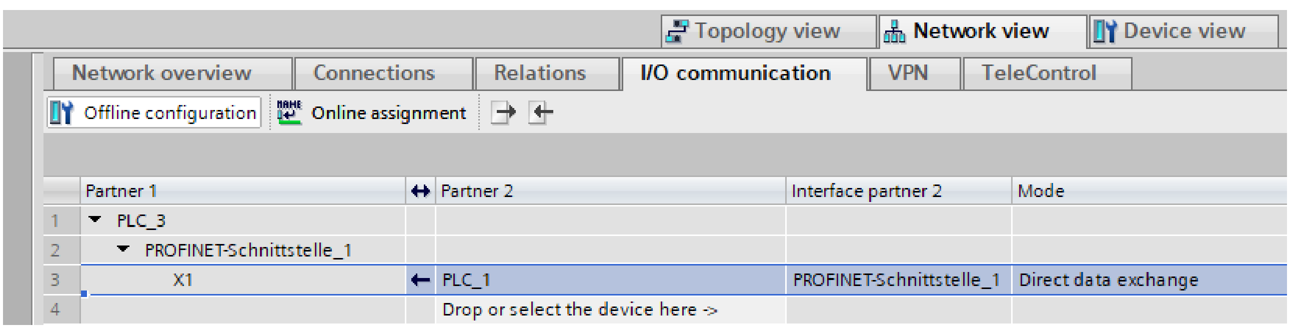


Figure 6-91 Connection for direct data exchange in receiver 2

7. Select the communication relation.
8. Navigate to the properties to "General" > "Direct data exchange" > "Transfer areas".
9. Create a new transfer area by double-clicking on "<Add new>". Assign a meaningful name for the transfer area.
   A transfer area for direct data exchange is configured.
10. Select the transfer area.

11. For "Partner address", select the existing address area in the sender as the output transfer area.

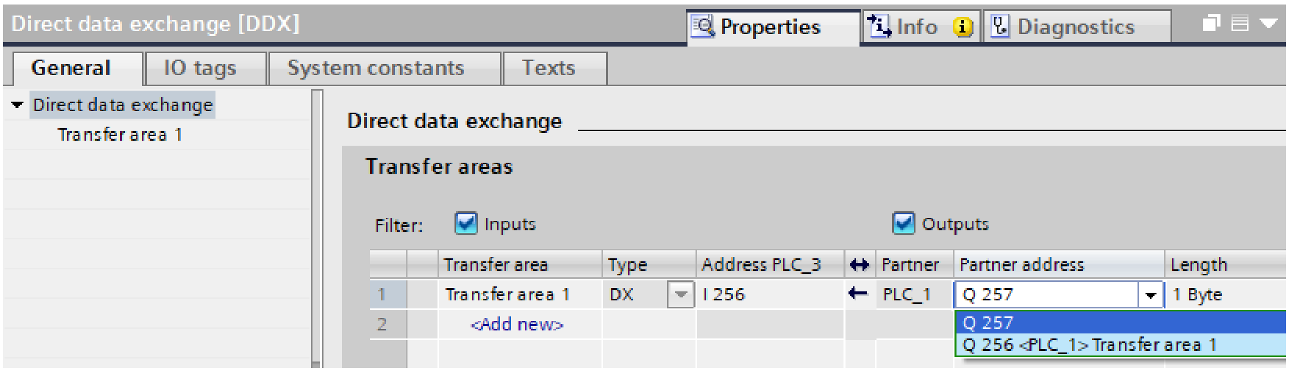


Figure 6-92 Transfer area for direct data exchange of multiple IO controllers

A transfer area for direct data exchange between receiver 2 and the sender is configured.

Editing the transfer area

Set the properties of the transfer area under "General" > "Direct data exchange" > "Name of transfer area" > "Detail of the transfer area".

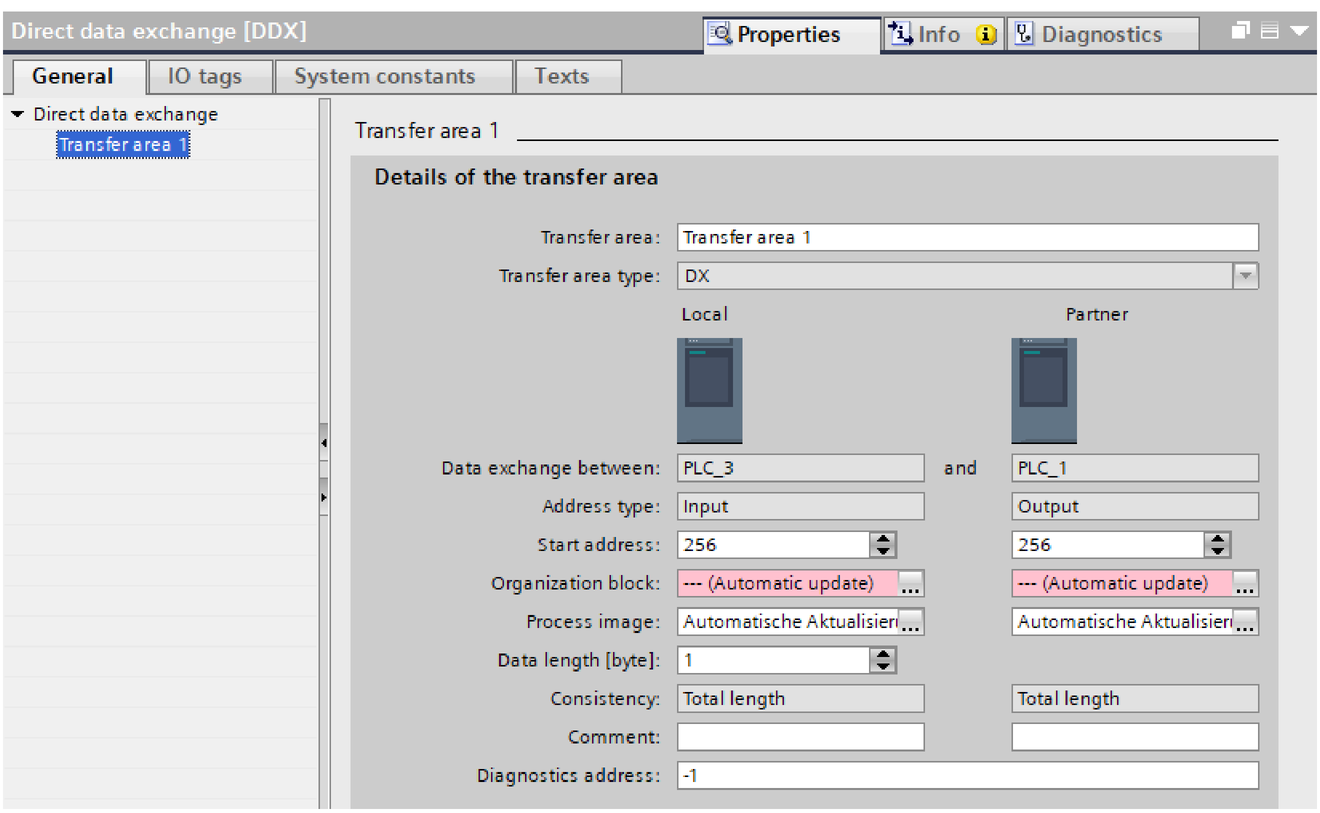


Figure 6-93 Properties of the transfer area

Table 6-6 Overview of the settings of the transfer area

Parameter	Local	Partner
Start address	Set the start address of the input or output transfer areas in the local CPU.	Set the start address of the input or output transfer areas in the partner CPU.
Organization block	Assign the transfer area to an isochronous mode interrupt OB or the "MC-Servo" OB.	Assign the transfer area to an isochronous mode interrupt OB or the "MC-Servo" OB.
Process image	Select a process image partition, e.g. PIP 1. If you have assigned "MC-Servo" as the organization block, STEP 7 automatically sets "PIP OB Servo" as the process image.	Select a process image partition, e.g. PIP 1. If you have assigned "MC-Servo" as the organization block, STEP 7 automatically sets "PIP OB Servo" as the process image.
Data length [bytes]	Set the size of the transfer area.	-

Downloading the configuration to the device

Rules:

- Download the configuration to all CPUs involved.
- If you make changes to the configuration of the direct data exchange, download these changes to all CPUs involved.

6.9 Device replacement without exchangeable medium

Definition

IO devices which have no slot for exchangeable medium (e.g. ET 200SP, ET 200MP) or IO devices which support the PROFINET functionality "Device replacement without exchangeable medium/PG" can be replaced without having an exchangeable medium with saved device names inserted and without having to assign the device name with the PG. The new IO device is given a device name by the IO controller and not by the exchangeable medium or the PG.

To assign the device name, the IO controller uses the configured topology and the neighborhood relations established from the IO devices.

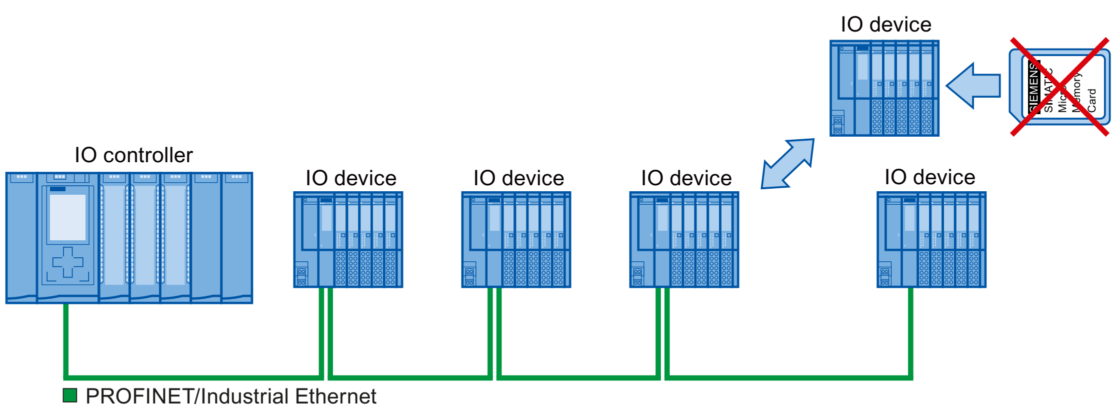


Figure 6-94 Device replacement without exchangeable medium

NOTE

When you exchange a device, make sure that the PROFINET cable is again inserted into the correct port, as it is configured in STEP 7.

Otherwise, the device names can be assigned incorrectly.

Advantages

With the Device replacement without exchangeable medium/PG PROFINET functionality, you can profit from the following advantages:

- After the replacement of the IO device, it automatically obtains its device name from the IO controller. You no longer have to assign device names with the PG or exchangeable medium.
- You can save on the storage medium for the IO device that replaces the removed one.
- Simple device name allocation with series machines that have the same configuration and set topology. It is no longer necessary to assign device names via exchangeable medium/PG.

Which devices support device replacement without exchangeable medium?

An overview of the devices that support the "Device replacement without exchangeable medium" function is provided in this FAQ (https://support.industry.siemens.com/cs/ww/en/view/36752540).

6.9.1 Device replacement without exchangeable medium/PG function

Neighborhood

Neighborhood is the physical relationship between two ports of adjacent PROFINET devices. A PROFINET device is connected in this case through one of its ports over a physical Ethernet line to a specific port of the second PROFINET device in close proximity (neighbors).

Both terminal devices, for example, IO controllers and IO devices with a port, and network components, for example, switches, IO controllers and IO devices with multiple ports, are considered PROFINET devices.

Failure and replacement of an IO device

The following example describes device replacement without exchangeable medium in the case of failure of an IO device.

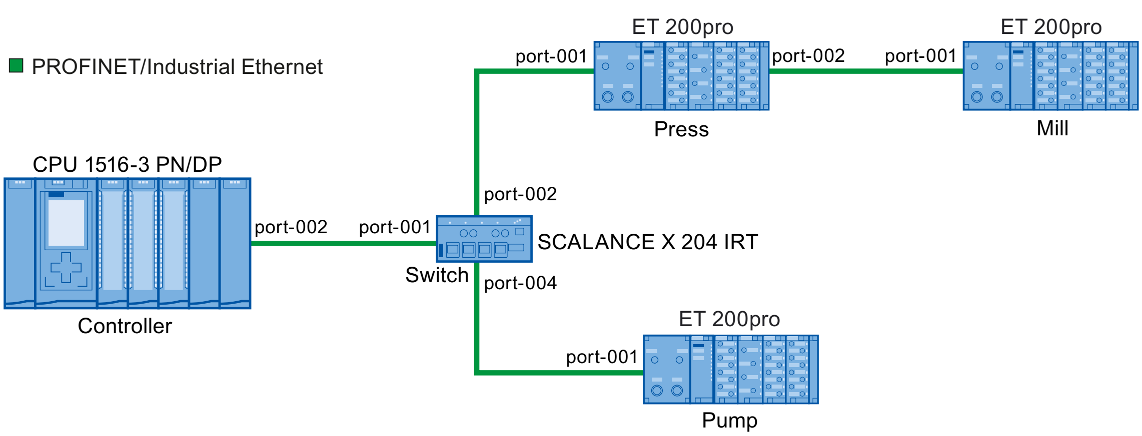


Figure 6-95 Example configuration of device replacement without exchangeable medium

For the device replacement, the following information is available to the IO controller:

PROFINET device	Device alias
Controller	"port-001.Switch"
Switch	"port-002.Controller", "port-001.Press", "port-001.Pump"
Press	"port-002.Switch", "port-001.Mill"
Mill	"port-002.Press"
Pump	"port-004.Switch"

The IO device with the device name "Mill" fails in this example:

Requirement

The PROFINET device replacing the removed one has no device name.

Principles of name assignment

The failure of the "Mill" IO device is considered here.

1. The IO controller queries the device name of the replaced IO device.
2. The IO controller detects that the IO device with the device alias "port-002.Press" does not have a device name.
3. The IO controller assigns the device name "Mill" to the replaced IO device via the device alias "port-002.Press" of the failed IO devices.

NOTE

If you insert a device with the Device replacement without exchangeable medium/PG PROFINET functionality at a different location than the configuration, a different device name is then assigned to the device.

Reset a wrongly inserted device to delivery state before you use it again.

6.9.2 Replacing an IO device without exchangeable medium

Introduction

Replacement of IO devices is sometimes frequently required in automation systems. The IO devices are generally assigned a device name by either inserting an exchangeable medium or via the programming device. The CPU identifies the IO device by using these device names.

Under certain circumstances, replacing an IO device can be done without inserting an exchangeable medium or without the PG.

Requirements

- The topology of the PROFINET IO system with the respective IO devices must be configured.
  By configuring the topology, the neighbor relationships of all the PROFINET devices located in the PROFINET IO system are announced to the PROFINET IO system or the IO controller. From the neighbor relationships specified by the set topology and the actual neighbor relationships established by the real PROFINET devices, the IO controller can identify the replaced IO device without a name and assign to it the configured name and the IP address and then again list it in the user data traffic.
- The affected IO devices in the automation system must support device replacement without exchangeable medium.
  If the individual IO devices in the automation system do not support device replacement without exchangeable medium, a corresponding alarm is output for the IO device.

NOTE

Use only new IO devices as replacements or restore configured IO devices to their delivery state.

Activating/deactivating device replacement without exchangeable medium

The "Device replacement without exchangeable medium" function is activated in the IO controller by default.

To deactivate device replacement without exchangeable medium, follow these steps:

1. In the device or network view of the STEP 7, select the PROFINET interface in the corresponding IO controller.
2. In the interface properties under "Advanced options > Interface options", clear the "Support device replacement without exchangeable medium" check box.

To reactivate device replacement without exchangeable medium, you must select the "Support device replacement without exchangeable medium" check box again.

6.9.3 Permit overwriting of PROFINET device name

With the option "Permit overwriting of PROFINET device names of all assigned IO devices", you can overwrite PROFINET device names of IO devices. This option reduces the effort during automatic commissioning, for example, when replacing a device.

Requirements

- IO controller supports the "Permit overwriting of device names of all assigned IO devices" option, for example CPU 1215C DC/DC/DC as of firmware version 4.0

How the "Permit overwriting of PROFINET device names of all assigned IO devices" option works

The IO controller (CPU) can overwrite the PROFINET device names of IO devices in the IO system when the "Permit overwriting of PROFINET device names of all assigned IO devices" option is enabled.

Multiple use IO systems can only be operated when this option is enabled. The IO controller checks prior to overwriting if the type of the IO device matches the configured type.

If the option is not selected, the IO controller cannot overwrite the device names of the IO devices. In this case, you must manually assign the PROFINET device name on the IO device if the PROFINET device name changes in the configuration or when a device is replaced, or delete the device names of the IO devices prior to an automatic commissioning.

Response during commissioning

Select the option "Permit overwriting of PROFINET device names of all assigned IO devices" only if the following requirements are fulfilled:

- All the configured IO devices are available.
- All IO devices are wired correctly in accordance with the topology configuration.
- No IO device is jumpered.

If configured IO devices are missing or are jumpered (partial commissioning), do not use the option.

You can also use the option for standard machine projects in which you adjust the configuration later via ReconfigIOSystem. Note that the valid configuration is always the one that was transferred to the IO controller in the control data record using the ReconfigIOSystem Mode:=2 instruction. As soon as you enable the reconfiguration with ReconfigIOSystem Mode:=3, the PROFINET device names are overwritten as defined in the data record.

> **⚠ CAUTION**
>
> **Error at partial commissioning**
>
> If device names are assigned incorrectly during partial commissioning or incorrect wiring, these have to be deleted manually after a correction of the wiring in order to attain correct assignment of the device names.

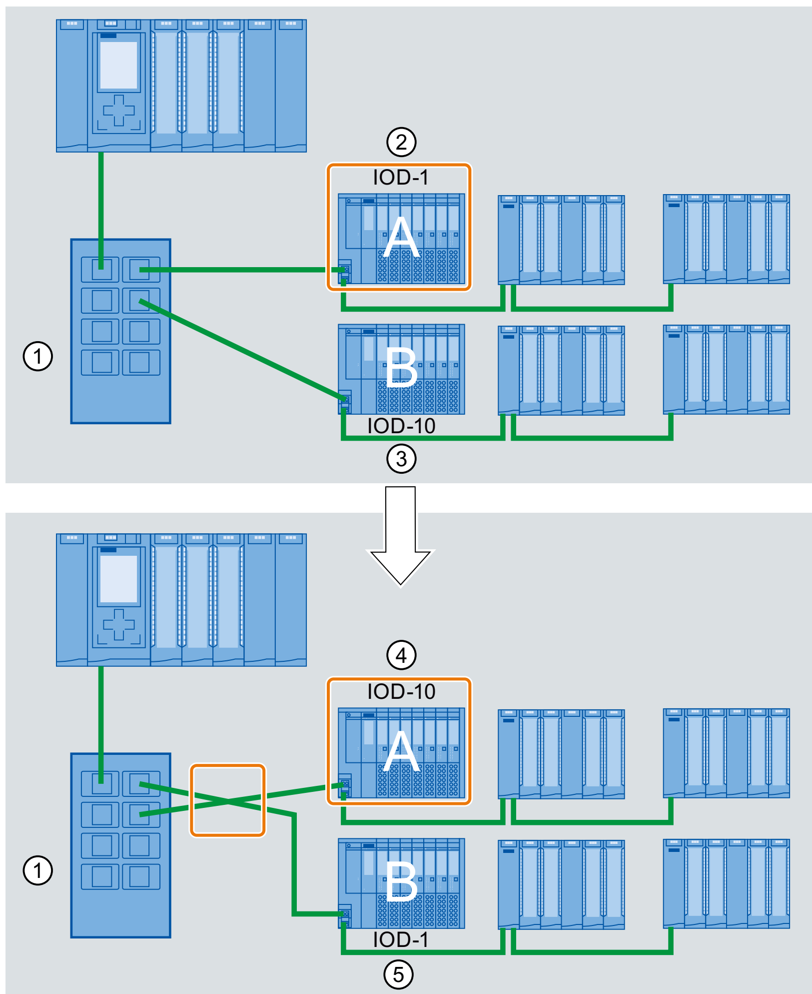
Behavior during operation

As soon as you replace a device, the new device is overwritten with the configured PROFINET device name.
The PROFINET device name is not overwritten when the MAC address of the IO device is already actively being used in the project.

> ⚠ **WARNING**
>
> **Wrong PROFINET device name**
>
> When the "Permit overwriting of PROFINET device names of all assigned IO devices" option is enabled, incorrectly connected devices can be assigned an incorrect PROFINET device name from the configuration.
>
> Depending on the connected I/O, there is risk of death, severe injuries or damage caused by malfunctions.
>
> To rule out any danger, check whether the suitable replacement device has been connected in case of a device replacement and whether the port interconnection matches the configured preset topology.

Typical source of danger

When replacing an IO device ("standard case"), it is almost guaranteed that the replaced device will be connected according to the configured port interconnection.
The figure below shows a scenario whereby the connections of two identical PROFINET cables are swapped at two switch ports. Because the IO controller assigns the device names according to the preset topology, the incorrect connection of the devices has serious effects on the naming.
Due to the control of different actuators, the plant could become hazardous in this case.



①	Switch with connected PROFINET cables
②	Device A, device name "IOD-1": controls motor 1
③	Device B, device name "IOD-10": controls motor 10
④	Device A controls motor 10!
⑤	Device B controls motor 1!

Procedure

Proceed as follows to change the "Permit overwriting of PROFINET device names of all assigned IO devices" option:

1. Select the PROFINET interface of the CPU for which you want to change the option in the network view or in the device view.
2. Select the area "Advanced options", section "Interface options".
3. Change the option.

6.10 Standard machine projects

Introduction

Standard machine projects are STEP 7 projects that use a set of innovative functions allowing simple configuration and commissioning of flexible automation solutions for standard machines or for machines with a modular structure.

A hardware configuration consisting of an IO controller and any number of connected IO devices represents a "PROFINET IO system master". This master is configured with a maximum configuration based on a template from which various options can be derived for different standard machines, for example with different configuration variants of the IO system.

Greater flexibility at all levels

Standard machine projects have the following central characteristics:

- Different variants of a standard machine can be loaded from precisely one project with an engineered maximum configuration (IO system options). The standard machine project covers all variants (options) of the IO system.
- An IO system option can be integrated in an existing network locally using simple tools.

Flexibility is provided in more ways than one:

- With suitable configuration, adaptation of the IP address parameters of the IO controller is possible locally using simple tools. This allows a standard machine to be integrated in different plants with little effort or to be included in a network several times.
  IO systems with this property are known as **multiple use IO systems**.
- With suitable configuration and programming, different setups of IO system options can be operated locally that differ in terms of the selection of IO devices used or in terms of the arrangement of the IO devices.
  Since the specific configuration of the IO system is controlled by the user program, this is known as **configuration control for IO systems**.
- Independently of the functions described above, with suitable configuration and programming, you can use different station options of central devices or distributed I/O devices in one project. The devices can be different in terms of the selection and arrangement of the modules.
  Since the concrete configuration of the station is controlled by the user program, this is also known as **configuration control at the device layer**.

Application examples

You can find application examples for configuration control at the device layer, configuration control for IO systems and multiple-use IO systems on the Internet (https://support.industry.siemens.com/cs/ww/en/view/29430270).

Additional information

For more information on multiple use IO systems, refer to section Multiple use IO systems (Page 251).
For more information on the configuration control for IO systems, refer to section Configuration control for IO systems (Page 259).
For more information on configuration control, refer to the system manual S7-1500, ET 200MP (http://support.automation.siemens.com/WW/view/en/59191792).

6.10.1 Multiple use IO systems

6.10.1.1 What you should know about multiple use IO systems

Multiple use automation solutions

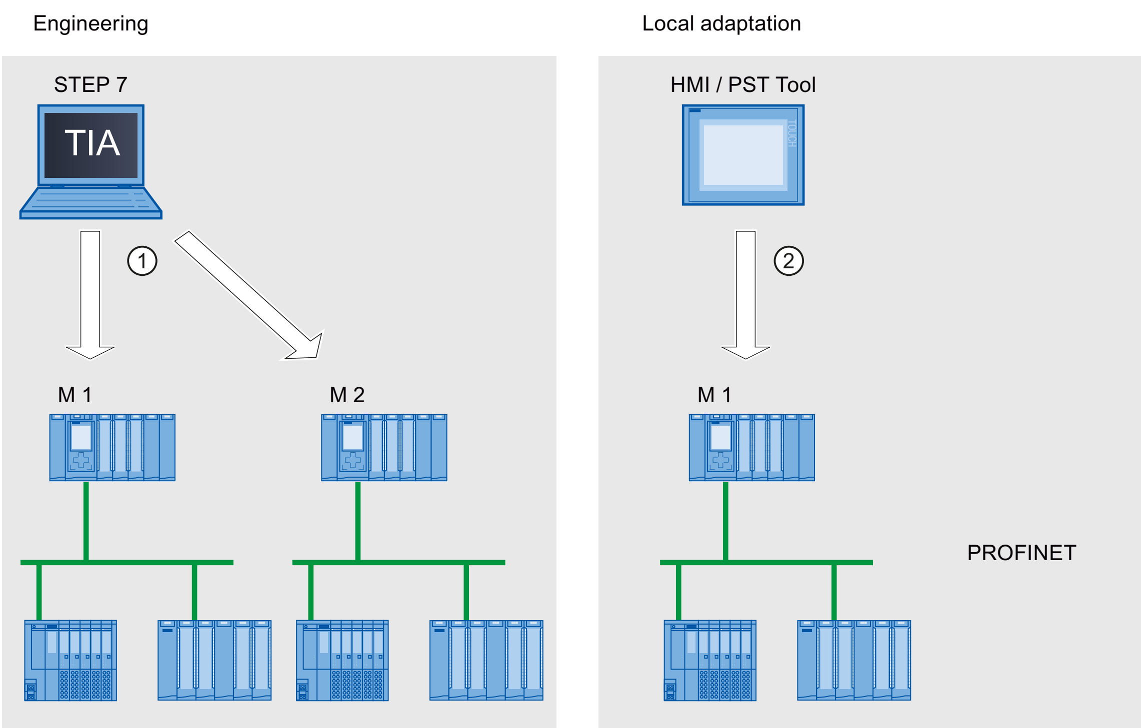
For a flexible reusable automation solution as is the case with series machines, the following use cases are typical:

- The machine (and therefore also the PROFINET IO system) is used more than once by the customers.
- The machine is used in different plants by various customers.

For this to be possible, the automation solution must meet the following requirements:

- A project (configuration and program) can be loaded on various machines of the same type without changes.
- Only a few easy adaptations need to be performed locally to integrate the machine into an existing network infrastructure.

The following figure shows how an automation solution with a multiple use IO system is loaded on different automation systems and then one automation system is adapted to the existing network infrastructure locally.



① Load configuration with multiple use IO system

② Set IP address and device name locally on the IO controller

Figure 6-96 "Multiple use IO system" principle

Principle

The automation components for a machine include a PROFINET IO system, consisting of an IO controller (PROFINET interface of a CPU) and the IO devices assigned to it.
With the "Multiple use IO system" setting for the IO system, you turn a STEP 7 project into a "Standard machine project".
The "Multiple use IO system" setting triggers various settings and checks of the configuration by STEP 7. The settings ensure that the IO system is self-contained and there are no dependencies on components outside the IO system.

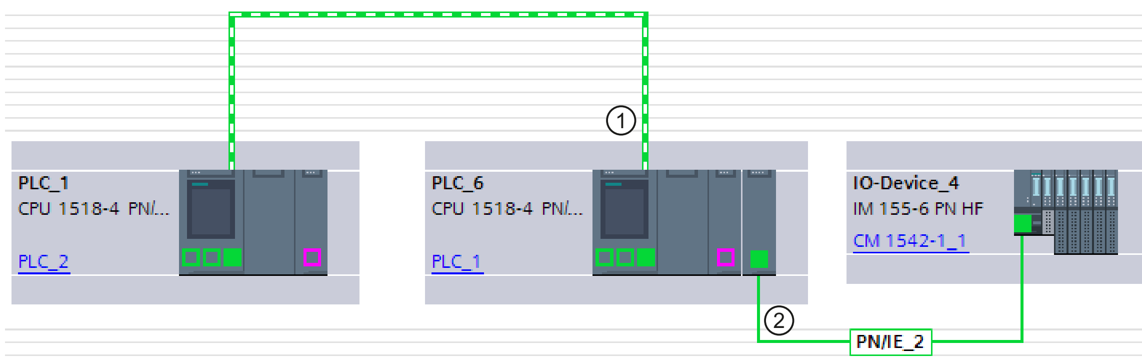
Requirements

- STEP 7 as of V13
- The IO controller supports the "Multiple use IO system" function, for example a CPU 1512SP-1 PN as of firmware version 1.6.

Rules

The following rules apply to a multiple use IO system:

- No IO device can be configured as shared device.
- The ports of the devices must be interconnected.
  Devices for which no port interconnection is configured, for example, interface module IM 154-6 IWLAN (ET 200pro PN), cannot be operated with STEP 7 V13 as IO devices on a multiple use IO system.
- If an IO device in a multiple use IO system is an I-device (CPU as an "intelligent" IO device):
  - If the I-device has a lower-level IO system, this I-device cannot be connected to the same PROFINET interface as the higher-level IO controller.
    Note: If the I-device is configured using a PROFINET GSD, STEP 7 cannot check for compliance with this rule. In this case, you are responsible for ensuring compliance with the rule.



① I-device in multiple use IO system. The PROFINET interface is operated as IO device. No other IO system is connected here.

② A lower-level IO system on the I-device is connected to another PROFINET interface.

Figure 6-97 Example configuration for "Multiple use IO system" with I-device

  - The PROFINET interface of the I-device must be set to "Parameter assignment of PN interface by higher-level controller".
- If MRP (Media Redundancy Protocol) is configured:
  - All IO devices on the multiple use IO system must belong to the same MRP domain.
- If IRT (Isochronous Real Time) is configured:
  - All IO devices in multiple use IO systems must belong to the same sync domain.
  - The sync domain must not include any other IO devices.
- IE/PB Links cannot be operated as an IO device in a multiple use IO system with STEP 7 V13.

Configuration

You specify whether or not a configuration can be used multiple times in the properties of the IO system.
All other parameter settings for the configured devices are then set automatically by STEP 7 and checked during compilation.

Boundary conditions

To prevent a standard machine project from having dependencies on other devices outside of the machine, observe the following:

- A standard machine project consists of an IO controller and the corresponding IO devices. You should therefore configure only one CPU as IO controller and the corresponding IO devices in the standard machine project.
- Do not use connection configured at both ends for the communication. Instead, use only a connection configured at one end or unspecified connections if necessary.

  Background: To configure the communication in a STEP 7 project, it is always possible to set the IP address parameters in the project. For multiple use IO systems, however, this strategy is not possible since the IP address parameters of the IO controller and the assigned IO devices are assigned locally. At the time of the configuration, the IP address parameters are therefore unknown.

  If you nevertheless want to configure communication with devices on PROFINET, for example with a central coordinator, you can only use communications mechanisms that allow dynamic assignment of the IP address parameters in the user program.

  Example: Open User Communication

  If, for example, the device is configured as an active end point (initiator of the connection), the IP address parameters can be stored, for example, in a data block. You then supply the data block with the currently valid IP address parameters during commissioning. For this dynamic type of IP address parameter assignment, there is no system support; in other words, if you change the configuration of the system, the IP address parameters are not automatically adapted.

  You will find a description of handling instructions for Open User Communication under this keyword in the STEP 7 online help.

6.10.1.2 Configuring multiple use IO systems

Requirements

- STEP 7 as of V13
- The IO controller supports the "Multiple use IO system" function, for example a CPU 1512SP-1 PN as of firmware version 1.6.

Procedure

The configuration of a series machine using an S7-1500-CPU as an example is described below.

To create a standard machine project, follow these steps:

1. Create a project.
2. Configure a CPU as the IO controller, for example a CPU 1518-4 PN/DP as of firmware version 1.5.
3. Configure the required IO devices and assign the IO devices to the IO controller.
4. Configure the port interconnection between the devices.
5. Select the IO system so that you can edit the properties in the inspector window.
6. Select the "Multiple use IO system" check box in the "General" area of the inspector window.

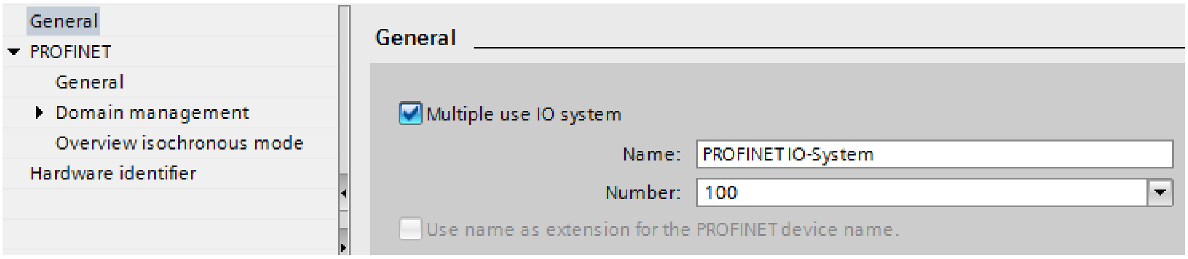


Figure 6-98 Activate "Multiple use IO system"

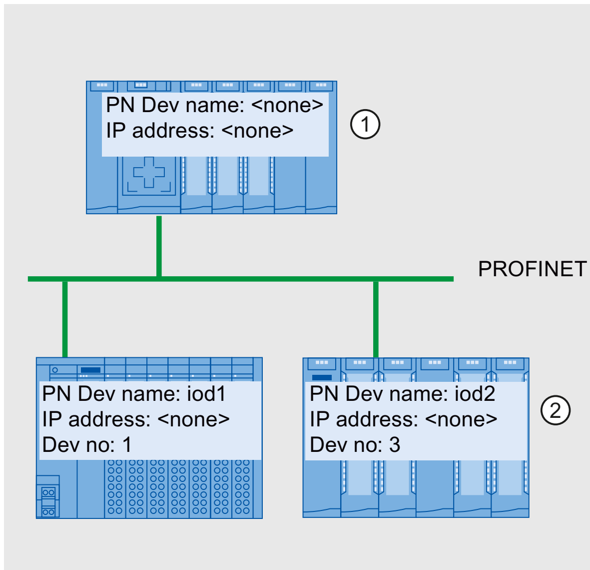
Result: The following settings are made by STEP 7:

- The device name of the IO controller (CPU) in the standard machine project is set to "PROFINET device name is set directly at the device". The IO controller (CPU) has no PROFINET device name initially.
- The IP protocol of the IO controller (CPU) is set to "IP address is set directly at the device". The CPU has no IP address initially.
- The "Support device replacement without exchangeable medium" option is selected automatically. This option enables automatic commissioning. A commissioning engineer does not have to assign device names to the IO devices locally. The IO controller assigns the device name and IP address to the IO devices based on the preset topology and other settings during startup.
- The device name of the IO devices is set to "Generate PROFINET device name automatically" (from the configured name of the IO device).
- The IP protocol of the IO devices is set to "IP address is set by the IO controller during runtime". The IO devices have no IP address initially. If an IO device is not a typical distributed I/O system (e.g., ET 200 systems), but rather another device such as an HMI device, change the option to "IP address is set directly at the device"; see below.
- The device number for the IO devices is automatically assigned and is used locally for making the IP address unique.

The option "Permit overwriting of PROFINET device name" must be selected (CPU parameters, properties of the PROFINET interface, Ethernet addresses area) so that the IO controller can adapt the device name later at the operator.

This option is disabled by default.

The following figure shows the above-described settings for the IP address and PROFINET device name.



① After the configuration is loaded from the standard machine project, the IO controller has no device name and no IP address.

② Following loading, the IO devices have a device name and a device number but no IP address.

Figure 6-99 Settings for the IP address and the PROFINET device name

How an IO device obtains an IP address locally

Below, you will find an explanation of the "IP address is set by the IO controller during runtime" and "IP address is set directly at the device" options, which can generally be configured for a multiple use IO system.

If you have set the "Multiple use IO system" option for the IO system, STEP 7 automatically sets the "IP address is set by the IO controller during runtime" option for the IO devices.

In this case, the IO controller assigns the IO device an IP address that results from the locally assigned IP address of the IO controller (see next section). This option is appropriate if the IO device is a field device, e.g., ET 200MP, ET 200SP, ET 200AL, or another distributed I/O system.

If the IO device is not a "standard" field device, for example, an HMI device for a Windows operating system, the "IP address is set by the IO controller during runtime" option described above does not work. In this case, choose the "IP address is set directly at the device" option. You must then assign the IP address to the device locally and take steps to ensure that this address is suitable for the IP addresses of the other IO devices and the IP address of the IO controller.

6.10.1.3 Adapt multiple use IO systems locally

A few steps are needed to adapt the machine that was loaded with the standard machine project.

Only the device name and IP address of the IO controller must be adapted locally. The device names and IP addresses of the IO devices result from these adaptations. In this example, the effects of local settings are described for two specific machine modules.

The local settings are possible, for example, with the CPU display and commissioning tools such as Primary Setup Tool (PST) or PRONETA. You do not need a programming device with STEP 7 to make these settings, even though it is possible to do so.

Requirements

- The machine was loaded with a standard machine project (see Configuring multiple use IO systems (Page 254)).
- The display is ready for operation and the desired tool for assigning the IP address and device name is available (e.g., PST Tool, STEP 7).
- The ports of the IO controller and IO devices are interconnected according to the configuration.

Procedure

Observe the boundary conditions and instructions for commissioning an S7-1500. For more information on commissioning an S7-1500 CPU, refer to the system manual S7-1500, ET 200MP (http://support.automation.siemens.com/WW/view/en/59191792).

To adapt a standard machine locally, follow these steps:

1. Integrate the machine into the network.
2. Connect the device for assigning the IP address and device name to the CPU, for example a PG/PC with the appropriate software.
3. Assign the desired device name and IP address to the IO controller.
4. Start up the CPU.

The IO controller then assigns the adapted PROFINET device name and a unique IP address to the IO devices.

The following rules apply to the assignment:

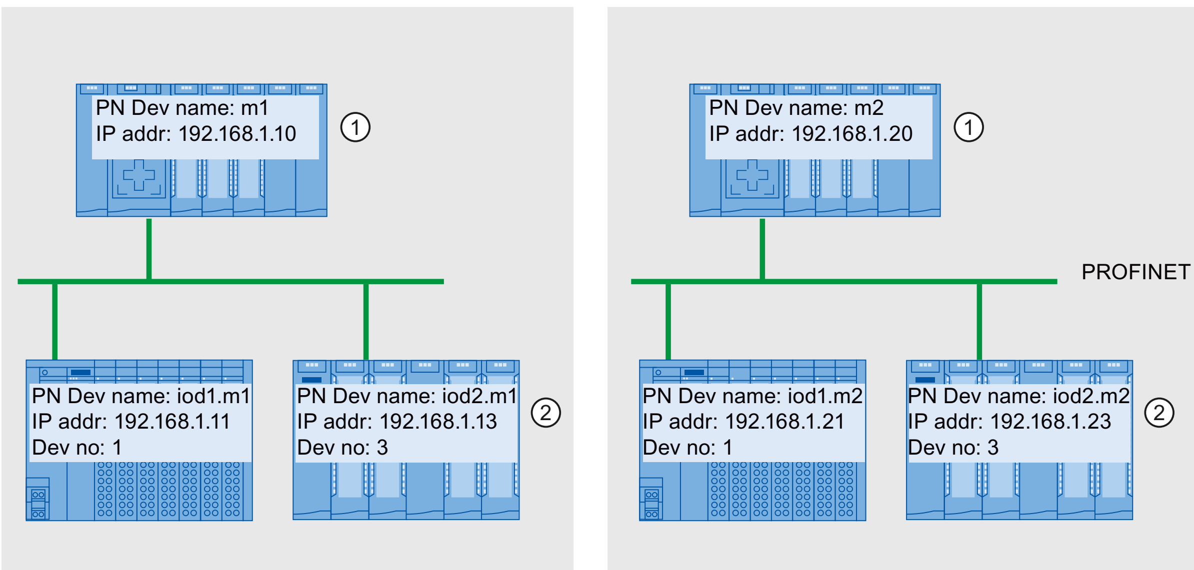
- The device names of the IO devices are formed by chaining together the following name components, separated by a period:
  <configured name of the IO device from the standard machine project>.<name of the associated IO controller set on the device>
- The IP addresses of the IO devices result from the locally configured IP address of the associated IO controller and the device number (sum).

NOTE

Make sure that duplicate IP addresses cannot be created on the subnet during the assignment. The IO controller does not assign a new IP address in this case.

In the following figure, the device name "m1" and the IP address 192.168.1.10 have been assigned to the IO controller of the first machine.

The device name "m2" and the IP address 192.168.1.20 has been assigned for the second machine.
Refer to the figure for the resulting device names and IP addresses.



① Set device name and IP address on the IO controller

② After startup, the IO devices have an updated device name (<configured device name>.<device name of IO controller>) and an adapted IP address (= <IP address of IO controller> + <device number>)

Figure 6-100 Example of the assignment of IP addresses and device names at "Multiple use IO system"

See also

Configuring multiple use IO systems (Page 254)

6.10.2 Configuration control for IO systems

6.10.2.1 Information about configuration control of IO systems

Configuration control of IO systems makes it possible to generate several concrete versions of a standard machine from a standard machine project.

You are given the flexibility to vary the configuration of an IO system for a specific application as long as the real configuration can be derived from the set configuration. The configured configuration therefore represents the superset of all real configurations that can be derived from it.

The following figure shows an example of how two IO systems with a different number of IO devices arise from one standard machine project.

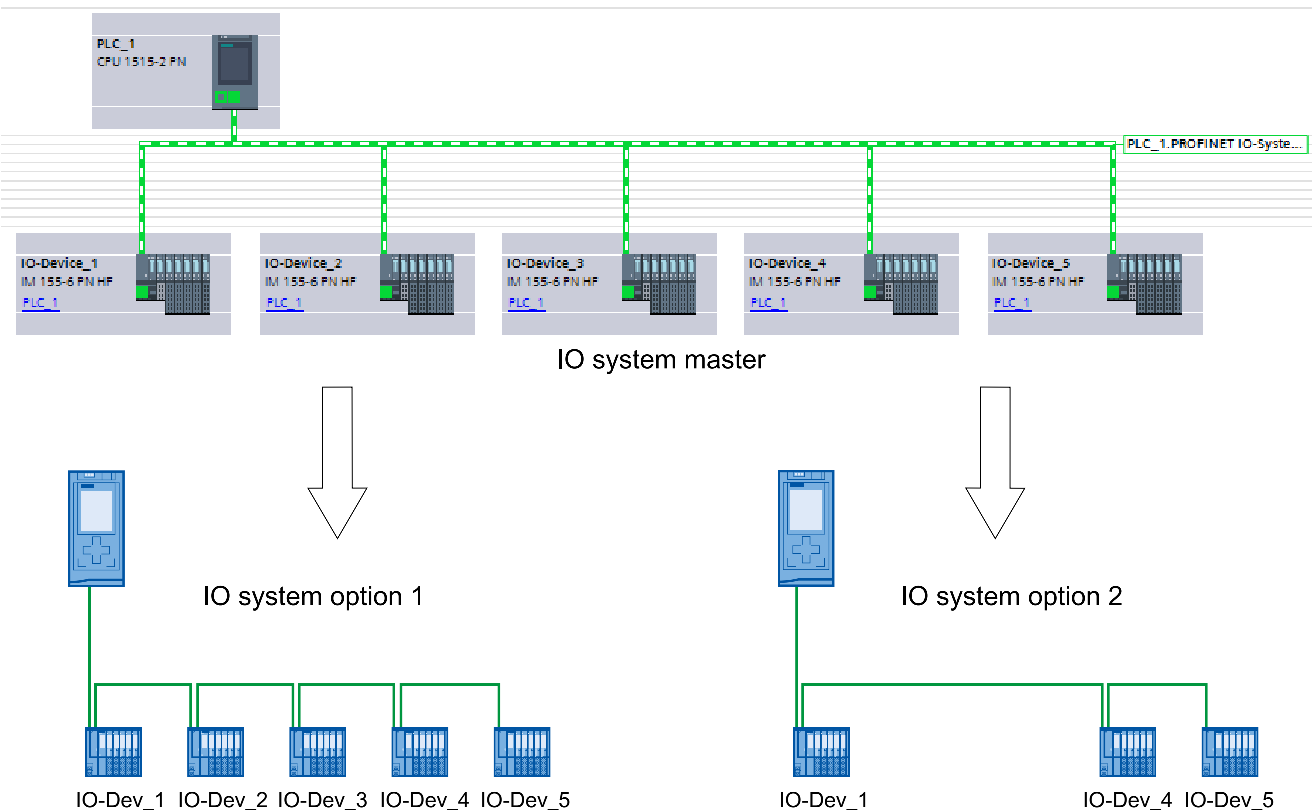


Figure 6-101 Example of configuration control for PROFINET IO systems

In the following sections, you find a description of how to configure and program a PROFINET IO system to commission, for example, a standard machine on-site without using configuration software.

Concept

The principle of configuration control is already known at the device level for the flexible use of submodules/modules ("option handling"). Different configurations can be derived from one engineering project both for central as well as for distributed I/O.

With S7-1500 CPUs as of firmware version V1.7, this principle can also be applied at the IO system level. You have the option of omitting, adding or changing the order of stations (IO devices) of a PROFINET IO system in a specific plant.

Configuration control for devices and configuration control for IO systems can be combined; the functions are independent of each other.

It is possible to operate variants deviating from a maximum configuration of an IO system. In a standard machine project, you can prepare a kit of IO devices which can be flexibly customized for various configurations using configuration control.

The following variations are available:

- Variation of the number of IO devices involved

  You include optional IO devices for the configuration control in the configuration by transferring a suitable data record with the required configuration in the user program.
- Variation of the order of IO devices involved

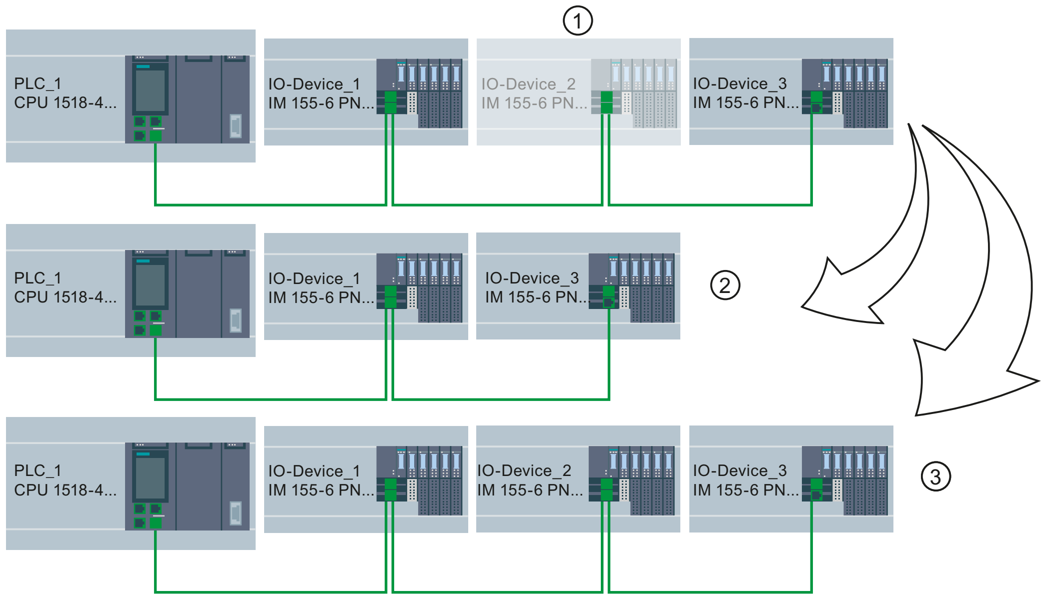
  You adapt the port interconnection of the IO devices to the topology being used by transferring a suitable data record with the required topology in the user program.

The following figure shows how you serve two different configurations with an IO device marked as optional in the network view of STEP 7.

- Configuration without the optional IO device:

  In this case, you use the instruction "ReconfigIOSystem" to transfer a data record to the PROFINET interface containing the information that no optional IO device is to be included in the configuration.
- Configuration with the optional IO device:

  In this case, you use the instruction "ReconfigIOSystem" to transfer a data record to the PROFINET interface adding the optional IO device to the configuration.



① Determined through parameter assignment: IO Device_2 is optional IO device

② Configuration without the optional IO device

③ Configuration with the optional IO device

Figure 6-102 Example configuration with "Optional IO device" and the associated configuration options

Summary: The procedure in principle

The following phases are distinguished when it comes to the implementation of a standard machine concept:

1. Engineering phase: Creating a standard machine project and loading into specific machine or plant:
   - Completely configuring all IO devices (options) ever required in a specific machine or plant
   - Configuring as optional those IO devices that will be omitted in specific machines or plants
   - Preparing user program (see Configuring flexible order of IO devices (Page 269)) with the possibility of selecting on-site the actually existing configuration via switch or HMI device
2. Commissioning phase: Preparing specific machine or plant for operation:
   - Integrating machine or plant in the on-site network (see Adapt multiple use IO systems locally (Page 257))
   - Selecting the currently existing configuration of the IO system via configured option

6.10.2.2 Configuring IO devices as optional

Requirements

- IO controller supports configuration control for IO systems, for example CPU 1516-3 PN/DP as of firmware version 1.7
- STEP 7 V13 SP1 or higher
- The rules (Page 274) for the establishment and operation of a standard machine project have been considered.

Port interconnection

As of STEP 7 V15.1, port interconnection is not necessary for optional IO devices.
A port interconnection between the devices of the IO system that you want to customize with the user program is mandatory in the following cases.

- You have configured IRT.
- You have configured MRP.
- You are using STEP 7 <= V15.

Procedure

To configure an IO device as optional IO device, proceed as follows:

1. Create a project.
2. Configure an S7-1500 CPU firmware version V1.7 or higher as IO controller.
3. Configure the required IO devices and assign the IO devices to the IO controller.
4. Select the IO device you want to mark as optional.
5. Select the area "PROFINET interface [X1]" > Advanced options".
6. Enable the "Optional IO device" option.

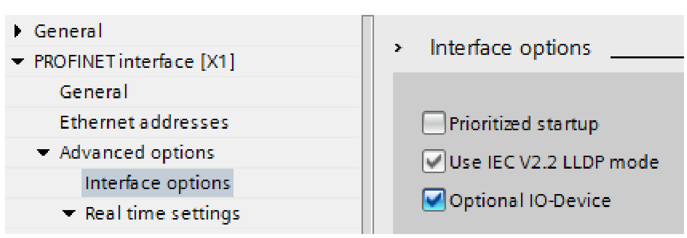


Figure 6-103 Configure the IO device as an optional IO device

7. Repeat steps 4 to 6 for all IO devices to be configured as optional.
8. Load the configuration onto the CPU.

Result: Once this configuration is loaded, the system behavior is as follows:

- The CPU is prepared for the configuration control of the IO system.
- All IO devices are disabled.

- Irrespective of whether you customize the configuration with the user program (adding optional IO devices) or make no changes to the loaded configuration: You must call the instruction "ReconfigIOSystem" in the user program and notify the current configuration to the system!
  The system will not be operational without calling the instruction "ReconfigIOSystem".
  For further information on the proceeding see Enabling optional IO devices in the program (Page 263).

Fast parameter assignment in the "IO communication" table

You can also specify whether or not an IO device is optional in the "IO communication" tab.

In an additional "Optional IO device" column, a selectable check box is available for each IO device that indicates whether or not an IO device is optional. Here, you can adjust the setting centrally.

6.10.2.3 Enabling optional IO devices in the program

Requirements

- IO controller supports configuration control for IO systems, for example CPU 1516-3 PN/DP as of firmware version 1.7
- STEP 7 V13 SP1 or higher
- At least one IO device was configured as optional IO device.
- The rules (Page 274) for the establishment and operation of a standard machine project have been considered.

Procedure

Observe the information on and rules for commissioning in the documentation for SIMATIC S7-1500, for the ET 200SP CPUs and for the CPU 1516pro-2 PN.

The following description of the proceeding only includes steps required to understand the program-controlled activation of an optional IO device.

To activate or deactivate IO devices, follow these steps:

1. Create a data record "CTRLREC" for the instruction "ReconfigIOSystem". You can find information on the structure of the data record in the STEP 7 online help.
2. Call the instruction "ReconfigIOSystem" and select MODE 1 to deactivate all IO devices.
   If you set the CPU to STOP or POWER OFF state in order to modify the plant in this status (for example to add an optional IO device), explicit deactivation using "ReconfigIOSystem" with mode 1 is not necessary. In this case, i.e. following a STOP-RUN transition and following a POWER-OFF > POWER-ON transition, all IO devices are deactivated automatically.
3. When you have brought the plant to a safe status that allows restructuring without any danger:
   Put the plant together according to your intended application. Add the required optional IO devices at the points at which you planned this in the configuration (observe the order!) or remove optional IO devices that you no longer require.
4. Network the IO devices.

5. Startup the S7-1500 system and call again the instruction "ReconfigIOSystem". Select MODE 2 to transfer the data record CTRLREC.
6. Following successful transfer of the data record, call again the instruction "ReconfigIOSystem". Select MODE 3 to activate all IO devices forming part of the current configuration.
   Result: The CPU **activates** the following IO devices:
   – All IO devices that you have not set as optional IO devices.
   – All optional IO devices listed in the control data record (CTRLREC).
   The following IO devices remain **disabled**:
   – Docking units (IO devices changing during operation).
   – Optional IO devices that are not listed in the control data record.

NOTE

Call the instruction "ReconfigIOSystem" for all values of the parameter MODE with the same control data record (CTRLREC)!

If you use different data records for the values of the MODE parameter, this results in an inconsistent customization of the configuration and thus to rather incomprehensible error messages.

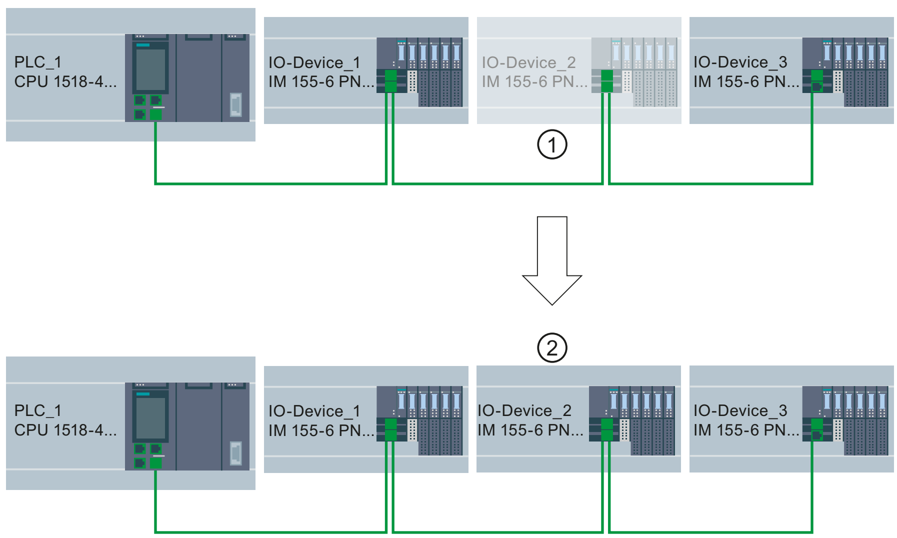
Example: Data record structure for the activation of an IO device

The IO device "IO-Device_2" is to be activated as the only IO device in the user program.

To do this, you only require the hardware identifier of "IO-Device_2".

Recommendation: Use the system constants of the hardware identifiers instead of the absolute values as shown in this example. With this procedure, the content of the DB is not influenced by changes to the hardware identifiers as the result of changes to the configuration.

The data record is to be stored in a data block and to be transmitted to the PROFINET interface of the IO controller in the user program using the instruction "ReconfigIOSystem".



① IO device_2 is configured as optional IO device.

② Once the data record is transmitted and the configuration is activated using the instruction "ReconfigIOSystem" IO device_2 is included in the configuration and participates in the data exchange with the IO controller.

Figure 6-104 Example: Activating an optional IO device

Creating data block

In this example, the control data record is created in a data block. The data block is structured as follows:

Line 2: Array definition: Array of type Word with 4 elements. Array of Word is permitted as the data type.

Line 3: Version of the data record (currently: V1.0).

Line 4: Number of optional IO devices to be activated (here: 1).

Line 5: List of the hardware identifiers of the IO devices, inserted here as system constants.

Line 6: Number of port interconnections that are set in the user program (here: 0).

Line 7: Additional data records (optional)

myCTRLREC

	Name	Data type	Start value	Comment
1	▼ Static			
2	▼ ArrMachineConfig0	Array[0..3] of Word		Array for one optional IO device
3	ArrMachineConfig0[0]	Word	16#0100	
4	ArrMachineConfig0[1]	Word	16#0001	
5	ArrMachineConfig0[2]	Word	"IO-Device_2~IODevice"	
6	ArrMachineConfig0[3]	Word	16#0000	
7	▶ ArrMachineConfig1	Array[0..8] of Word		

Figure 6-105 Data block with control records

Parameter MODE of instruction "ReconfigIOSystem"

You control how the "ReconfigIOSystem" instruction works with the MODE parameter.

The following values are possible for the MODE input parameter:

MODE	Description
1	All IO devices of the IO system can be disabled by calling the instruction with MODE 1. The "ReconfigIOSystem" instruction uses the "D_ACT_DP" instruction internally. "ReconfigIOSystem" returns errors that are detected by D_ACT_DP in the following output parameters: • STATUS (error code) • ERR_INFO (hardware identifier of the IO device causing the error). In STATUS and ERR_INFO, the CPU enters the last determined error/HW identifier and in so doing overwrites an existing error code. For this reason, additional errors can be present besides the entered error.
2	For controlling the actual configuration of the IO system, the instruction transfers the data record to the PROFINET interface, which is addressed with LADDR block parameter (HW identifier of the PROFINET interface).
3	All non-optional IO devices in the IO system and optional IO devices that are listed in the control data record CTRLREC are enabled. The optional IO devices that are **not** listed in the CTRLREC data record remain disabled. If IO devices that are part of docking units (alternating IO devices in operation) are listed in the CTRLREC control data record, the PN IO system reacts as follows: • IO devices of the docking units remain disabled when ReconfigIOSystem is called with MODE 3. This reaction corresponds to the reaction of a configuration without configuration-controlled IO devices. IO devices of docking units are disabled by default and must be enabled in the user program.

MODE	Description
	For more information on docking units, refer to the section Docking systems (Page 278).

Rules for the call sequence of "ReconfigIOSystem"

- Always supply the instruction "ReconfigIOSystem" with the same control data record (CTRLREC input parameter)!
- Call sequence following POWER OFF -> POWER ON transition:
  - ReconfigIOSystem call with MODE 1 (optional).
  - ReconfigIOSystem call with MODE 2 (mandatory, even without previous reconfiguration!).
  - ReconfigIOSystem call with MODE 3 (mandatory).
- Call sequence following STOP > RUN transition:
  - ReconfigIOSystem call with MODE 1 (optional).
  - ReconfigIOSystem call with MODE 2 (mandatory, even when configuration was modified in STOP state). Otherwise not required).
  - ReconfigIOSystem call with MODE 3 (mandatory).
- Call sequence for reconfiguration in RUN state:
  - ReconfigIOSystem call with MODE 1 (mandatory).
  - ReconfigIOSystem call with MODE 2 (mandatory).
  - ReconfigIOSystem call with MODE 3 (mandatory).

Explanations and recommendations concerning the rules

- If you do not list an IO device to be configured as optional IO device in the control data record or data block resp. this IO device does not form part of the configuration and does not take part in data exchange with the CPU.
- If you do not activate any optional IO device at all and work with the loaded configuration without reconfiguration, you still have to follow the proceeding described in the above section and transmit the control data record to the CPU.

  The control data record has the simple structure with the following tags:
  - Version (High Byte =1, Low Byte = 0)
  - Number of optional devices to be activated = 0
  - Number of port interconnections that are set in the user program = 0
- Following a STOP > RUN transition and following a POWER-OFF > POWER-ON transition, all IO devices are deactivated automatically. For this reason, no ReconfigIOSystem call with MODE 1 is required for configuration control to function properly.

  If you use your project as a universally valid sample for programming the configuration control, we still recommend to perform the ReconfigIOSystem call with MODE 1 prior to any reconfiguration. This way, the sample can also be used for reconfigurations in RUN mode.

- Commissioning extensive I/O systems (more than 8 optional IO devices) while using IRT at the same time:
  To keep the startup times short when activating the optional IO devices (ReconfigIOSystem, mode 3), note the following tip: Check the device numbers of the IO devices. The device numbers should follow the topological interconnection starting at the IO controller in ascending order. The further an IO device is from the IO controller topologically, in other words the more IO devices there are between the IO controller and the IO device in question, the higher the device number should be.
  You set the device numbers in the "Ethernet addresses - PROFINET" area in the Inspector window with the PROFINET interface selected.
  Example of the assignment of device numbers with a linear topology:

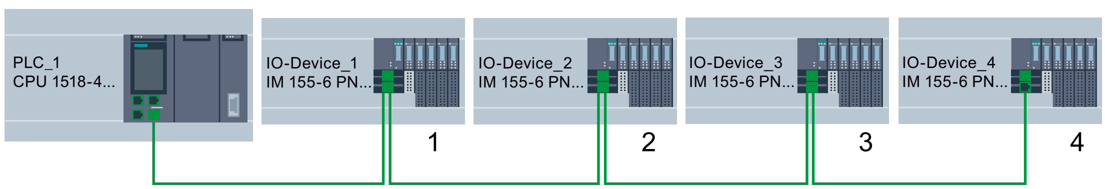


Figure 6-106 Example: Device numbers in a line topology

- The CPU processes the instruction "ReconfigIOSystem" to transfer the control data record asynchronously.
  For this reason, you must call "ReconfigIOSystem" in a loop repeatedly when calling the instruction in the startup program until the output parameter "BUSY" or "DONE" indicate that the data record has been transferred.
  Tip: To program the loop, use the SCL programming language with the instruction REPEAT ... UNTIL.

```
REPEAT
    "ReconfigIOSystem"(REQ := "start_config_ctrl",
                       MODE := 1,
                       LADDR := 64,
                       CTRLREC := "myCTRLREC".ArrMachineConfig0,
                       DONE => "conf_DONE",
                       BUSY => "conf_BUSY",
                       ERROR => "conf_ERROR",
                       STATUS => "conf_STATUS");
UNTIL NOT "conf_BUSY"
END_REPEAT;
```

Additional information

For information on the basic structure of the data record and on using the instruction "ReconfigIOSystem" see the STEP 7 online help.

See also

Configuring IO devices as optional (Page 262)

6.10.2.4 Configuring flexible order of IO devices

The following section shows how you can create the conditions required to change the order of IO devices in a PROFINET IO system.
This function is also supported with optional IO devices. For simplicity, a maximum configuration without optional IO devices is shown below.

Concept

A typical application for a standard machine project consists of composing an entire plant from a set of various plant units which only differ with respect to the different arrangement of the units, e.g. in the case of transport systems. Each plant unit consists of a functional unit of mechanics (rails or conveyor belts) and electrics (power supply, IO device with IO modules, sensors, actuators, motors, PROFINET port for data exchange with central control ...).
The following figure shows how, simply by exchanging two rail segments, a new transport system is created that is adapted with an upstream points to the local conditions.

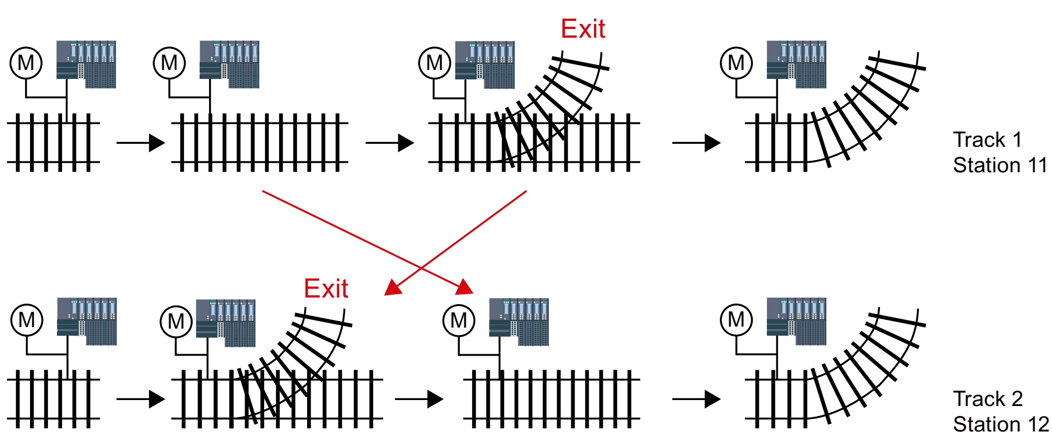


Figure 6-107 Example: Changing the arrangement of IO devices

From the automation viewpoint, no change in the project is required for the flexible adjustment of the PROFINET configuration.
The order of the IO devices is determined by the port interconnection. For each IO device, you define in the port properties the partner port and thus the neighboring device connected at the respective local port. If the partner port is to be defined by the user program, the option "Partner set by user program" is to be selected as partner port.

The figure below shows the initial configuration of the transport system shown above, which is to permit the order of the connected IO devices to be changed via the user program. In the example, the order of IO-Device_2 and IO-Device_3 is to be controlled via the user program.

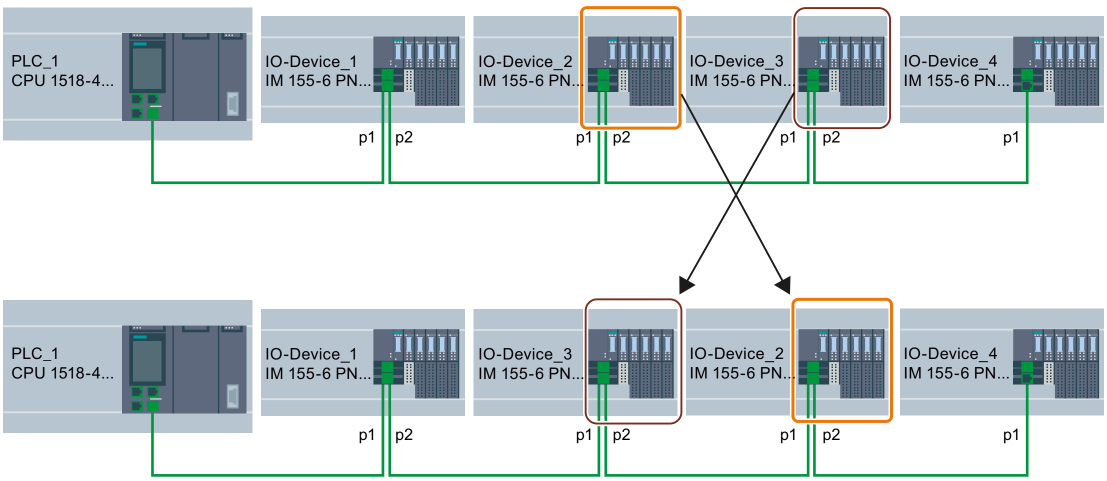


Figure 6-108 Example configuration: Configuring flexible order of IO devices

To determine how the partner port settings are to be selected, you must note for each device and each port of a device which partner can be interconnected.

- If the partner is always the same in the different configurations provided, you select the partner port for this partner.
- If the partners vary in the different configurations, you select "Setting partner by user program".

For the example in the figure above, the following port settings result:

Device	Local port	Partner port setting	Explanation
PLC_1	p1	p1 (IO device_1)	Partner of PLC_1 at port 1 is IO device_1 (always)
IO device_1	p1	p1 (PLC_1)	Partner of IO device_1 at port 1 is PLC_1 (always)
IO device_1	p2	Partner is set by user program	Partner of IO device_1 at port 2 is either IO device_2 or IO device_3 => Setting partner by user program
IO device_2	p1	Partner is set by user program	Partner of IO device_2 at port 1 is either IO device_1 or IO device_3 => Setting partner by user program
IO device_2	p2	Partner is set by user program	Partner of IO device_2 at port 2 is either IO device_3 or IO device_4 => Setting partner by user program
IO device_3	p1	Partner is set by user program	Partner of IO device_3 at port 1 is either IO device_2 or IO device_1 => Setting partner by user program

Device	Local port	Partner port setting	Explanation
IO device_3	p2	Partner is set by user program	Partner of IO device_3 at port 2 is either IO device_4 or IO device_2 => Setting partner by user program
IO device_4	p1	Partner is set by user program	Partner of IO device_4 at port 1 is either IO device_3 or IO device_2 => Setting partner by user program
IO device_4	p2	Any partner	No partner planned at port 2

Requirement

- IO controller supports configuration control for IO systems, for example CPU 1515-2 PN as of firmware version 1.7
- STEP 7 V13 SP1 or higher
- The rules (Page 274) for the establishment and operation of a standard machine project have been considered.

Procedure

To set the partner port for a program controlled interconnection, proceed as follows:

1. Select the PROFINET interface of the device (IO controller or IO device) whose port you want to set.
2. In the properties of the PROFINET interface, select the area "Port interconnection" (Extended options > Port [...] > Port interconnection).
3. From the drop-down list, select "Setting partner by user program" as partner port.
4. Repeat steps 1 to 3 for each port to be interconnected via the user program.

See also

Customizing arrangement of IO devices in the program (Page 272)

6.10.2.5 Customizing arrangement of IO devices in the program

Requirements

- IO controller supports configuration control for IO systems, for example CPU 1516pro-2 PN as of firmware version 1.7 as an IO controller
- STEP 7 V13 SP1
- At least one partner port was configured as "Partner set by user program".
- The rules (Page 274) for the establishment and operation of a standard machine project have been considered.

Procedure

The proceedings corresponds to the proceeding for activating optional IO devices.

Only the structure of the data record must be extended for the program-controlled assignment of the ports. The extension is described in the following sections.

Example: Data record structure for the assignment of partner ports

For the data record structure, you need the HW identifications of the ports.

The data record is to be stored in a data block and to be transmitted to the PROFINET interface of the IO controller in the user program using the instruction "ReconfigIOSystem".

As the input parameter RECORD of the instruction "ReconfigIOSystem" is of the VARIANT data type, you first have to create a data type for the data block.

In the following sections, you find a description of the structure of the PLC data type as well as of the structure of the data block based on this type.

Selecting derived configuration

For the following selected configuration it is shown below what the data record must look like so that the IO devices are interconnected in the planned order by the user program.

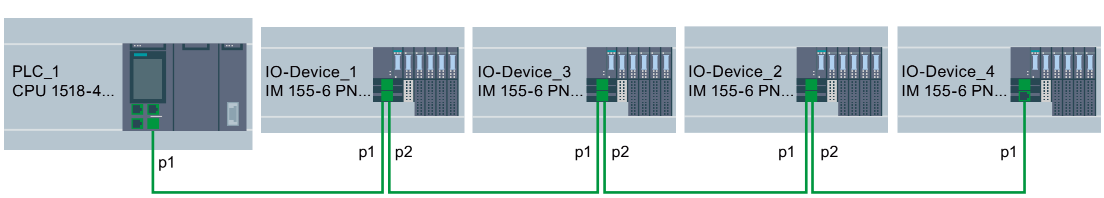


Figure 6-109 Example configuration: Customizing arrangement of IO devices in the user program

This example is based on the flexible configuration from the previous section (Page 269) with the settings for the respective partner ports described there.

The partner ports in the specific derived configuration have been defined so that it is possible to name the HW identifications of the ports involved.

The following table only contains those devices whose ports can be defined by the user program. Only these devices are relevant for the data record structure.

Device	Local port	Partner port setting	Partner port of the selected configuration
IO device_1	p2 = Port 2 HW identifier: 251	Partner is set by user program	Port 1 of IO device_3 HW identifier: 261
IO device_2	p1 = Port 1 HW identifier: 281	Partner is set by user program	Port 2 of IO device_3 HW identifier: 291
IO device_2	p2 = Port 2 HW identifier: 311	Partner is set by user program	Port 1 of IO device_4 HW identifier: 321

Creating data block

For the derived configuration, the structure of the data block "DB-IO-SYSTEM-Port-Interconnections" is explained as an example.
This data block is used when calling the instruction "ReconfigIOSystem" at input parameter "CTRLREC".
Instead of the absolute values for the hardware identifiers of the ports, the system constants of the hardware identifiers are used here.
The data block is structured as follows:
Line 2: Declaration of an Array of Word (only this data type is possible).
Line 3: Version of the control data record: V1.0.
Line 4: Number of optional IO devices: 0.
Line 5: Number of specified port interconnections: 3.
Line 6: Port interconnection 1, local port.
Line 7: Port interconnection 1, partner port.
Line 8: Port interconnection 2, local port.
Line 9: Port interconnection 2, partner port
Line 10: Port interconnection 3, local port.
Line 11: Port interconnection 3, partner port.

IO-SYSTEM-Port-Interconnections

	Name	Datentyp	Startwert	Kommentar
1	Static			
2	ArrMachineConfig	Array[0..8] of Word		data record for port interconn settings
3	ArrMachineConfig[0]	Word	16#0100	
4	ArrMachineConfig[1]	Word	16#0000	
5	ArrMachineConfig[2]	Word	16#0003	
6	ArrMachineConfig[3]	Word	"IO_device_1~PROFINET_interface~Port_2"	
7	ArrMachineConfig[4]	Word	"IO_device_3~PROFINET_interface~Port_1"	
8	ArrMachineConfig[5]	Word	"IO_device_2~PROFINET_interface~Port_1"	
9	ArrMachineConfig[6]	Word	"IO_device_3~PROFINET_interface~Port_2"	
10	ArrMachineConfig[7]	Word	"IO_device_2~PROFINET_interface~Port_2"	
11	ArrMachineConfig[8]	Word	"IO_device_4~PROFINET_interface~Port_1"	

Figure 6-110 Data block with data record for port interconnections

Interconnection not listed in data block

If the partner port was configured as "Setting partner by user program" in the port properties and this port is not listed in the data record or data block resp., then the CPU sets this port to the setting "any partner". If no data record is transmitted at all, the CPU sets this "any partner" setting for all program-controlled assignments.

Additional information

For information on the basic structure of the data record and on using the instruction "ReconfigIOSystem" see the STEP 7 online help.

6.10.2.6 System behavior and rules

Below, you find a description of how an IO system whose configuration is controlled by the user program behaves in operation.
In addition, rules and restrictions are listed here which must be considered when configuring the maximum structure of the configuration in a standard machine project.

System behavior

- System diagnostics:
  If an optional IO device is deactivated, the IO device is displayed as "deactivated" from the system diagnostics viewpoint (online view or Online & Diagnostics).
- Topology view:
  Offline view: As configured. No interconnection is shown for ports with partner ports configured as "Setting partner by user program".
  Online view: Ports and interconnections with deactivated IO devices are shown in a different shade of green as error-free ports and interconnections of activated IO devices.
- Representation in the Web server:
  The names of devices are shown as configured (Properties > General > Project information).
  The assigned PROFINET device name for the CPU is shown on the "Communication" website, at the "Parameter" tab.
  IP address parameters: Currently assigned IP address parameters are shown on the site "Module state".
  Topology: The current topology resulting from any customizations via user program is shown in the Web server. IO devices configured as optional are shown as "deactivated" IO devices in the Web server.

Rules

The rules for standard machine projects as described here (Page 251) apply.
For configuration-controlled IO systems, the following additional rules apply:

- When configuring MRP (Media Redundancy Protocol):
  The ports configured as ring ports must not be interlinked via user program.
  However, devices with ring ports (devices of an MRP domain) can be optional IO devices.
- When configuring docking stations (= IO devices changing during operation):
  Neither the docking station nor the first IO device of a docking unit may be optional IO devices.
  The ports of the docking units must not be interlinked via user program.
- When configuring IRT:
  The order of synchronized IO devices ("IRT devices") must be defined by the configuration and must not be changed in the different variants of a standard machine. For this reason, the ports of the IRT devices must not be interlinked via user program.
  However, you have the possibility to configure IRT devices primarily as optional IO devices.
  You also have the option to interconnect, by user program, RT devices that are, for example, separated from this line by a switchport (see figure).

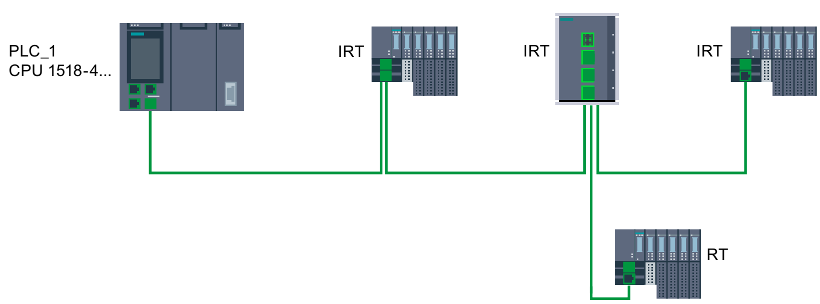


Figure 6-111 Example: Configuration control for IO systems with separated RT device

6.11 Saving energy with PROFIenergy

Saving energy with PROFIenergy

PROFIenergy is a PROFINET-based data interface for switching off consumers centrally and with full coordination during pause times regardless of the manufacturer or device type. Through this, the process should only be provided with the energy that is absolutely required. The majority of the energy is saved by the process; the PROFINET device itself only contributes a few watts of savings potential.

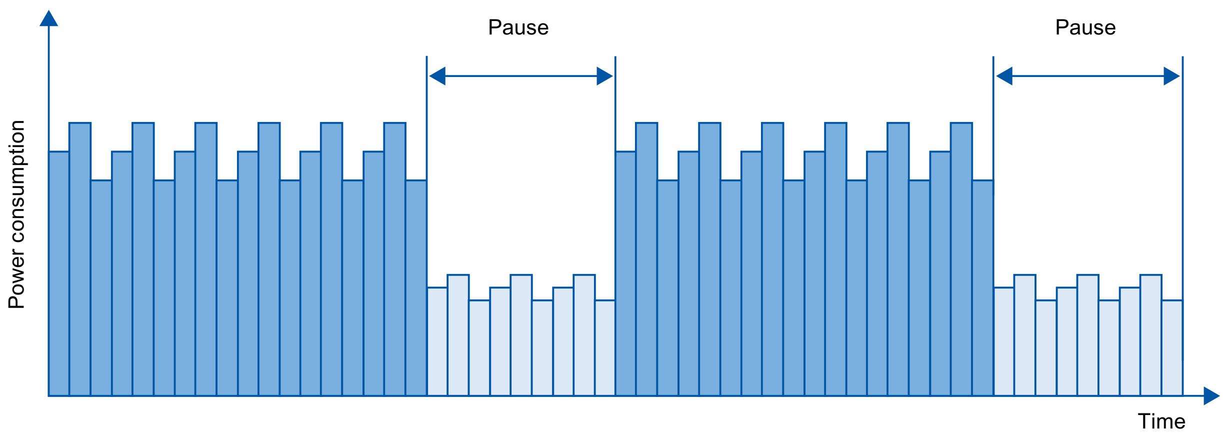


Figure 6-112 Energy savings during pauses with PROFIenergy

Basic information

In order to use the PROFIenergy functionality, the existing IO controller has to be "upgraded" to a so-called "PROFIenergy controller" by means of a function block in the PROFINET IO system and at least one PROFIenergy-capable IO device ("PROFIenergy device") has to exist. The PROFIenergy commands (for example to start or end a pause) are sent by the "PROFIenergy controller" to the individual "PROFIenergy devices". Each "PROFIenergy device" decides individually how it responds to the PROFIenergy command (response is device- and manufacturer-specific).

No additional hardware is needed; the PROFIenergy commands are directly interpreted by the PROFINET devices.

Principle of operation

At the beginning and end of pauses, the system manager enables or disables the pause function of the system; then the IO controller sends the PROFIenergy command "Start_Pause"/"End_Pause" to the PROFINET devices. The device interprets the contents of the PROFIenergy command and switches off or back on.

Through other PROFIenergy functions, device information can be accessed during pauses. The user can use this information in order to optimally time the transfer of the "Start_Pause"/"End_Pause" command.

PROFIenergy instructions for IO controllers

Two instructions are needed for controlling and monitoring the PROFIenergy functions.

The instruction PE_START_END allows you to easily activate and deactivate the idle state of PROFINET devices. This occurs by means of an incoming edge or outgoing edge. The instruction PE_START_END provides a simple interface for implementing the PROFIenergy commands Start_Pause and End_Pause.

The instruction PE_CMD allows you to transmit all PROFIenergy commands, including Start_Pause and End_Pause. The other commands can be used, for example, to query the current status of the PROFINET device or the behavior during the pauses. The instruction PE_CMD is a convenient means for handling all PROFIenergy functions.

PROFIenergy instruction for I-devices

The instruction PE_I_DEV allows you to also implement PROFIenergy on I-devices. The instruction receives PROFIenergy commands on the I-device and forwards these to the user program for execution. After executing the command, the user program calls the PE_I_DEV instruction again in order to send the acknowledgment to the IO controller. For these replies, each command offers you a helper instruction that supplies the reply data to the instruction.

The instructions can be found in the "Instructions" task card of the STEP 7 program editor.

Configuration and programming

The functions can be comfortably integrated into existing systems. No configuration is required for the use of PROFIenergy. However, amendments to the user program are required:

- Before the "Start_Pause" command, you must ensure that your system is brought into a condition that is suitable for a pause.
- A sequential control system for the beginning of the pause of the devices and for the punctual restarting of the device on break must be programmed (depending on the required startup times that the respective PROFINET device demands).
- The error messages of the PE_CMD instruction must be evaluated, and the required reaction must be programmed (for example, cancellation or continuation of further commands on lower-level PROFINET devices).

NOTE

In the case of the ET 200S distributed I/O system, you have to configure the application of PROFIenergy in STEP 7. You configure PROFIenergy by selecting the "Use energy saving for this potential group" check box in the PM-E DC24V/8A RO power module.

If you want to use PROFIenergy for an I-device, you have to configure this in STEP 7. Information about configuring PROFIenergy for an I-device is available in the section Configuring PROFIenergy with I-devices (Page 140).

Application examples

- SIMATIC S7 library for simple configuration of PROFIenergy.
  The application example is available here
  (https://support.industry.siemens.com/cs/ww/en/view/109478388).
- Application guide for implementation of shutdown concepts with PROFIenergy.
  The application example is available here
  (https://support.industry.siemens.com/cs/ww/en/view/96837137).
- Saving energy with SIMATIC S7 PROFIenergy with I-device
  The application example is available here
  (https://support.industry.siemens.com/cs/ww/en/view/41986454).

6.12 Docking systems

Using alternating IO devices ("alternating partners") during operation in a docking station

The following figure shows an automation cell with a docking station and several docking units.

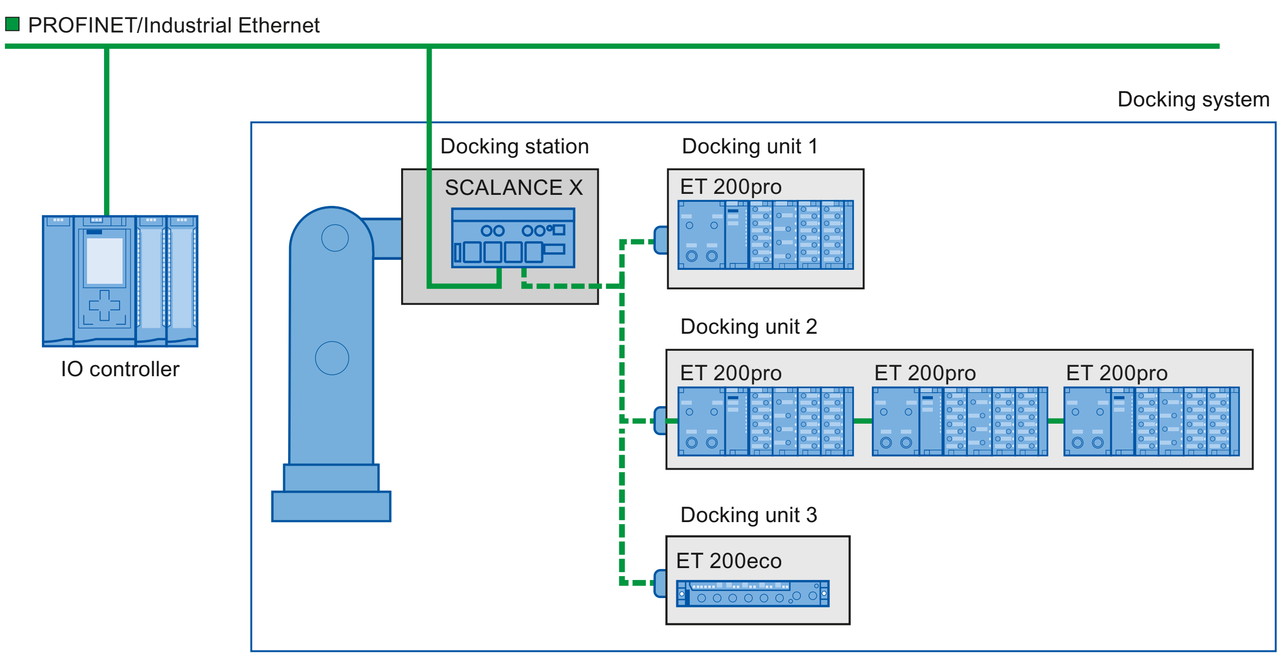


Figure 6-113 Alternating IO devices (partner ports) in a docking system

Area of application for alternating IO devices during operation

You can use the PROFINET function "Alternating IO devices during operation" ("alternating partners"), for the tool changeover for robots, for example. Typical tools include:

- Welding guns
- Positioning tools for manufacturing parts

Requirements for interconnecting alternating partner ports during operation

You can interconnect IO devices with alternating partner ports during operation in the following situations:

- The alternating IO device (docking unit) has no IRT communication configured.
- The PROFINET interface is connected to the Ethernet subnet.
- The PROFINET devices support topology configuration.
- The IO controller, the alternating IO devices (docking unit) and the switch (docking station) on which the alternating IO devices are to be operated support this feature.

NOTE

Unique IP address

Each docking unit of a docking system must be configured with a unique IP address in a shared project and operated on the same IO controller as all other docking units of the system.

Applicative conditions

The following points should be observed when implementing a docking system with alternating IO devices during operation:

- The IO devices of all docking units must be deactivated by default in the configuration.
- At any time, only one docking unit can be active, i.e., only the IO devices of one docking unit can be activated. All IO devices of other docking units must be deactivated or become deactivated before the IO devices of a docking unit can be activated. You activate an IO device with the "D_ACT_DP" instruction.
- A physical connection to this docking unit and its IO devices must be created in order to activate a docking unit. The IO devices are then switched on (power on). At the same time, all the IO devices of this docking unit must be activated in the user program with the "D_ACT_DP" instruction.

NOTE

Automatic deactivation in "Startup" mode of the CPU

If the CPU is in "Startup" mode, IO devices of a docking system that alternate during operation are deactivated automatically.

- After the feedback "IO device activated", you can access the IO device by means of direct I/O access.

- Call the "D_ACT_DP" instruction to activate and deactivate the IO device as close as possible to the start of the OB 1 cycle.

NOTE

Number of alternating IO devices during operation ("alternating partner port") - number of docking units

If you wish to achieve the shortest possible tool changeover times, you must observe the following points that are dependent on the CPU or the CP that is being used:

– Only those IO devices that have been configured with the PROFINET function "Prioritized startup" can start up in an optimized fashion. The number of IO devices with configuration for this PROFINET function is restricted.

– Only a limited number of IO devices can be activated at the same time. This number depends on the available "D_ACT_DP" resources. A docking unit should not contain more than the corresponding number of IO devices. If more IO devices are operated in a docking unit, the IO devices must be activated one after the other, which takes correspondingly longer.

Example: An S7-CPU 1516-3 PN/DP can operate a maximum of 32 IO devices with prioritized startup and can simultaneously activate 8 IO devices by means of "D_ACT_DP".

Therefore, for a scheduled optimum use, a docking unit should include no more than 8 IO devices and no more than 32 IO devices should be used in all the alternating docking units.

6.12.1 Configuring docking systems

Configuring docking systems

The possible connections to the individual IO devices must be configured in STEP 7.

Procedure in STEP 7

1. Configure your system as usual, but do not yet configure the topological interconnections of the individual PROFINET devices.
2. Navigate to the "Topology view" tab.
3. Select the port which you want to operate with alternating partners during operation.
4. Navigate to the "Properties" tab of the inspector window and select "Port interconnection" in the area navigation.
5. Under "Partner port", select the "Alternative partners" option.
6. Select the desired partner ports: To do this, click "<Add alternative partners... and choose a partner port. Repeat this process until all required partner ports are connected.

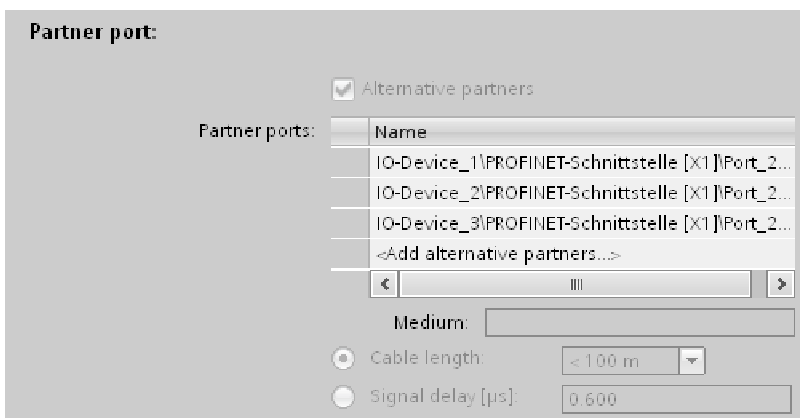


Figure 6-114 Configuring docking systems in STEP 7

Tip: You can also use drag-and-drop operation to connect alternative partner ports in the topology view.

Result

You have interconnected the respective port with one or more alternating IO devices. The connections with the individual alternating partner ports during operation are shown in the topology view by a green, dashed line.

Multiple IO devices as a docking unit ("Alternating partner port")

A docking unit may also consist of several IO devices connected in series. If you use IO devices connected in series as a docking unit, ensure that the topology of the IO devices is configured. A docking unit consisting of two IO devices connected in series (Tool_3_1 and Tool_3_2) is shown at the bottom right in the figure below.

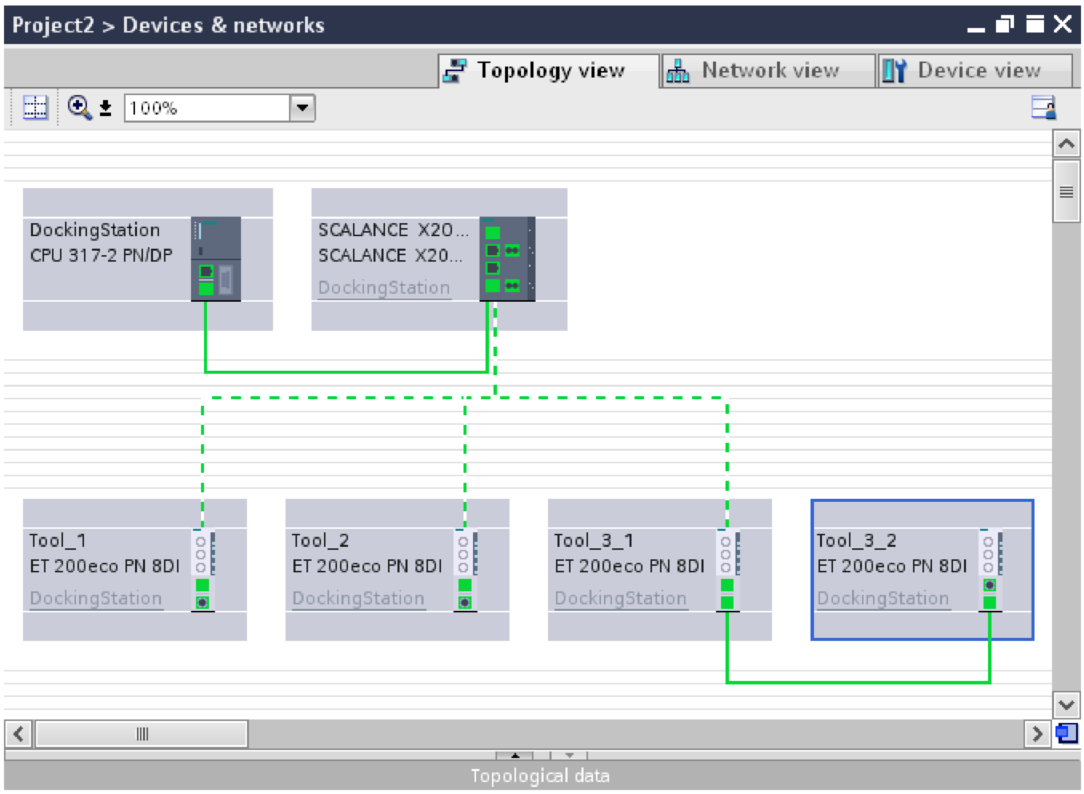


Figure 6-115 Docking system in the topology view of STEP 7

Restriction in the interconnection

The interconnection with a partner port is not possible in the following cases:

- The partner port does not have a suitable type of cable. In this case, a media convertor from the catalog must be inserted.
- The partner port is blocked (deactivated).
- The two ports that are to be interconnected belong to the same interface (it is only possible to interconnect the ports from different interfaces in a station).
- The two ports that are to be interconnected belong to different Ethernet subnets.

Deleting interconnections

Select the port of the alternating partner and remove the interconnection with the "Remove" button.

6.13 Accelerating startup

6.13.1 Options for accelerating the startup of IO devices

Reducing the startup time of IO devices

The time needed for the startup of IO devices depends on various factors and can be reduced in different ways.

You can achieve a significant reduction in the startup time using the "Prioritized startup" PROFINET function.

To further reduce the startup times, use the following measures in addition to the "Prioritized startup":

- Optimize the port settings
- Optimize the cabling of the ports
- Perform measures in the user program (for docking systems only)

These measures will accelerate the startup of IO devices even without "Prioritized startup". However, you can only achieve the fastest startup times of about 500 ms by combining all the measures with "Prioritized startup".

NOTE

Startup time up to 8 s

In the following situation, despite prioritized startup, a startup time of up to 8 s may occur:

On a docking point, multiple physical IO devices dock as an IO device with identical device names and identical IP configurations (for example, docking point for automatic transport system).

Dependencies

The length of the startup time for an IO device (distributed I/O) with the "Prioritized startup" PROFINET function depends on the following factors:

- IO devices (distributed I/O)
- IO structure of the IO device (distributed I/O)
- Modules of the IO device (distributed I/O)
- IO controller
- Switch
- Port setting
- Cabling
- Configured RT class of the IO device

NOTE

Startup time and RT class of the IO device

An IO device with IRT communication requires longer than an IO device with RT communication for the accelerated startup.

The longer startup for IRT is due to the necessity of synchronizing the IO device before the communication can be set up.

> **NOTE**
> **IWLAN and prioritized startup**
> PROFINET devices which are connected via access points to PROFINET IO do not support the "Prioritized startup" PROFINET function.

6.13.2 Prioritized startup

Definition

Prioritized startup denotes the PROFINET functionality for accelerating the startup of IO devices in a PROFINET IO system with RT and IRT communication. It reduces the time that the correspondingly configured IO devices require in order to return to cyclic user data exchange in the following cases:

- After the power supply has returned
- After a station has returned
- After activation of IO devices

> **WARNING**
> **Data exchange despite multiple use IP addresses/device names in PROFINET IO system**
> To achieve higher ramp-up times, the IO controller checks the uniqueness of device name of IP address parallel to the device ramp-up. In the case of incorrect or double assignment of device name or IP address, a short-time data exchange is possible until the IO controller reacts to the error. In this time the IO controller can exchange IO data with the false device.
> During commissioning, ensure that no IP addresses / device names are assigned multiple times in the PROFINET IO system.

Advantages

The PROFINET functionality "prioritized startup" enables PROFINET IO applications in which machine parts or tools and their IO devices have been permanently replaced. Waiting times of several seconds between the scheduled processes of the restart are reduced to a minimum by this optimization. This accelerates the production process with alternating IO devices (PROFINET functionality "docking systems"), e.g. in tool changer applications, and enables a greater throughput in production.

The PROFINET functionality "prioritized startup" also offers a considerable increase in performance for applications where a quick startup time of the IO devices after "power on" or after station failure / station return is required, or when activating IO devices.

Area of application

You can use prioritized startup, for example, for changing tools for robots in the automotive industry. Typical tools are, for example:

- Welding guns
- Positioning tools for the manufacture of car body parts

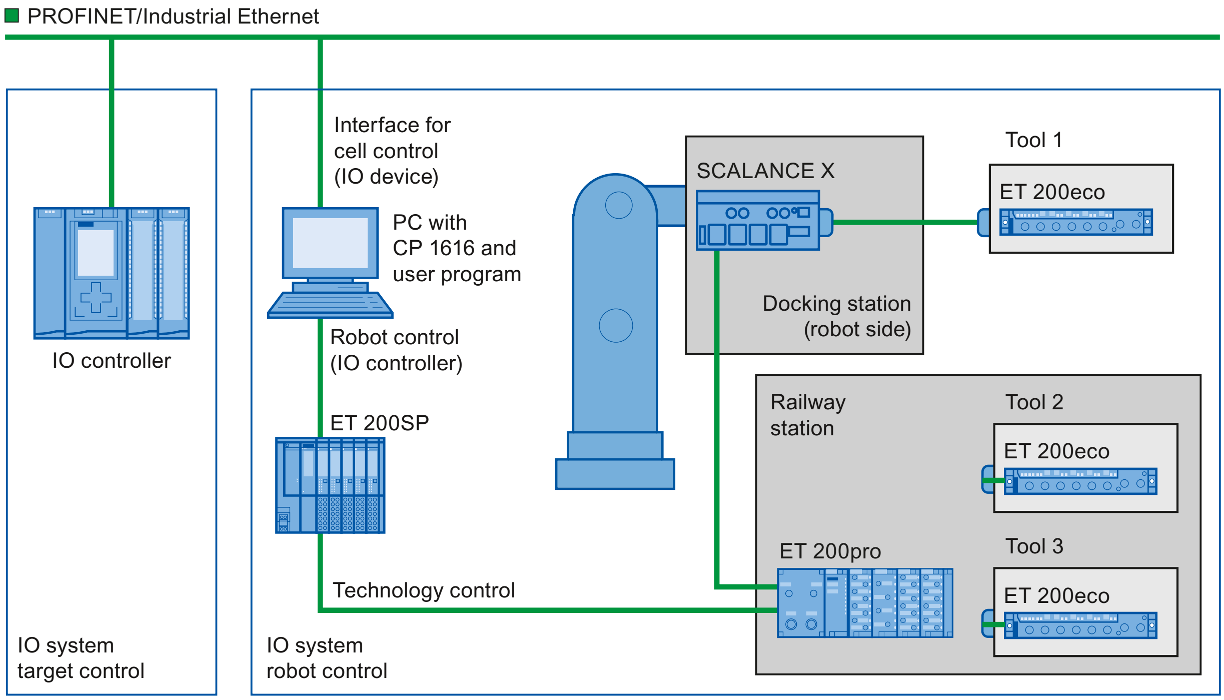


Figure 6-116 Example configuration of a tool changer: Tools 1-3 configured with "Prioritized startup".

6.13.3 Configuring prioritized startup

Requirements

You can enable the "Prioritized startup" PROFINET function for the IO devices (distributed I/O) only in the following cases:

- The IO controller used can prioritize selected IO devices during startup.
- The IO device used supports prioritization.

Procedure

1. Select the IO device in the network view or device view for which you wish to accelerate startup.
2. Open the IO device properties in the Inspector window.
3. Select "PROFINET interface > Advanced options > Interface options".
4. Select the "Prioritized startup" check box.

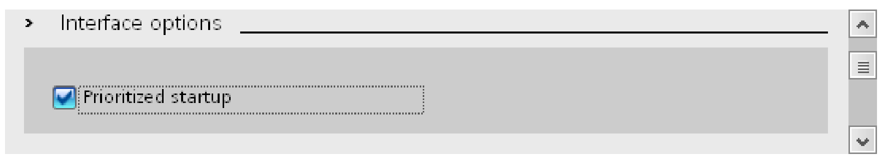


Figure 6-117 Configuring prioritized startup in STEP 7

5. Download the configuration to the IO controller.

NOTE

Prioritized startup after a startup for the first time

A prioritized startup of the IO device is always available to you after the first configuration of this IO device in the very first startup of the PROFINET IO system. Even in the case of spare parts or a reset to factory settings, the first startup is a standard startup for the respective configured IO devices.

NOTE

Number of IO devices (distributed I/O) with prioritized startup

You can only start up a maximum number of IO devices with the "Prioritized startup" PROFINET functionality within one PROFINET IO system. This maximum number depends on the IO controller used.

6.13.4 Optimize the port settings

Optimizing port settings on the IO device and IO controller

The transfer medium and the duplex option are checked during startup of the IO device in the case of copper cabling.
These checks require time, but with specific presets of these options you can save the time the check requires. Make certain that the settings made correspond to the actual conditions (using the correct cables).

Optimizing port settings for accelerated startup

To optimize port settings for accelerated startup, follow these steps:

1. Select the ports of the IO controller or the partner port of the corresponding IO device.
2. Navigate to "Port options > Connection" in the Inspector window. Select the setting "TP 100 Mbps full duplex" under "Transmission rate/duplex"
3. Clear the "Enable autonegotiation" check box.

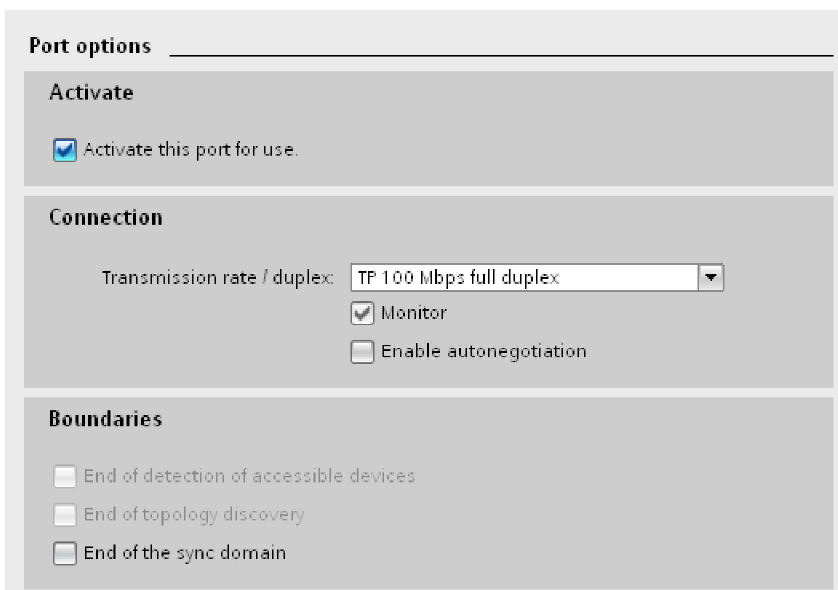


Figure 6-118 Optimizing port settings for accelerated startup in STEP 7

These settings are automatically applied during topology configuration for the partner port.

6.13.5 Optimize the cabling of the ports

Requirements

You have made the following settings for the port in question to reduce the startup time of the IO device:

- Fixed transmission rate
- Autonegotiation deactivated

The time for negotiating the transmission rate during startup is saved.

If you have disabled autonegotiation, you must observe the cabling rules.

Properties of ports

PROFINET devices have the following two types of ports:

Type of port	PROFINET devices
Switch port with crossed pin assignment	For IO devices: Port 2 For S7-CPUs with two ports: Ports 1 and 2
Terminal device port with normal pin assign-ment	For IO devices: Port 1 For S7-CPUs with one port: Port 1

Crossed pin assignment means that the pin assignment for the ports for sending and receiving between the respective PROFINET devices is inverted internally.

Validity of the cabling rules

The cabling rules described in the following paragraph apply exclusively for the situation in which you have specified a fixed port setting in STEP 7.

Rules for cabling

You can connect several IO devices in series using patch cables. To do this, connect port 2 of the IO device (distributed I/O) with port 1 of the next IO device. The following graphic provides an example with two IO devices.

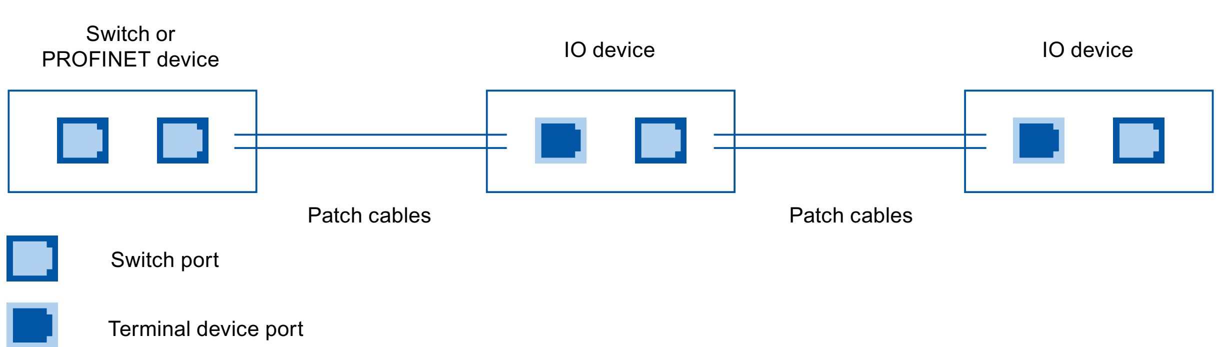


Figure 6-119 Optimized cabling for accelerated startup

6.13.6 Measures in the user program

Introduction

You can make certain changes in the user program to accelerate the startup for IO devices of docking systems that alternate during operation.

Making the required changes to the user program

To accelerate the startup by making changes to the user program, follow these steps:

1. Call the "D_ACT_DP" instruction to activate and deactivate the IO device at the start of the OB 1 cycle.
   If the IO device is replaced, the device that is no longer required must be deactivated and the new device must be activated.
2. If you control the "Power ON" command of the IO devices via the application, then start the activation of the IO devices via the corresponding call of the "D_ACT_DP" instruction as near to the same time as possible. The IO device has to be physically connected for this.
3. You access the IO device using direct access commands after the "D_ACT_DP" instruction has provided the feedback "IO device activated".

6.14 Dealing with timeouts while exchanging data

6.14.1 Terminating communication relation during faults

The prerequisite for data exchange between an IO controller and an IO device is an existing communication relation (Application Relation, AR).

In a communication relation, data is transmitted cyclically and acyclically. If a protocol error or a timeout is detected during acyclic data record communication (e.g. program instructions "RDREC" or "WRREC", or reading of a diagnostic data record), the IO controller terminates the communication relation. In this case, the IO controller assumes that a communication error has occurred. This also ends the cyclic IO data exchange. The error code is returned as a static error in the instruction to the data record transfer. The error is indicated in the diagnostics buffer of the IO controller as the event "IO device failure - Timeout in acyclic PROFINET services".

Possible causes for a termination of the communication relation include:

- A high network load
- Too many ARP telegrams (Address Resolution Protocol).
- Too many UDP frames (User Datagram Protocol).
- Erroneous RPC calls (Remote Procedure Call).

Maintain communication relation

If you can tolerate timeouts during the data record communication from the point of view of your application and no negative repercussions are to be feared, you can change the described default behavior. If, for example, timeouts occur during the data record communication, the communication relation is still preserved. Cyclic IO data exchange continues.
A data exchange error is then signaled as a temporary error (status code: 16#DE80_C300 or 16#DF80_C300) in the instruction for data record transfer.
Starting from STEP 7 V19, you have two options for enabling maintenance of the communication relation. Depending on the FW version of the S7-1500 CPU used, you can enable maintenance as follows:

- S7-1500 CPU as of FW version V3.1:
  - You can configure the maintenance in the CPU properties.
  - You can transfer a data record 0xB072 to an integrated PROFINET interface of the S7-1500 CPU.
- S7-1500 CPU as of FW version V3.0: You can transfer a data record 0xB072 to an integrated PROFINET interface of the S7-1500 CPU.

You must enable maintenance of the communication relation separately for each required PROFINET interface:

- When configuring in the CPU properties, you must enable maintenance for each required PROFINET interface separately.
- When transferring a data record 0xB072, you must transfer this data record separately to each required PROFINET interface.

You will find the various procedures for maintaining the communication relation in the sections below.

Maintaining the communication relation in S7-1500 R/H systems

For S7-1500 R/H CPUs, enable maintenance of the communication relation as described above. However, the S7-1500 R/H system does not synchronize all changes automatically. Therefore you must make the following changes:

- Enable in the CPU properties: Enable maintenance of the communication relation in the CPU properties of one CPU of the S7-1500 R/H system. STEP 7 automatically synchronizes this setting between the two CPUs.
- Transfer a data record: A PROFINET interface, e.g. X1, has different hardware identifiers for the two CPUs. Therefore, write the data record during runtime to the respective addressed PROFINET interface of both CPUs.

In the case of S7-1500 R/H systems, you can only maintain the communication relation at the PROFINET interface X1.

6.14.2 Enabling maintenance of the communication relation in the CPU properties

Requirements

If you want to enable maintenance of the communication relation in the CPU properties, the following requirements must be met:

- STEP 7 V19
- S7-1500 CPU FW version V3.1 or higher
- IO system, e.g. "PLC_1.PROFINET IO System (100)", has been configured at the relevant PROFINET interface

Procedure

The following procedure shows an example of how to enable maintenance of the communication relation at the integrated PROFINET interface X1. The procedure is identical for all integrated PROFINET interfaces of the S7-1500 CPU. To enable maintenance of the communication relation in the CPU properties, follow these steps:

1. Switch to the network view of STEP 7.
2. Configure an S7-1500 CPU, e.g. CPU 1516-3 PN/DP, with the required IO devices.
3. Select a PROFINET interface of the S7-1500 CPU, e.g. X1.
4. In the Inspector window, go to "General > Advanced options > Interface options".
5. In the "Interface options" area, select the "Maintain PROFINET IO communication on data record communication timeout" option.

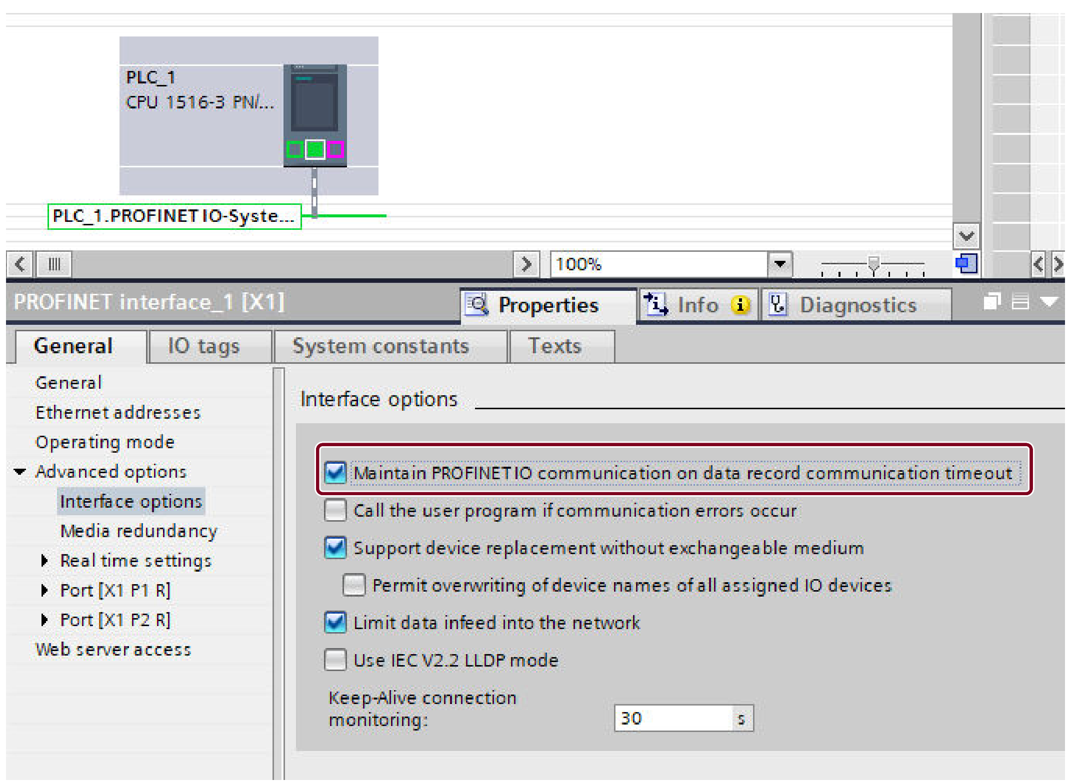


Figure 6-120 Selecting the "Maintain PROFINET IO communication on data record communication timeout" option

6. If required, repeat steps 3 to 5 for another integrated PROFINET interface of the S7-1500 CPU.
7. Load the changed settings into the S7-1500 CPU.
   Result: The communication relation is maintained at the respective configured PROFINET interface of the S7-1500 CPU.

The "Maintain PROFINET IO communication on data record communication timeout" option is only available if you have added an IO system to the relevant PROFINET interface of the S7-1500 CPU. If no IO system exists at this PROFINET interface, the option is displayed in gray. In this case, the option cannot be selected.

If you delete an existing IO system after selecting the "Maintain PROFINET IO communication on data record communication timeout" option, the check box remains selected. The option is displayed in gray in this case and cannot be changed.

Behavior of the option when replacing devices in STEP 7

The "Maintain PROFINET IO communication on data record communication timeout" option can only be selected in STEP 7 V19 for configured S7-1500 CPUs as of FW version V3.1. If you exchange or replace a S7-1500 CPU, the option behaves as follows, depending on the configured FW version:

- S7-1500 CPU with FW version V3.1 → S7-1500 CPU with FW version V3.1:
  STEP 7 applies the settings of the previous S7-1500 CPU to the new S7-1500 CPU.
- S7-1500 CPU with FW version V3.1 → S7-1500 CPU with FW version < V3.1:
  STEP 7 shows a warning message. The option is disabled and can no longer be selected.
- S7-1500 CPU with FW version V3.0 → S7-1500 CPU with FW version V3.1:
  STEP 7 shows an information message. The option can be selected and is deselected in the default settings.

The changes do not take effect on the new S7-1500 CPU until the configuration has been loaded. Therefore, load the changed configuration into the new S7-1500 CPU.

6.14.3 Enabling maintenance of the communication relation by data record transfer

Requirements

If you want to enable maintenance of the communication relation by transferring a data record, the following requirements must be met:

- STEP 7 version V18 or higher
- S7-1500 CPU FW version V3.0 or higher
- At least one IO device has been configured at the addressed PROFINET interface

Procedure

Transfer a data record 0xB072 to the PROFINET interface of the S7-1500 CPU. Coding in this data record indicates whether the communication relation is to be terminated or remain active following an error. The data record acts only on the addressed PROFINET interface.

Option for transferring the data record 0xB072: Definition of the data record structure in the static local data of a function block and call of this function block in the program cycle OB. To do this, follow these steps:

1. Create a function block in STEP 7.
2. For the static local data (Static) of the function block, create a structure of the data record 0xB072.
   – The following table shows the structure of the data record 0xB072:

Byte	Element	Coding	Explanation
0-1	BlockType	0xB072	Header The data record length is counted from byte 4 "BlockVersionHigh".
2-3	BlockLength	8	
4	BlockVersionHigh	0x01	
5	BlockVersionLow	0x00	
6-7	Reserved	-	-
8-9	ReadWriteBehaviorProperties	0, 1	0: Default behavior. The communication relation is terminated in the case of the scenarios described above. 1: The communication relation is maintained in the case of the scenarios described above.
10-11	Reserved	-	-

3. Call the created function block in the program cycle OB (OB 1) of your user program. A programming example for this function block can be found in the next section.

The newly set behavior is not permanently stored in the S7-1500 CPU. The default setting becomes active again after each POWER OFF/POWER-ON, memory reset, or loading of the hardware configuration. The changed behavior is not active until the write job in the program cycle OB has finished.

NOTE

Changing the communication behavior for each PROFINET interface

The call of the function block in the program cycle OB relates only to the addressed PROFINET interface of the S7-1500 CPU. If you want to maintain the communication relations for other PROFINET interfaces, you must call the function block separately in the program cycle OB for each additional integrated PROFINET interface of the S7-1500 CPU.

Maintaining the communication relation in S7-1500 R/H systems

You can also change the default behavior for S7-1500 R/H CPUs by transferring a data record 0xB072.
However, the S7-1500 R/H system does not automatically synchronize this change in behavior for the respective addressed PROFINET interface at both CPUs, since the PROFINET interfaces have different hardware identifiers. Therefore, write the data record during runtime to the respective addressed PROFINET interface of both CPUs. You can only maintain the communication relation at the PROFINET interface X1.

6.14.4 Example: Transfer data record for behavior change via FB

Task

You want to maintain the communication relation (AR) between the IO controller and IO device even in the event of a temporary data record transfer error, e.g. due to elevated network load.

Requirements

An S7-1500 CPU FW version V3.0 or higher with at least one configured IO device at the addressed PROFINET interface is required.

Solution

Transfer the data record 0xB072 to the PROFINET interface of the S7-1500 CPU. This instructs the S7-1500 CPU to maintain an existing communication relation with the IO device.

The following example shows you how to transfer the data record to the PROFINET interface using a function block (FB). Create a structure of the data record 0xB072 in the block interface under "Static". The following figure shows the structure of the data record in "Static" section.

FbBehaviorAR

	Name	Data type	Default value	Retain	Comment
1	▶ Input				
2	▶ Output				
3	▼ InOut				
4	bStart	Bool	0	Set in IDB	Start writing Data record 16#B072
5	InterfaceID	HW_IO	64	Non-retain	Default for Interface X1
6	wBehaviorProperties	Word	16#1	Non-retain	0: AR will be dropped, 1: AR stays active
7	▼ Static				
8	▼ statArRecord	Struct		Non-retain	Data record 16#B072
9	statBlockType	Word	16#B072	Non-retain	Data record number
10	statBlockLength	Word	16#8	Non-retain	Length of block (default: 8)
11	statBlockVersionHigh	Byte	1	Non-retain	Byte 1 of blockversion
12	statBlockVersionLow	Byte	0	Non-retain	Byte 2 of blockversion
13	statReserved	Word	16#0	Non-retain	Reserved for future usage
14	statBehaviorProperties	Word	1	Non-retain	0: AR will be dropped, 1: AR stays active
15	statReserved1	Word	16#0	Non-retain	Reserved for future usage
16	▶ instWrrec	WRREC			Instance of write record
17	▶ Temp				
18	▶ Constant				

Figure 6-121 Example: Interface of the function block for obtaining the communication relation

In addition to the data record 0xB072, the interface contains other tags that you use to transfer the data record. You need the tags in the "InOut" section to assign parameters for the "WRREC" instruction in the function block. The "WRREC" instruction is executed in an organization block (OB1) when the FB is called.

The program in the function block and the call in the program cycle OB are shown in the following sections.

Program in the function block

The data record is transferred at a positive signal at the "#bStart" tag. The parameters for the tags "#wBehaviorProperties", "#bStart" and "#InterfaceID" are passed to the function block when called in the program cycle OB.

```
//==========================================================
// Called in OB1
//==========================================================
#statArRecord.statBehaviorProperties := #wBehaviorProperties;
 #instWrrec(REQ := #bStart,
 ID := #InterfaceID,
 INDEX := DWORD_TO_DINT(16#B072),
 LEN := 12, // 0 : any length,
 this record uses 12 Bytes,
 RECORD := #statArRecord);
 IF #instWrrec.ERROR THEN
 RETURN; // add error handling
 END_IF;
```

```
IF #instWrrec.DONE = TRUE THEN
#bStart := FALSE;
END_IF;
```

Calling in a program cycle OB

Drag-and-drop the created function block into the desired program cycle OB (e.g. OB1). The "InOut" parameters are not linked to tags in this example. Instead, they are controlled directly in the instance of the block call via a watch table.

```
//=========================================================
// Call FbBehaviorAR
//=========================================================
"Inst_FbBehaviorAR_1"(bStart:=_bool_inout_,
 InterfaceID:=_hw_io_inout_,
 wBehaviorProperties:=_word_inout_);
```

Control via the watch table

1. Create a new watch table.
2. Drag-and-drop the required tags from the instance of the function block into the watch table.

FHB_PN_AR-Uebersicht ▸ PLC_5 [CPU 1516-3 PN/DP] ▸ Watch and force tables ▸ Watch table_1

	i	Name	Address	Displa..	Monitor val...	Modify v...	⚡	Comment
1		// Parameters for control						
2		"Inst_FbBehaviorAR_1".InterfaceID		DEC			☐	X1: 64, X2: 72
3		"Inst_FbBehaviorAR_1".wBehaviorProperties		Hex			☐	0: AR default, 1: AR stays active
4		"Inst_FbBehaviorAR_1".bStart		BOOL			☐	Start writing data record
5		// Parameters for the status display						
6		"Inst_FbBehaviorAR_1".Inst_WRREC.DONE		BOOL			☐	Writing the record is done
7		"Inst_FbBehaviorAR_1".Inst_WRREC.BUSY		BOOL			☐	Writing the record is in progress
8		"Inst_FbBehaviorAR_1".Inst_WRREC.ERROR		BOOL			☐	Error while writing the record
9		"Inst_FbBehaviorAR_1".Inst_WRREC.STATUS		Hex			☐	Function result/error message

Figure 6-122 Watch table for transferring the data record

3. Load your project into the S7-1500 CPU.
4. Click "Go online".

You transfer the data record to the PROFINET interface as follows:

1. For the "Inst_FbBehaviorAR_1".InterfaceID" tag, enter the appropriate value for the interface for which you want to change the behavior.
2. To change the behavior, enter the value "1" for the "Inst_FbBehaviorAR_1".wBehaviorProperties" tag.
3. Control the "Inst_FbBehaviorAR_1".bStart" tag to transfer the "TRUE" data record.
4. Repeat steps 1 to 3 for an additional interface where you want to change the behavior.

You can see the status of the transmission to the PROFINET interface as follows:

- "Inst_FbBehaviorAR_1".Inst_WRREC.DONE: The data record was successfully transferred to the PROFINET interface.
- "Inst_FbBehaviorAR_1".Inst_WRREC.BUSY: The data record is currently being transferred to the PROFINET interface.
- "Inst_FbBehaviorAR_1".Inst_WRREC.ERROR: The transfer of the data record to the PROFINET interface was faulty.
- "Inst_FbBehaviorAR_1".Inst_WRREC.STATUS: Output of block status or error information. The corresponding interpretations can be found in the STEP 7 help system.

Note that the variables „Inst_FbBehaviorAR_1".Inst_WRREC.DONE and „Inst_FbBehaviorAR_1".Inst_WRREC.ERROR are only present for the duration of a program cycle.

Additional steps for S7-1500R/H systems

Each addressed PROFINET interface of the two R/H CPUs has different hardware identifiers. Therefore, transfer the data record to the respective addressed PROFINET interface of both CPUs for S7-1500 R/H systems.

The values for the "Inst_FbBehaviorAR_1".InterfaceID" tag for S7-1500 R/H systems are listed in the following table:

Table 6-7 Overview of CPUs and associated hardware identifiers of the PROFINET interfaces

CPU	PROFINET interface	Hardware identifier of the PROFINET interface
CPU with redundancy ID "1" (prefixed name: PLC_1)	X1	65164 (prefixed name: Local1~PROFINET-interface_1)
	X2	PROFINET IO is currently not supported.
CPU with redundancy ID "2" (prefixed name: PLC_2)	X1	65364 (prefixed name: Local2~PROFINET-interface_1)
	X2	PROFINET IO is currently not supported.

NOTE

Transfer of the data record to the backup CPU

Do not transfer the data record to the addressed PROFINET interface of the backup CPU until after the S7-1500 R/H system has reached the "Run REDUNDANT" system state. Otherwise, the data record cannot be transferred to the addressed PROFINET interface of the backup CPU.

The CPU redundancy error OB (OB72) starts when the S7-1500 R/H system has reached the "Run REDUNDANT" system state. The "Fault_ID" tag of the OB72 contains the error code "B#16#03" or "B#16#06".

6.15 Configuring SNMP and DCP in the PROFINET network

6.15.1 Configuring the SNMP

Component interaction

PROFINET networks use the SNMP protocol for managing devices and the network infrastructure.

For information exchange, authentication must take place through the transfer of community strings. When SNMP is used, a community string must always have been assigned. Empty community strings are not allowed.

You can find configuration options for SNMP in the properties of the following devices:

- S7-1500 CPU as an IO controller/I-device
- IO devices that support the configuration of SNMP

To learn how to configure SNMP for the various components, see the following sections.

NOTE

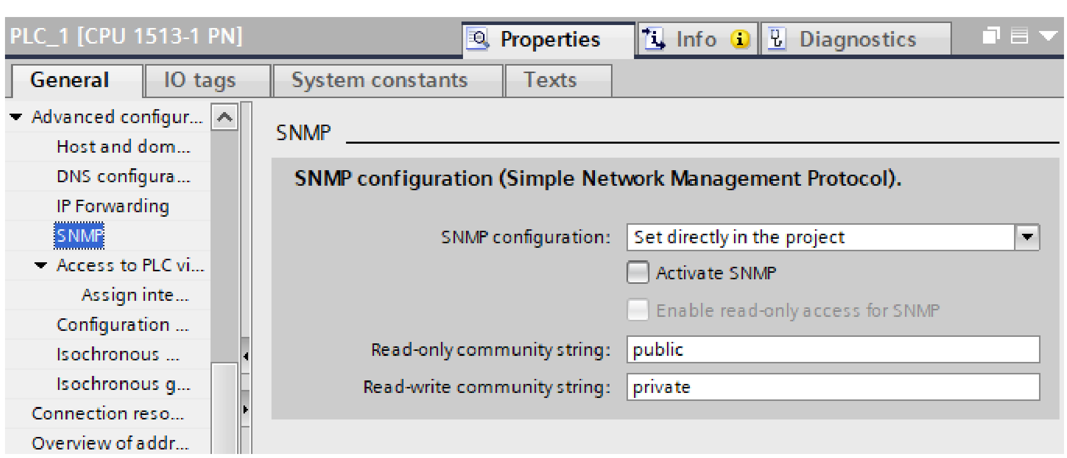
Avoidance of erroneous access attempts

Inform maintenance personnel about the SNMP-configuration in your plant.

Configuring SNMP for an IO controller

To configure SNMP for an IO controller, follow these steps:

1. Open the Properties of the IO controller in the network view or device view.
2. Navigate to the "Advanced configuration > SNMP" area in the Inspector window.
3. If you want to use SNMP, select the "SNMP " option in the "SNMP" area.
   Once you enable SNMP, a warning message appears informing you of the reduced protection against unauthorized access to functions and data of this device.
4. If you only need read access for SNMP, also select the "Enable read-only access for SNMP" option .
5. Change the Read-only and Read-write community strings.
   This increases the security of your system against cyberattacks. Empty community strings are not allowed.



Result: You have configured SNMP in the properties of your IO controller.

Configuring SNMP for an IO device

If an IO device supports the configuration of SNMP, you have the option of defining a separate SNMP configuration for this IO device as of STEP 7 V19. For IO devices that were imported as a GSD file, you can also modify the SNMP configuration in STEP 7 as of V19 or with other tools.

To configure SNMP for an IO device in your PROFINET network, follow these steps:

1. Open the properties of the IO device in the network or device view.
2. Select "Advanced configuration" in the Inspector window.
3. If you want to use SNMP, select the "Enable SNMP" option.
   Once you enable SNMP, a warning message appears informing you of the reduced protection against unauthorized access to functions and data of this device.
4. If you only need read access for SNMP, also select the "Enable read-only access for SNMP" option .
5. Change the Read-only and Read-write community strings.
   This increases the security of your system against cyberattacks. Empty community strings are not allowed.

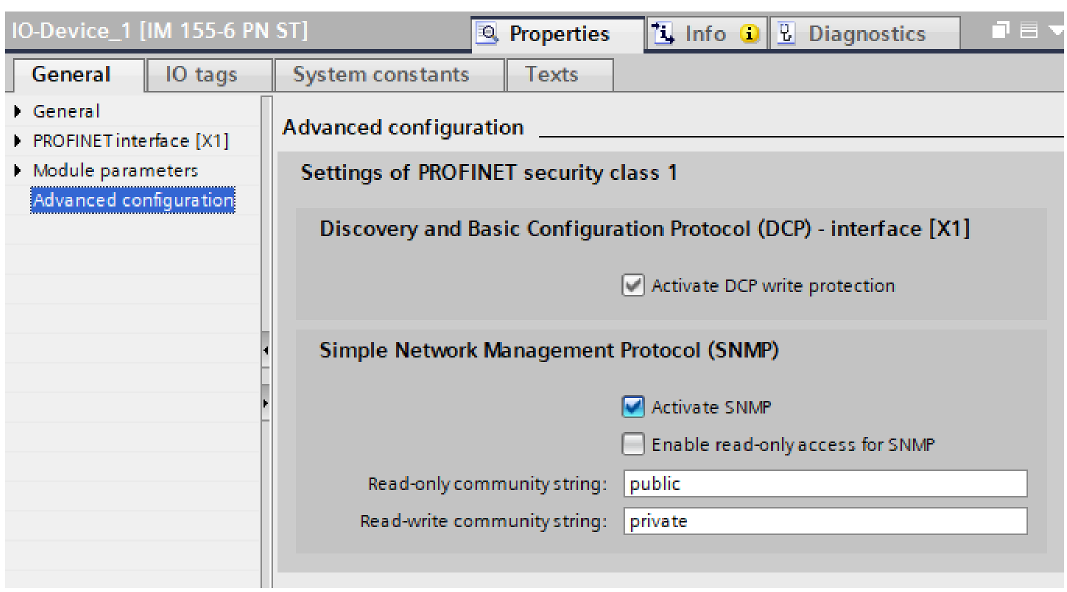


Figure 6-123 Configuring SNMP on an IO device

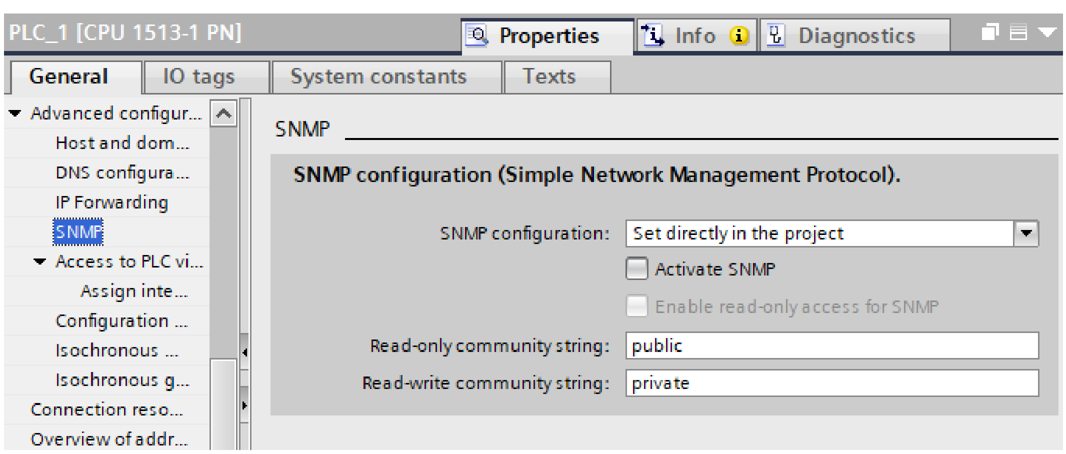
Result: You have configured SNMP in the properties of your IO device.

Configuring SNMP for an I-device (IO controller and I-device in the same project)

Since STEP 7 V18 you have the option of enabling SNMP in the CPU properties of an IO controller. When a CPU assumes the role of an I-device, you can also define the setting for SNMP in the CPU properties.

You configure SNMP when creating an I-device the same as for an IO controller. To access the SNMP configuration options for an I-device, follow these steps:

1. Open the Properties of the CPU that has been configured as an I-device in the network or device view.
2. Navigate to "Advanced configuration > SNMP" in the Inspector window.
3. Select one of the following options from the drop-down list next to "SNMP configuration":
   - Set directly in the project:
     The I-device adopts the SNMP configuration loaded from the I-device configuration to the I-device at its PROFINET interface X1. Changes to the SNMP configuration by a higher-level IO controller are prevented.
   - Via PROFINET interface ... (e.g. X1) from higher-level IO controller:
     The I-device adopts the SNMP configuration from the higher-level IO controller at its PROFINET interface X1.
     Requirements:
     The "Parameter assignment of PN interface by higher-level IO controller" option is enabled at the I-device.
     An IO controller has been selected in the "Higher-level IO controller of the PN interface" drop-down list.
4. If you want to use SNMP, select the "Enable SNMP" option.
   Once you enable SNMP, a warning message appears informing you of the reduced protection against unauthorized access to functions and data of this device.
5. If you only need read access for SNMP, also select the "Enable read-only access for SNMP" option .
6. Change the Read-only and Read-write community strings.
   This increases the security of your system against cyberattacks. Empty community strings are not allowed.



Result: You have configured SNMP in the properties of your project-integrated I-device.

Configuring SNMP for an I-device as a GSD file

To use an I-device in another project or Engineering System, export the configured I-device as a general station description file (GSD file) in STEP 7. When you import the GSD file in the desired project, STEP 7 shows the configured I-device as a GSD device (DP standard device) in the device or network view.

Before the GSD file export, define how you want to use SNMP on the DP standard device. Starting from STEP 7 V19, you have the following options in the settings of SNMP:

- Configuring of SNMP in the local settings of the I-device:
  The exported GSD file prevents changes from being made to the SNMP configuration in another project. The SNMP configuration options are not displayed on the inserted DP standard device.
- Configuring of SNMP by a higher-level IO controller in another project:
  The SNMP configuration must be created in the other project in the same way as for a standard IO device.
  Requirement:
  The option "Parameter assignment of PN interface by higher-level IO controller" is enabled at the I-device.

If you want to configure SNMP on the DP standard device in another project, make the following settings on the I-device CPU:

1. At the PROFINET interface of the CPU that is configured as an I-device:
   – Activate the "Parameter assignment of PN interface by higher-level IO controller" option in the Inspector window under "Properties > Operating mode".
2. In the SNMP settings of the I-device CPU:
   – Select the entry "via PROFINET interface ... (e.g. X1) from higher-level IO controller" from the drop-down list next to "SNMP configuration".
   In this case, the I-device takes over the SNMP configuration from the higher-level IO controller at its PROFINET interface X1.

Result: If using the I-device in another project as a DP standard device, you can configure SNMP the same as for a standard IO device.

If using the I-device as a shared I-device in multiple projects, the following applies: Configure SNMP only in the project with the controller to which the interface of the I-device was assigned.

6.15.2 Configuring DCP

Component interaction

DCP is a protocol for automatic detection and basic configuration of PROFINET devices. Consequently, DCP needs read and write access to the individual components in a PROFINET network.

To protect the components from malicious or inadvertent write access, the "Activate DCP write protection" function is available starting from STEP 7 V19.
When the "Activate DCP write protection" function is enabled on an IO device, the DCP write protection operates as follows on the individual components:

- The DCP write protection takes effect on both the IO device and the IO controller when an active communication relation exists between the IO device and the IO controller.
- The DCP write protection takes effect on the associated IO controller of the IO device when the IO controller establishes a communication relation with the IO device.

When configuring DCP, take into account the communication relations throughout the PROFINET network:

- An IO device has only one communication relation with its IO controller.
- A shared device/I-device has communication relations with all higher-level IO controllers.
- An IO controller has communication relations with all its assigned IO devices and shared devices/I-devices.

Consequently, DCP write access by the IO controller is only permitted again after all communication relations with DCP write protection with devices have been terminated and disabled (e.g. via one call each of D_ACT_DP).

To learn how to configure DCP, see the following sections.

NOTE

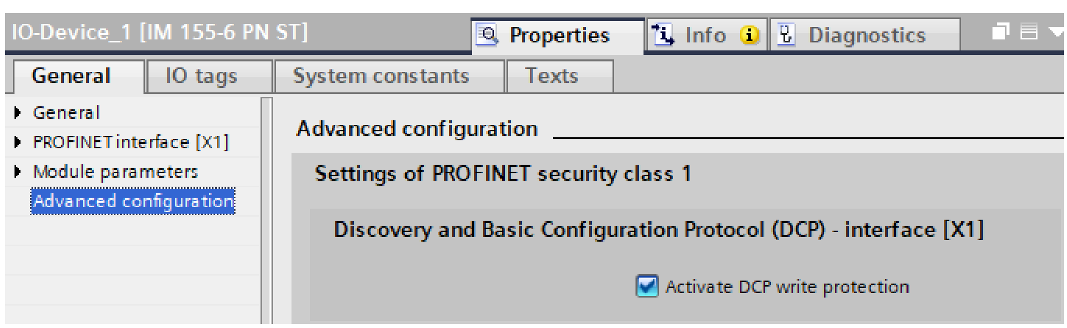
Avoidance of erroneous access attempts

Specific settings, such as changing of device names via DCP, only work for devices that are not currently participating in cyclic data exchange.

Configuring DCP for a communication relation on an IO device

To configure DCP for a communication relation between IO controller and IO device, follow these steps:

1. Open the Properties of the IO device in the network or device view.
2. Select the entry "Advanced configuration" under "General" in the Inspector window.



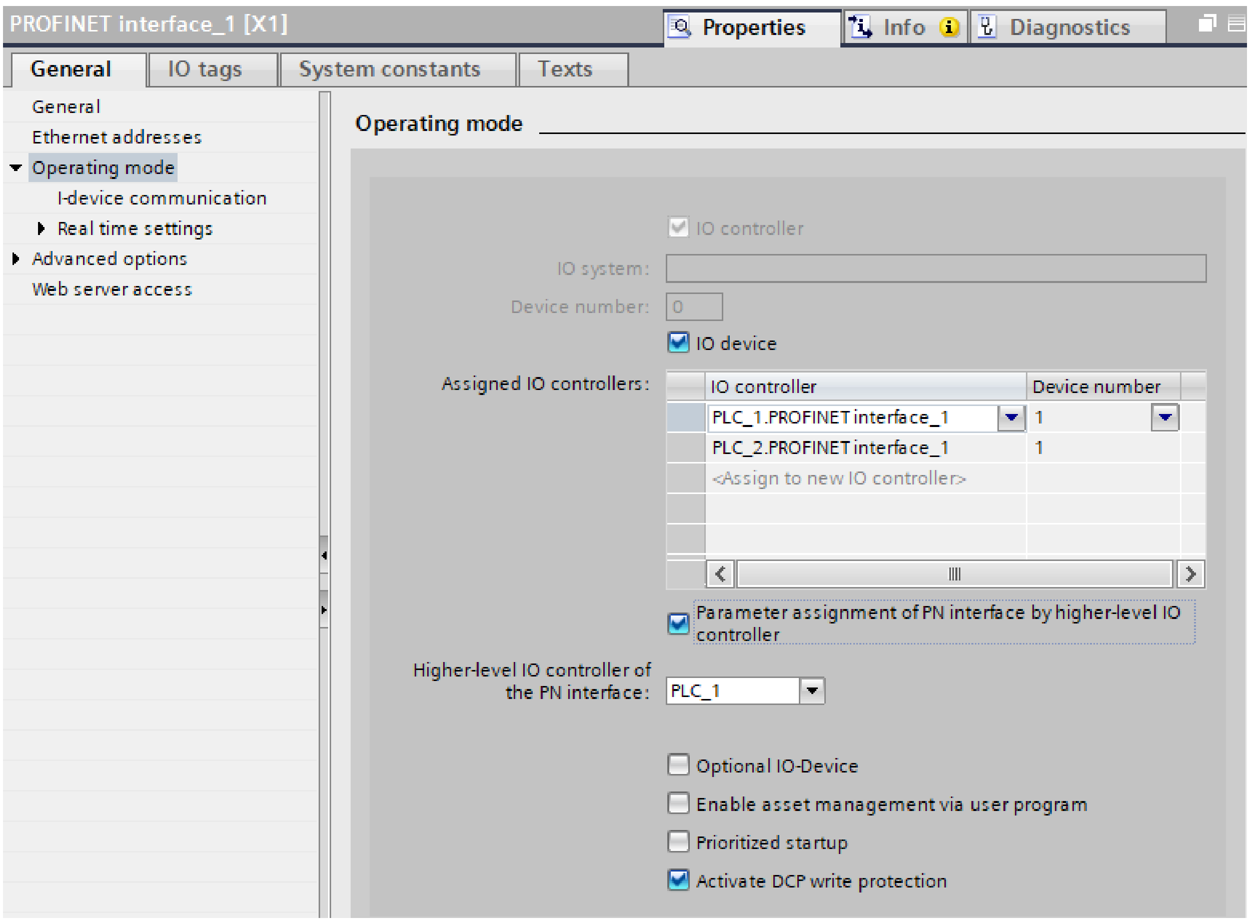
The "Activate DCP write protection" option is enabled by default on all IO devices that support this function. The option cannot be changed in the following cases:

- The IO device is not assigned to any IO controller.
- The IO controller to which the IO device is assigned does not support the "Activate DCP write protection" function.

Configuring DCP for a communication relation on an I-device

To configure DCP for a communication relation between an IO controller and I-device in a common project with the assigned IO controllers, follow these steps:

1. Open the properties of the required PROFINET interface, e.g. X1, in the network or device view.
2. Select the entry "Operating mode" under "General" in the Inspector window.



The "Activate DCP write protection" option is enabled by default on all I-devices that support this function. The option cannot be changed in the following cases:

- The I-device is not assigned to any IO controller.
- The IO controller to which the I-device is assigned does not support the "Activate DCP write protection" function.

If you use an I-device as a shared I-device for IO controllers in different projects, the I-device will be inserted into other projects as a DP standard device. In this case, you therefore configure the DCP the same as for a standard IO device. The DCP write protection takes effect on each IO controller that has a communication relation with the shared I-device and supports the DCP write protection function.

7 PROFINET with the redundant S7-1500R/H system

Introduction

The same basic rules apply to PROFINET IO communication with the redundant S7-1500R/H as with the S7-1500 automation system. The redundant S7-1500R/H system exchanges IO data cyclically with the IO devices.

The redundant S7-1500R/H system supports a limited scope of PROFINET functions.

The redundant S7-1500R/H system supports media redundancy (MRP) in the PROFINET ring.

You can use the following IO devices on the redundant S7-1500R/H system:

- IO devices with system redundancy S2
- IO devices with system redundancy R1 (S7-1500H as of firmware version V3.0)
- Standard IO devices over the "Switched S1 device" function of the CPU

When configuring PROFINET with the redundant S7-1500R/H system, you must observe a few specifics, e.g. special configuration requirements.

Restrictions

- Real-time communication:
  - RT with a fixed send clock of 1 ms
  - No IRT
  - No MRPD redundancy procedure
  - No PROFINET with performance upgrade
- No isochronous mode
- No direct data exchange
- No operation as I-device
- No access to shared devices
- No support of docking systems
- No support of series machine projects
- Port options (only for PROFINET interface X1):
  - No port disabling possible
  - No configuring of boundaries possible
- Functionality as IO controller:
  - No prioritized startup
  - No enabling/disabling of IO devices
  - No support of series machine projects, e.g. IO systems that can be used multiple times
- PROFINET interface X2 does not support IO functionality (CPU 1515R/CPU 1517H/CPU 1518HF)
- No diagnostics using the web server
- No IP address assignment by the DHCP server

7.1 Media redundancy in the redundant S7-1500R/H system

You can use the redundant S7-1500R/H system in media redundant networks (MRP). In these networks some or all PROFINET devices are connected to the S7-1500R/H via one or more PROFINET rings in a media-redundant manner. If a PROFINET ring is interrupted, then a reconfiguration may occur. After a short reconfiguration time, the PROFINET devices can be accessed again via an alternative path. Note that with the redundant S7-1500R system, the PROFINET ring is mandatory.

7.2 H-Sync Forwarding

Introduction

H-Sync Forwarding enables a PROFINET device with MRP to forward synchronization data (synchronization frames) of a S7-1500R redundant system only within the PROFINET ring.

In addition, H-Sync Forwarding forwards the synchronization data even during reconfiguration of the PROFINET ring. H-Sync Forwarding avoids a cycle time increase if the PROFINET ring is interrupted.

NOTE

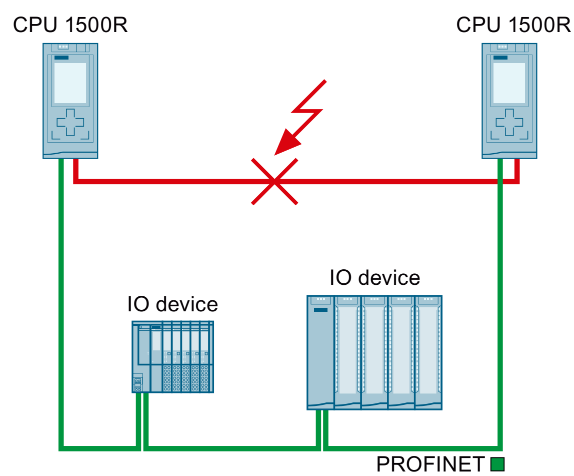
Support of H-Sync Forwarding

The technical specifications typically state whether a PROFINET device supports H-Sync forwarding.

The GSD file will also indicate whether the device supports H-Sync Forwarding. The device supports H-Sync Forwarding when the "AdditionalForwardingRulesSupported" attribute in the "MediaRedundancy" element is set to "true".

Conditions

- H-Sync Forwarding is not relevant for redundant S7-1500H systems. With the redundant S7-1500H system, the H-Sync frames are transmitted exclusively via the fiber-optic cables.
- **When you use PROFINET devices with more than two ports (e.g. switch) in the PROFINET ring of an R-system, these devices must support H-Sync Forwarding.**
  H-Sync frames leave the PROFINET ring with a switch without H-Sync Forwarding. This results in an additional load on the network. Another serious result is that the redundancy of other R-systems in the network can fail or startup can be prevented.
- **H-Sync Forwarding is recommended if you are using PROFINET devices with only 2 ports in the PROFINET ring of an R-system**.
  When you operate PROFINET devices without H-Sync Forwarding in the PROFINET ring of the redundant S7-1500R/H system, the following scenario will result in an additional cycle time increase:



1. The redundant S7-1500R system is in the RUN-Redundant system state.
2. The PROFINET cable which directly connects the two CPUs fails.
3. The PROFINET ring is interrupted.
4. The PROFINET ring is being reconfigured.
5. PROFINET devices without H-Sync Forwarding do not forward any H-Sync frames during the reconfiguration time of the PROFINET ring.
6. The cycle time increases by the reconfiguration time of the PROFINET ring.

Figure 7-1 Failure of the PROFINET cable between the CPUs

If the cyclic program exceeds the cycle monitoring time, the time error OB (OB 80) may be started. Redundancy is lost if the time error OB (OB 80) is not present or the cycle time was exceeded twice with OB 80.

NOTE

If failure of the PROFINET cable that directly connects the two CPUs of the redundant S7-1500R system is unlikely, you can use PROFINET devices without H-Sync Forwarding in the PROFINET ring of the redundant S7-1500R system.

Example: Both CPUs of the redundant S7-1500R system are located directly next to each other in the control cabinet. In this case, it is unlikely that the PROFINET cable will fail.

7.3 System redundancy S2

Introduction

System redundancy S2 is the connection of an IO device via an interface module that supports two ARs to the IO controllers in the redundant system.

IO devices with system redundancy S2 enable uninterrupted process data exchange with the S7-1500R/H redundant system in the event of:

- a CPU fails
- Interruption of the PROFINET ring
- Interruption of the line topology (for S7-1500H as of FW version V3.0)

An IO device with system redundancy S2 supports system redundancy ARs.

In a redundant system, an IO device with system redundancy S2 has a system redundancy AR with each of the two CPUs (IO controllers). An IO device thus supports ARs of two IO controllers simultaneously (for the same modules).

A system redundancy AR can be either a primary AR or a backup AR. An IO device activates the data of the primary AR at the outputs. The data of the backup AR are not initially evaluated.

In STEP 7, you configure system redundancy S2 for an IO device by assigning the IO device to both CPUs of the S7-1500R/H redundant system.

Behavior in system state RUN-Redundant

Both CPUs are IO controllers. The PROFINET communication runs simultaneously on both system redundancy ARs, in each case between one of the CPUs (IO controllers) and the IO device. If the primary CPU fails or is switched to STOP, then the backup CPU becomes the primary CPU and also switches the backup AR to primary AR. The data of this AR becomes active at the outputs.

Abbreviations used for AR in the figure below:

- P-AR: Primary AR
- B-AR: Backup AR

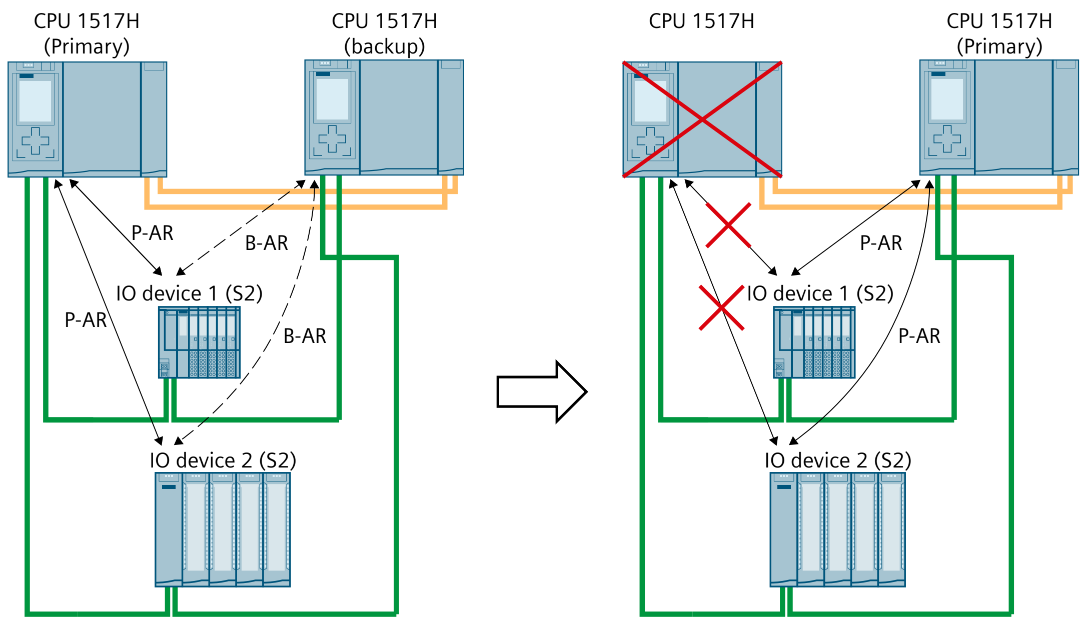


Figure 7-2 Primary backup AR

NOTE

S7-1500H with line topology

With line topology, only the primary AR can fail. In this case, the system switches to the backup AR, but the operating states and the roles of the CPUs remain unchanged.

Behavior in system state RUN-Solo

Only the primary CPU is an IO controller. The PROFINET communication runs on the primary AR between the primary CPU (IO controller) and the IO device. There is no AR between the backup CPU and the IO device.

7.4 System redundancy R1

Introduction

System redundancy R1 is the connection of an IO device via two interface modules, each of which supports an AR to the IO controllers in the redundant system.

IO devices with system redundancy R1 enable uninterrupted process data exchange with the S7-1500H redundant system when:

- a CPU fails
- an interface module fails
- a subnet fails.

Unlike system redundancy S2, system redundancy R1 has a separate interface module for each of the two ARs. Due to these redundant interface modules, the availability is higher than with an S2 device.

In STEP 7 you configure an IO device with system redundancy R1 by:

- Connecting the left interface module of the R1 device to the left H-CPU in the network view.
- Connecting the right interface module of the R1 device to the right H-CPU in the network view.

Reference

You can find information about the setup variants with the redundant system S7-1500R/H in the System Manual Redundant System S7-1500R/H (https://support.industry.siemens.com/cs/us/en/view/109754833).

Behavior in system state RUN-Redundant

The PROFINET communication runs on both system redundancy ARs simultaneously, each between one of the CPUs (IO controller) and an interface module of the R1 device.

In the event of a failure of an interface module of an R1 device or a line break in the line topology, only the ARs switch over. The roles of the H-CPUs do not change.

Abbreviations used for AR in the figures below:

- P-AR: Primary AR
- B-AR: Backup AR

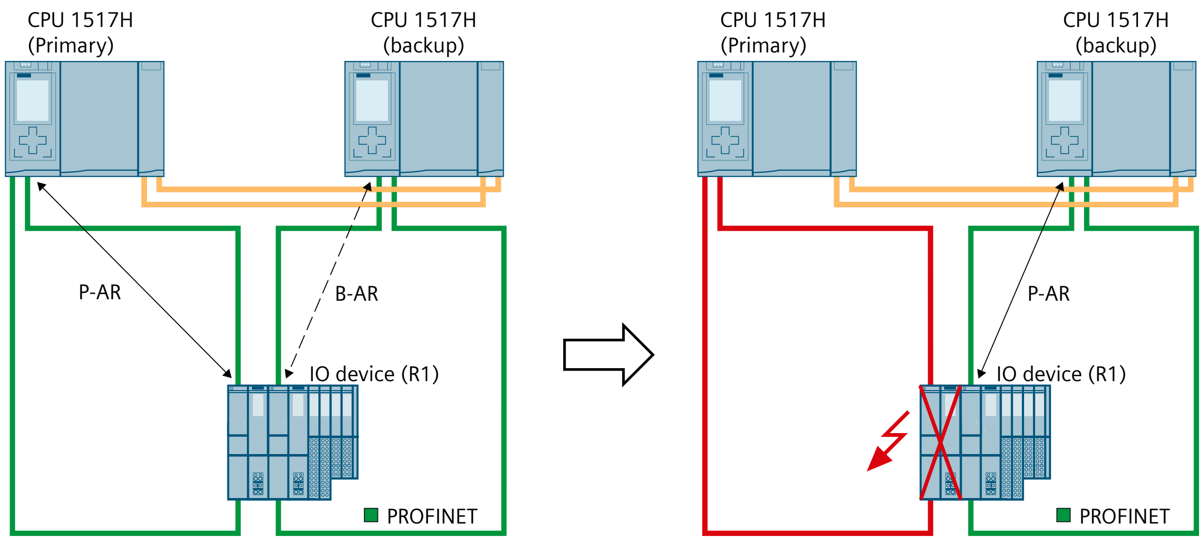


Figure 7-3 Failure of an interface module of an R1 device in the redundant S7-1500H system

If, however, the primary CPU fails or is switched to STOP, the S7-1500H redundant system then behaves as follows:

- The previous backup CPU becomes the new primary CPU.
- The new primary CPU uses the AR to the other, functioning interface module of the IO device and still has access to the inputs and control over the outputs of the IO device.
- The AR between the previous primary CPU and the assigned interface module is disconnected.

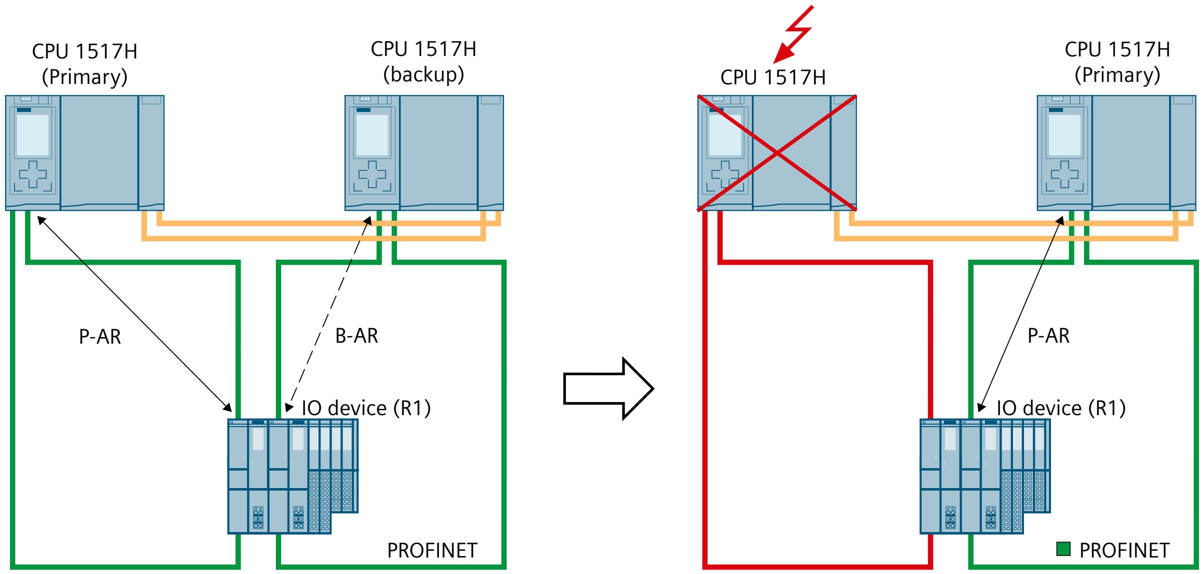


Figure 7-4 Failure of a CPU in the redundant S7-1500H system

NOTE

S7-1500H with line topology

With line topology, only the primary AR can fail. In this case, the system switches to the backup AR, but the operating states and the roles of the CPUs remain unchanged.

Behavior in system state RUN-Solo

Only the primary CPU is an IO controller. PROFINET communication runs on the AR between the primary CPU (IO controller) and the interface module of the IO device assigned to the primary CPU. The backup CPU separates all its ARs.

7.5 Switched S1 device

Introduction

As of FW version V2.8, the S7-1500R/H redundant system supports the "Switched S1 device" function.

The "Switched S1 device" function of the CPU enables operation of standard IO devices on the S7-1500R/H redundant system.

Standard IO devices, too, are always assigned to both CPUs of the S7-1500R/H redundant system. In contrast to an IO device with system redundancy S2, a standard IO device supports only one AR. The AR is available to the primary CPU of the redundant S7-1500R/H system.

In STEP 7 you configure an IO device connected via the "Switched S1 device" function by assigning a standard IO device to both CPUs of the redundant S7-1500R/H system.

As of FW version V3.0, the backup CPU of S7-1500H can establish an AR to an S1 device in a line topology. The primary CPU then receives data via the backup CPU.

NOTE

Standard IO devices in the redundant system S7-1500R

Standard IO devices usually do not support H-Sync Forwarding (Page 305).

To avoid a cycle time increase when the PROFINET ring is interrupted, integrate the standard IO devices downstream of a switch and not in the PROFINET ring.

Behavior in system state RUN-Redundant

The PROFINET communication runs on the AR between one of the CPUs (IO controller, in this example the primary CPU) and the standard IO device. There is no AR between the backup CPU and the standard IO device.
If the primary CPU fails or is switched to STOP, the S7-1500R/H redundant system responds as follows:

- The AR between the primary CPU and the standard IO device is disconnected.
- The previous backup CPU becomes the new primary CPU.
- The S7-1500R/H redundant system temporarily has no access to the inputs and no control over the outputs of the standard IO device. During this time, the configured substitute value behavior applies to the standard IO device.
- The new primary CPU builds an AR to the standard IO device.
- As soon as the new primary CPU has set up the AR, the S7-1500R/H redundant system has access to the inputs again and control over the outputs of the standard IO device.

Abbreviation used for AR in the figures below:

- AR: "normal" AR between a standard IO device and a CPU of the redundant system S7-1500R/H.

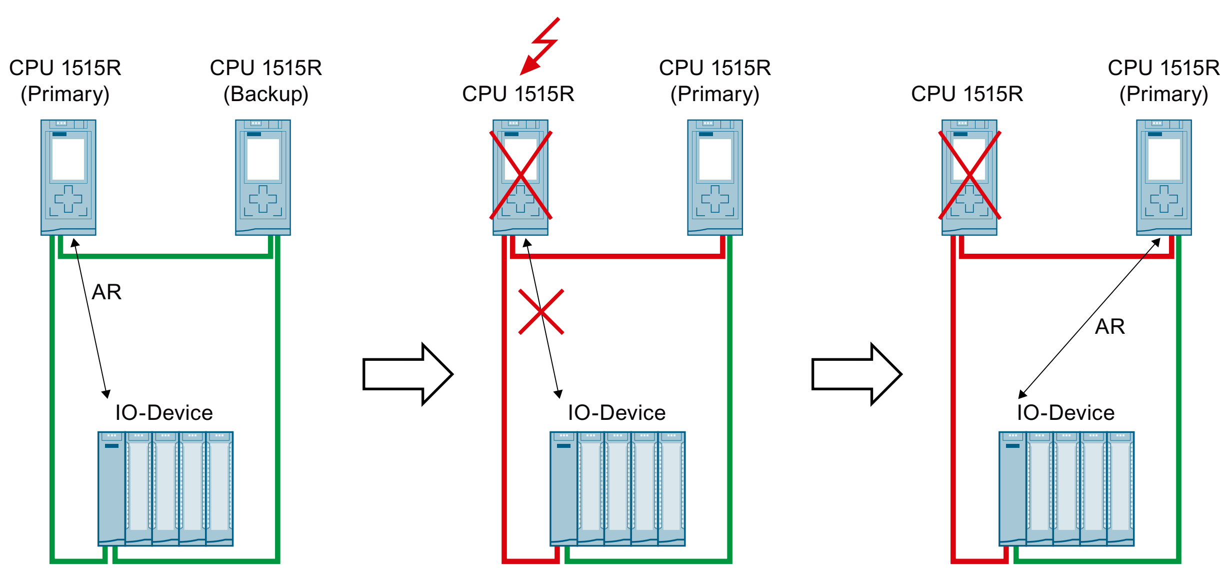


Figure 7-5 Behavior of standard IO devices in case of failure of the primary CPU

If a line in a line topology fails, the redundant S7-1500H system behaves as follows:

- If the line between the primary CPU and a standard IO device is interrupted, then the redundant S7-1500H system temporarily has no access to the inputs and no control over the outputs of the standard IO device. The status of the outputs depends on the substitute value behavior of the respective channels.
- The respective other CPU establishes an AR to the standard IO device.
- Once the new AR has been set up, the redundant S7-1500H system again has access to the inputs and control over the outputs of the standard IO device.

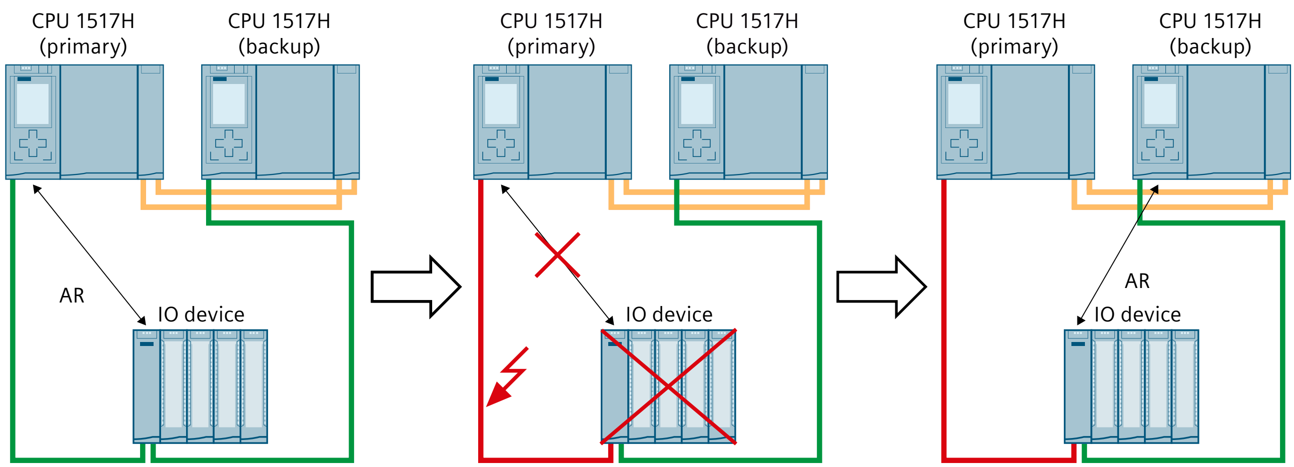


Figure 7-6 Behavior of standard IO devices in case of line failure

Behavior in system state RUN-Solo

Only the primary CPU is an IO controller. PROFINET communication runs on the AR between the primary CPU and the standard IO device. There is no AR between the backup CPU and the IO device.

OB behavior for standard IO devices with primary-backup switchover

If the primary CPU fails or goes to STOP, the standard IO devices temporarily fail. OB 72 "Redundancy error" is called, additional OB 86 "Module rack failure" for the failed IO devices are not called. To detect the failed IO devices, call the DeviceStates instruction in OB 72.

With the "Switched S1 device" function, the new primary CPU establishes the ARs to the standard IO devices again. OB 86 is called for each return of an IO device.

Influence changeover time of switched S1 devices

As of FW version V2.9, you can influence the changeover time between disconnection and return of switched S1 devices after a failure/STOP of the primary CPU.

Advantages:

- Optimization of the changeover time between disconnection and return of switched S1 devices
- Prioritization of the return of S1 devices in the PROFINET ring by setting a lower changeover time for S1 devices with important process functions.

The changeover time depends on the watchdog timer of the switched S1 device. The watchdog timer is the product of the update time multiplied by the accepted update cycles without IO data. With the parameters, you influence the time as of which the new primary CPU starts returning the S1 device. The return also depends on the amount of time that is required for the parameter assignment of the S1 device.

You set the parameters in STEP 7 in the properties of the interface of each S1 device:

- Update time: Advanced options > Real time settings > IO cycle > Update time
- Watchdog timer: Advanced options > Real time settings > IO cycle > Watchdog timer > Accepted update cycles without IO data

Setting a shorter changeover time of an S1 device: Reduce the values of the parameters/of a parameter.

Setting a longer changeover time of an S1 device: Increase the values of the parameters/of a parameter.

7.6 Main differences between IO device with system redundancy S2, R1 and standard IO device

Table 7-1 Main differences between IO device with system redundancy S2, R1 and standard IO device

Property	**IO device with system redundancy S2**	**IO device with system redundancy R1**	**Standard IO device**
Requirement for IO device	Device supports system redundancy S2	Device supports system redundancy R1	-
Maximum simultaneously supported ARs with regard to the same modules	2	2	1
Behavior when replacing the primary CPU	Continuous connection with S7-1500R/H redundant system Process data is transferred further.	Continuous connection with redundant S7-1500H system Process data is transferred further.	Temporary disconnection from S7-1500R/H redundant system. No process data is transferred until the standard IO device is available again. The status of the outputs depends on the substitute value behavior of the respective channels.

Property	IO device with system redundancy S2	IO device with system redundancy R1	Standard IO device
Behavior in case of failure of an interface module	Failure of the S2 device	R1 device continues to transmit process data via the redundant interface module	Failure of the standard IO device
Number of interface modules per device	1	2	1
Number of ARs per interface module	2	1	1

7.7 Installation guidelines

Configuration requirements for the topology with the redundant S7-1500R system

Topology properties	PROFINET ring
MRP role of the CPU	Manager (auto)
MRP role of the IO devices	Client
Other S7-1500R/H systems	No

Port interconnection:

- The two CPUs are directly interconnected via one of the two ports of PROFINET interface X1. No other devices of the ring are located in between.
- The two CPUs can be indirectly interconnected via the other two ports. The other devices of the ring are located in between. See the examples below.

Table 7-2 Number of PROFINET devices, IO devices in the redundant system

Maximum number of devices	Maximum number of S7-1500R
Maximum number of PROFINET devices in the PROFINET ring of the R system The maximum number includes switches, S7-1500R/H CPUs, S7-1500 CPUs (as of V2.5) and HMI devices. It does not include media converters.	50 (recommendation 16)[1)]
Maximum number of IO devices that can be connected to the R-CPUs	64

[1)] Recommendation: The number of devices in the PROFINET ring affects the availability of the S7-1500R system. The number of PROFINET devices including R-CPUs in the PROFINET ring should not exceed 16. If you operate significantly more devices in the PROFINET ring, the availability is reduced.

Configuration requirements for the topologies with the redundant S7-1500H system

Topology properties	PROFINET ring	Line topology	Combined topology	H-CPUs without additional devices
MRP role of the CPU	Manager (auto)	Not device in the ring		Not device in the ring
MRP role of the IO devices	Client	Not device in the ring	Depending on the installation location: Not device in the ring **or** client	-
Other S7-1500R/H systems	No	Yes		Yes

Table 7-3 Number of PROFINET devices, IO devices in the redundant system

Maximum number of devices	Maximum number of S7-1500H
Maximum number of PROFINET devices in the PROFINET ring of the R/H system The maximum number includes switches, S7-1500R/H CPUs, S7-1500 CPUs (as of V2.5) and HMI devices. It does not include media converters.	50
Maximum number of IO devices that can be connected to the H-CPUs	256

7.8 Configuring PROFINET IO on a redundant S7-1500R/H system

PROFINET IO configurations with the redundant S7-1500R/H system

The following figures show you examples of PROFINET IO configurations with the redundant S7-1500R/H system.

Abbreviations of AR used in the following figures:

- SR-AR: System redundancy AR between an IO device (S2/R1) and a CPU of the redundant S7-1500R/H system
- AR: "normal" AR between a standard IO device and the associated IO controller

S2 IO devices in the PROFINET ring

The redundant S7-1500R system consists of the two CPUs 1515R and two S2 devices in the PROFINET ring. The IO devices support MRP and H-Sync forwarding.

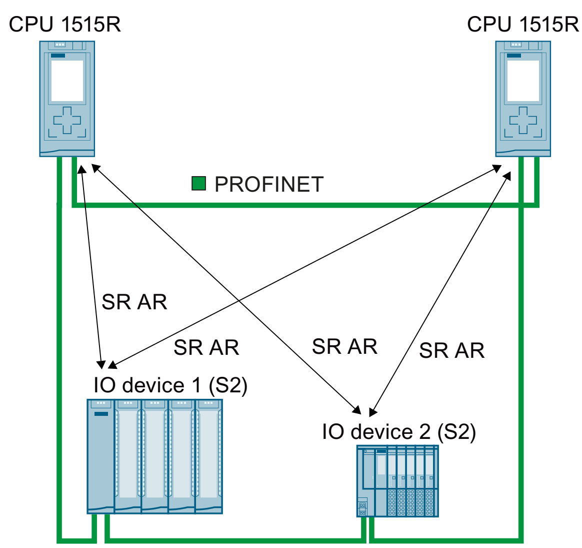


Figure 7-7 Example of a S7-1500R redundant system with two IO devices in the PROFINET ring

IO devices R1 in PROFINET rings

The redundant system S7-1500H consists of the two CPUs 1517H and two R1 devices in two PROFINET rings. The IO devices support MRP.

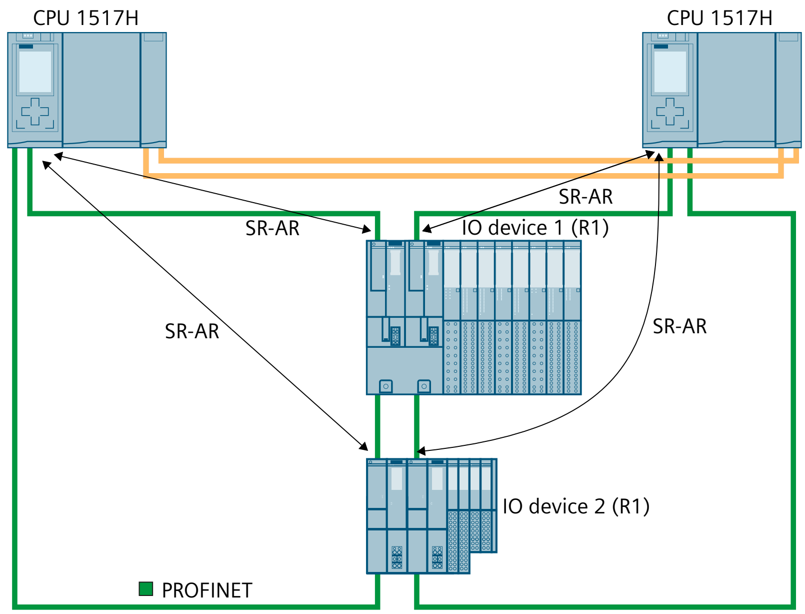


Figure 7-8 Example of a redundant S7-1500H system with two IO devices in two PROFINET rings

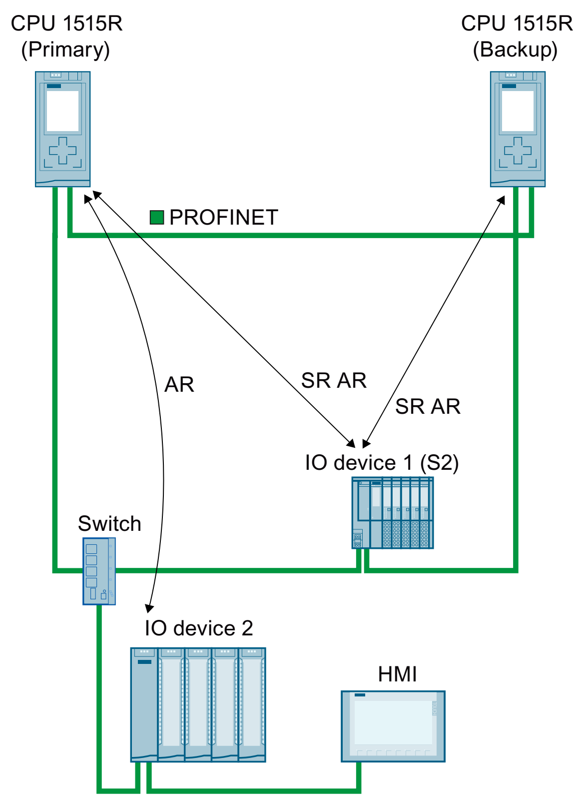
PROFINET devices downstream of a switch

To connect additional PROFINET devices to the PROFINET ring, use a switch. For the redundant S7-1500R system, ensure that a switch in the PROFINET ring supports H-SYNC Forwarding in addition to MRP.

You connect the following devices to the PROFINET ring with a switch.

- PROFINET devices with one port
- Non MRP-capable PROFINET devices
- PROFINET devices that do not support H-Sync Forwarding, such as standard IO devices.

The following figure shows a redundant S7-1500R system with connected PROFINET devices downstream of a switch.



IO device 1 (S2)	IO device 1 is located in the PROFINET ring. The IO device 1 supports system redundancy S2, MRP and H-Sync Forwarding.
Switch	The switch is located in the PROFINET ring. The switch supports MRP and H-SYNC Forwarding.
IO device 2	The IO device 2 is connected to the PROFINET ring via the switch. IO device 2 is a standard IO device that is connected to the redundant S7-1500R system via the "Switched S1 device" function.
HMI device	The HMI device is connected to the PROFINET ring via the switch.

Figure 7-9 PROFINET devices downstream of a switch

Additional PROFINET IO system in the configuration

The configuration can also contain additional IO controllers with their own IO devices. To have enough bandwidth in the PROFINET ring for the redundant S7-1500R/H system, place additional IO controllers with their I/O devices downstream of a switch.

When determining the maximum number of IO devices inside and outside the PROFINET ring, the redundant system in STEP 7 does not recognize the S7-1500 CPUs. If you use S7-1500 CPUs, you must check the maximum number yourself.

The CPU supports H-Sync forwarding and thus may be a node of the PROFINET ring.

The following figure shows a redundant S7-1515R system. An additional PROFINET IO system with a CPU 1516 as IO controller is located downstream of the switch. A separate standard IO device has been assigned to the IO controller.

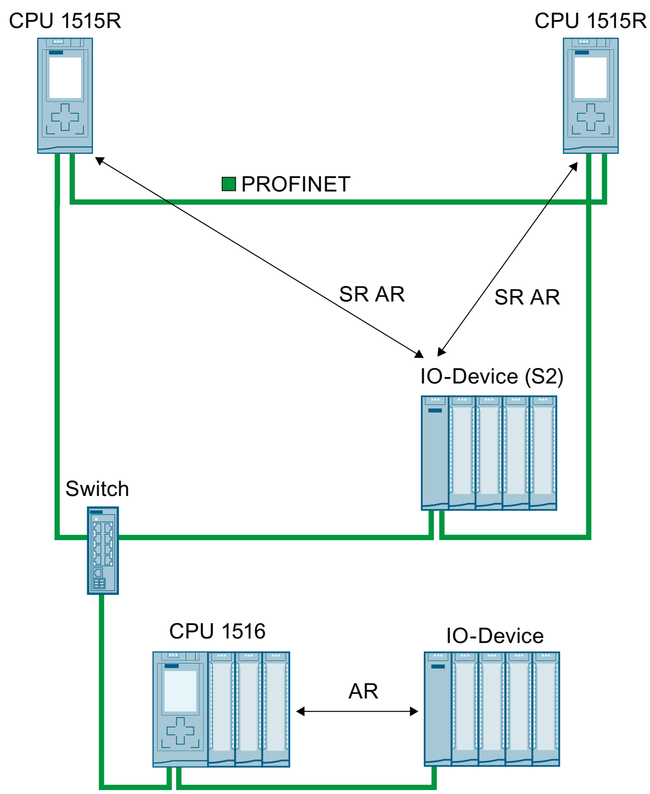


Figure 7-10 Example of a redundant S7-1500R system with an additional PROFINET IO system

Additional IO controllers can also be connected as I-device to the redundant S7-1500R/H system. An I-device is connected like a standard IO device to the redundant S7-1500R/H system.
The redundant S7-1500R/H system supports the use of I-devices only via GSD file and as standard IO device.

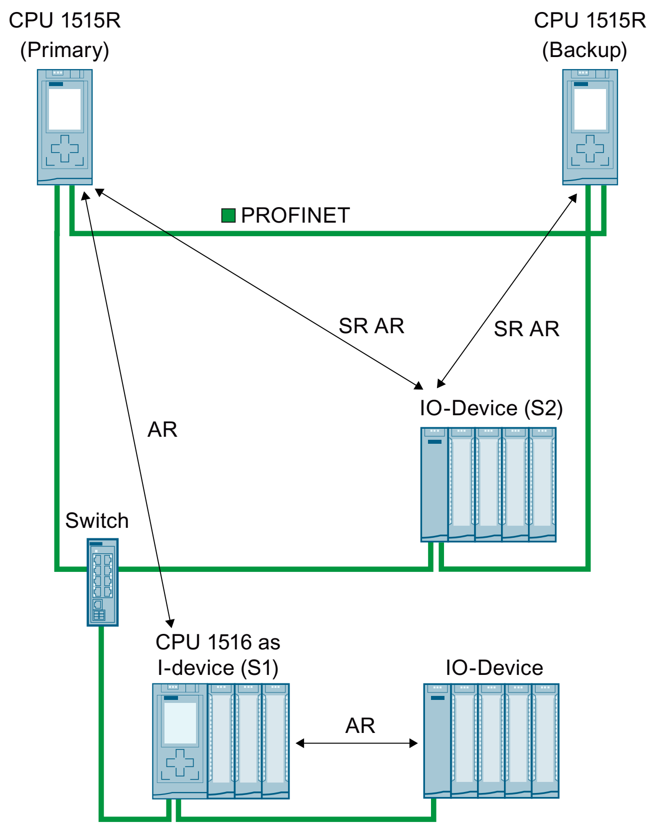


Figure 7-11 I-device downstream of a switch

Connection to another PROFINET IO system via the PN/PN coupler

The redundant S7-1500R/H system cyclically exchanges IO data with another PROFINET IO system via a PN/PN coupler.

In the following figure, the PN/PN coupler connects a redundant S7-1515R system to another PROFINET IO system. The left side of the PN/PN coupler is assigned to the redundant S7-1515R system. The assignment is system-redundant and media-redundant. The right side of the PN/PN coupler is assigned to the CPU 1516 (IO controller). The PROFINET IO system of the CPU 1516 has no redundancy in this example.

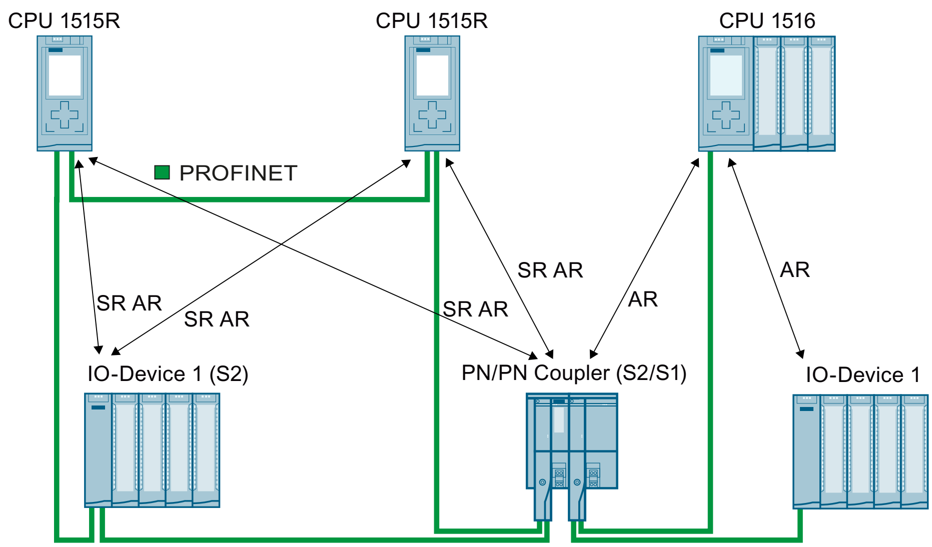


Figure 7-12 Configuration with the PN/PN coupler with one-sided system redundancy (S2/S1)

Connection to an additional redundant S7-1500R/H system via the PN/PN coupler

Two redundant S7-1500R/H systems cyclically exchange IO data via the PN/PN coupler.

In the following figure, the PN/PN coupler connects two redundant S7-1515R systems to each other. Each of the two PN/PN coupler sides are assigned to a redundant S7-1515R system. Each of the two sides of the PN/PN coupler are connected to the respective redundant S7-1515R system in a system-redundant and media-redundant manner.

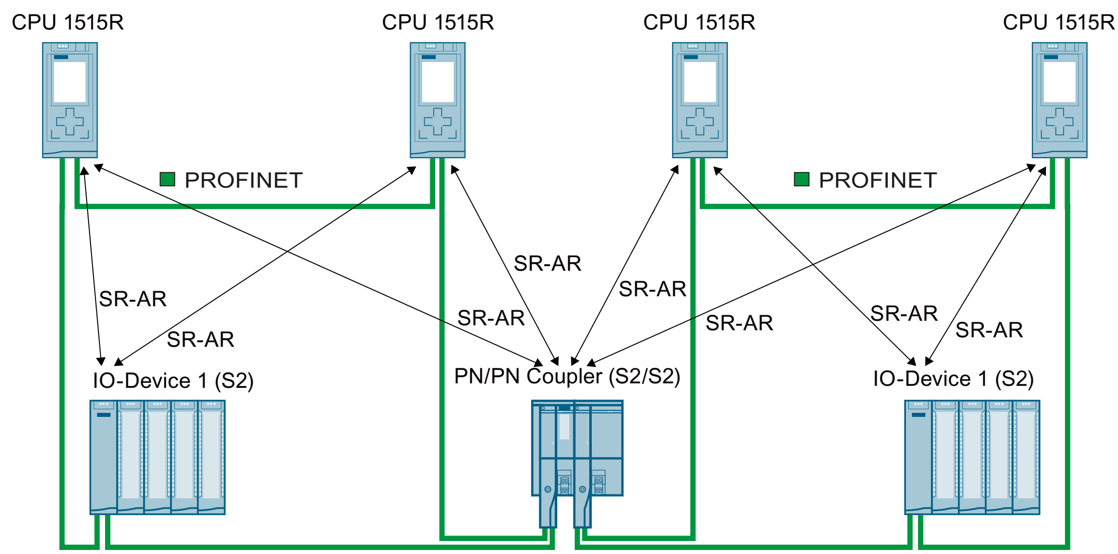


Figure 7-13 Configuration with the PN/PN coupler with two-sided system redundancy (S2/S2)

Reference

You can find more information on the PN/PN coupler in the PN/PN Coupler (https://support.industry.siemens.com/cs/ww/en/view/44319532) manual.

7.9 Assigning IO device to the redundant S7-1500R/H system

Using two examples, this section describes how to assign IO devices to the redundant S7-1500R/H system.

- Procedure for S7-1500R with S2 or S1 devices.
- Procedure for S7-1500H with R1 devices.

Requirements

- Redundant S7-1500R/H system
- IO device:

Redundant system	Switched S1 device (Standard IO device)	IO device with System redundancy S2	IO device with System redundancy R1
S7-1500H as of firmware version V3.0	X	X	X
S7-1500R/H as of firmware version V2.8	X	X	-
S7-1500R/H with firmware version less than V2.8	-	X	-

Procedure for S7-1500R/H with S2/S1 devices

To assign S2 or S1 devices to the redundant S7-1500H system, connect the interface module of the IO device to each S7-1500H CPU.

1. In the network view of STEP 7, select the PROFINET interface of the IO device.
2. Drag-and-drop a line between the PROFINET interface of the IO device and the PROFINET interface X1 of the left CPU.

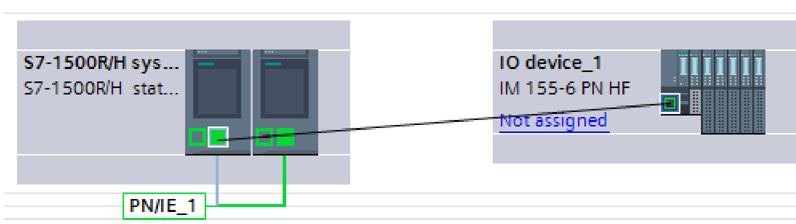


Figure 7-14 Assign IO device to the left CPU

3. Drag-and-drop a line between the PROFINET interface of the IO device and the PROFINET interface X1 of the right CPU.

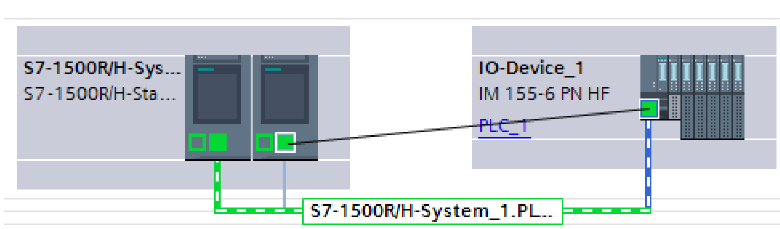


Figure 7-15 Assign IO device to the right CPU

Result: The IO device is connected to the redundant S7-1500R system.

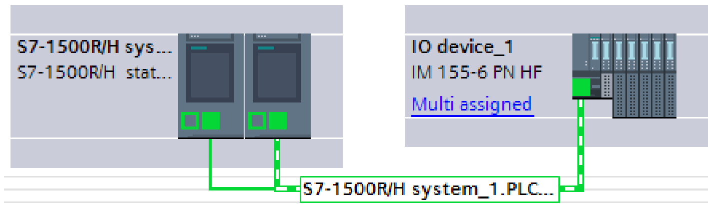


Figure 7-16 S2 device assigned

Procedure for S7-1500H with R1 devices

To assign R1 devices to the redundant S7-1500H system, connect each interface module of the R1 device to an H-CPU.

The left interface module of each R1 device must be connected to the left H-CPU in the network view. The right interface module of each R1 device must be connected to the right H-CPU in the network view.

To do so, proceed as follows:

1. Switch to the network view.
2. Drag-and-drop a line between the PROFINET interface of the left interface module of the ET200SP-R1_1 station and the PROFINET interface X1 of the left H-CPU.
3. Drag-and-drop a line between the PROFINET interface of the right interface module of the ET200SP-R1_1 station and the PROFINET interface X1 of the right H-CPU.
4. Switch to the device view of the ET200SP-R1_1 station and set the watchdog timer for both interface modules. To do this, navigate in the Inspector window to "Properties > PROFINET interface [X1] > Advanced options > Real-time settings > IO cycle".
5. Repeat steps 1 to 4 for the ET200SP-R1_2 station.

   Result: The R1 devices are connected to the S7-1500H redundant system.

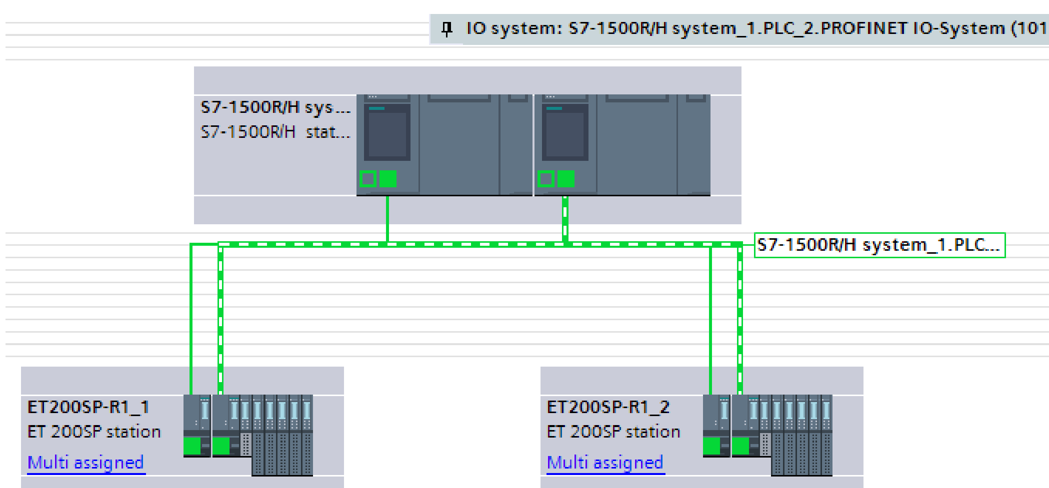


Figure 7-17 R1 devices assigned in the network view with system redundancy

NOTE

Alternative method for assigning IO devices.

For larger projects, we recommend assigning the IO devices as follows:

1. Switch to the network view.
2. Move the cursor over the station you want to assign.
3. Press the right mouse button and select "Assign to new DP master/IO Controller" in the shortcut menu.
4. Select both IO Controllers in the following dialog and then click on "Ok". This assigns the left interface module to the left IO Controller and the right interface module to the right IO Controller.
5. Repeat steps 2 through 4 for additional IO devices or mark several stations simultaneously.

I-device as standard IO device

If you want to assign an I-device to the redundant S7-1500R/H system, you have to include the configured I-device via its exported GSD file.

To operate an I-device as standard IO device on the redundant S7-1500R/H system, always use the GSD file of the I-device in the H-system configuration.

- SIMATIC CPU as I-device
  - First, configure the SIMATIC CPU in STEP 7 as an I-device with all transfer areas.
  - Export the I-device as a GSD file. The GSD export can be found in the properties of the PROFINET interface under "Operating mode > I-device communication > Export Generic System Description file (GSD)".
  - Install the GSD file in STEP 7.
- HMI device as I-device ("Direct key" function)
  - The GSD files for SIMATIC Comfort Panel and SIMATIC Mobile Panel can be found under the link at the end of this section.

Assign the I-device integrated via GSD file to the redundant system S7-1500R/H.

See also

GSD file HMIs (https://support.industry.siemens.com/cs/ww/en/view/73502293)

7.10 Configuring media redundancy (MRP) for a configuration with the redundant S7-1500R/H system

This section describes by means of two examples how to configure media redundancy (MRP) for a configuration with the redundant S7-1500R/H system.

- Configuring MRP for a configuration with S2 devices (ET 200SP) with the redundant S7-1500R system.
- Configuring MRP for a configuration with R1 devices (ET 200SP) with the redundant S7-1500H system.

Requirements

- Redundant S7-1500R/H system.
- All devices in the ring support the media redundancy protocol MRP.
- IO devices are assigned to the redundant S7-1500R/H system.
- The MRP domain "mrpdomain-2" was created in the domain settings (S7-1500H).

Configuring MRP for a configuration with S2 devices (ET 200SP) with the redundant S7-1500R system

Specifying MRP role and MRP domain of the CPUs of the redundant S7-1500R system

As soon as you create a redundant S7-1500R system in STEP 7, STEP 7 automatically assigns the MRP role "Manager (auto)" for the PROFINET interfaces X1 of the two CPUs.

Specifying MRP role and MRP domain for the S2 devices of the redundant S7-1500R system in STEP 7

To specify the media redundancy for additional devices in the ring, follow these steps:

1. In the network view of STEP 7, select the PROFINET interface X1 of one of the two CPUs of the redundant S7-1500R system.
2. In the Inspector window, navigate to "Properties > General > Advanced options > Media redundancy".
3. Click the "Domain settings" button.

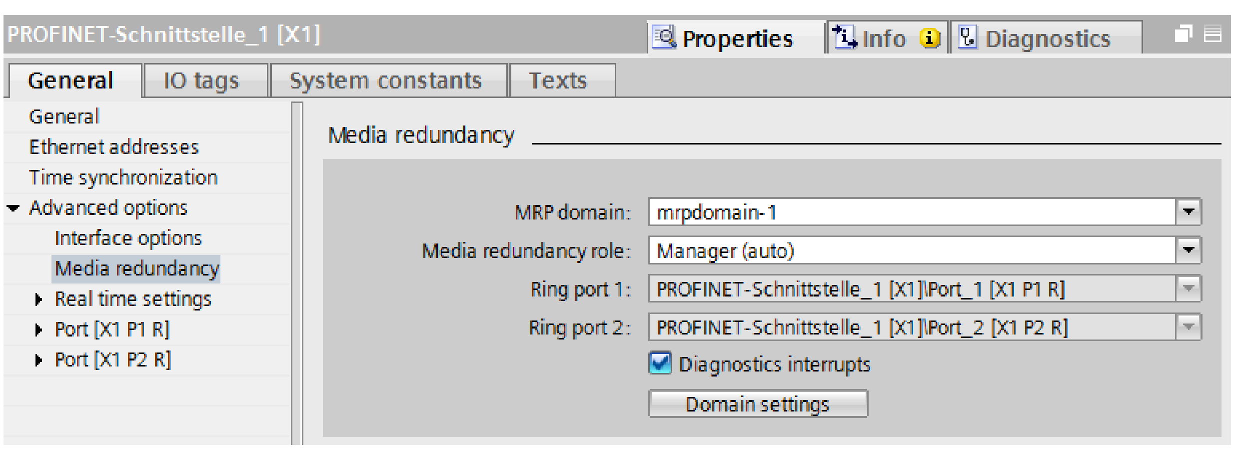


Figure 7-18 S7-1500R: MRP role "Manager (auto)"

In the Inspector window, STEP 7 displays the properties of the MRP domain in which the PROFINET interface X1 of the CPU is located.

4. In the "MRP role" column of the "Devices" table, assign the MRP role "Client" to all the other devices of the ring.

Devices

	PROFINET device name	MRP domain:	MRP role	Ring port 1	Ring port 2
1	io-device_1	mrpdomain-1	Client	Port_1 [X1 P1 R]	Port_2 [X1 P2 R]
2	io-device_2	mrpdomain-1	Client	Port_1 [X1 P1 R]	Port_2 [X1 P2 R]
3	plc_1.profinet-schnittstel...	mrpdomain-1	Manager (auto)	Port_1 [X1 P1 R]	Port_2 [X1 P2 R]

Figure 7-19 S7-1500R: Assigning MRP roles for ring devices

Configuring MRP for a configuration with R1 devices (ET 200SP) with the redundant S7-1500H system

Set MRP role and MRP domain of the CPUs of the redundant S7-1500H system

As soon as you create a S7-1500H redundant system in STEP 7, STEP 7 automatically assigns the MRP role "Not device in the ring" for the PROFINET interfaces X1 of the two CPUs. For the project engineering as PROFINET ring you have to change the MRP role to "Manager (auto)".

To change the MRP role of the redundant system, follow these steps:

1. Switch to the device view of the S7-1500H redundant system.
2. Select the PROFINET interface X1 of the upper H-CPU (mounting rail_0) of the S7-1500H redundant system.
3. In the Inspector window, navigate to "Properties > General > Advanced options > Media redundancy".
4. Change the MRP domain to "mrpdomain-1" (if required) and the media redundancy role for the H-CPU to "Manager (auto)".
5. Select the PROFINET interface X1 of the lower H-CPU of the S7-1500H redundant system.
6. Change the MRP domain to "mrpdomain-2" (if required) and the media redundancy role for the H-CPU to "Manager (auto)".
7. Enable the "Diagnostic interrupts" option.

NOTE

If the "Diagnostic interrupts" option is enabled, diagnostic interrupts are generated at the ring ports in the event of the following errors:

Wiring or port error:

– A neighbor of the ring port does not support media redundancy MRP.
– A ring port is connected to a non-ring port.
– A ring port is connected to the ring port of a different MRP domain.

You can find additional information in the STEP 7 online help.

Specifying MRP role and MRP domain for the R1 devices of the redundant S7-1500H system in STEP 7.

To define the media redundancy for the other devices (IO devices) of the rings, proceed as follows:

1. Switch to the device view of the left ET 200SP (ET200SP-R1_1).
2. Click on the PROFINET interface X1 of the left IM 155-6 PN R1.
3. Change the MRP domain to "mrpdomain-1" (if required) and the media redundancy role to "Client".
4. Click on the PROFINET interface X1 of the right IM 155-6 PN R1.
5. Change the MRP domain to "mrpdomain-2" and the media redundancy role to "Client".
6. Switch to the device view of the right ET 200SP (ET200SP-R1_2) and repeat steps 2 to 5.

NOTE

Alternative method for assigning MRP role and MRP domain.

1. Switch to the network view.
2. Click the connection PN/IE_1.
3. In the Inspector window, navigate to "Properties > General > MRP domains".
4. Scroll down to the "Devices" section.

- Under "PROFINET IO system", select the controller interface.
- Under "Devices" you configure the respective MRP domain and the MRP role.

7.11 MRP interconnection with the redundant system S7-1500R/H

- The S7-1500H CPUs support system redundancy R1 as of firmware version V3.0.
- The S7-1500R/H CPUs support MRP interconnection as of firmware version V2.9.

MRP interconnection enables the redundant data exchange via S7-1500R/H across two or more MRP rings.

Requirements for S7-1500R/H

In addition to the requirements for S7-1500, the following requirements are in effect for S7-1500R/H:

- For S7-1500R/H: The two CPUs of the redundant system are located in the same ring (as of firmware version V2.9).
- For S7-1500H: The two CPUs of the redundant system are located in one or two rings (as of firmware version V3.0).
- You have configured the media redundancy role of the two CPUs as "Manager (auto)" in the ring with the two R/H CPUs.

  You have configured the following in the other rings:
  - For one or more PROFINET devices the media redundancy role "Manager (auto)" or
  - For exactly one PROFINET device the media redundancy role "Manager"

  You have assigned the MRP client role to the other devices in all rings.

Topology with S2 devices

Topology with two rings, using the example of S7-1500R

The following figure shows the redundant coupling based on the example of S7-1500R in 2 rings.

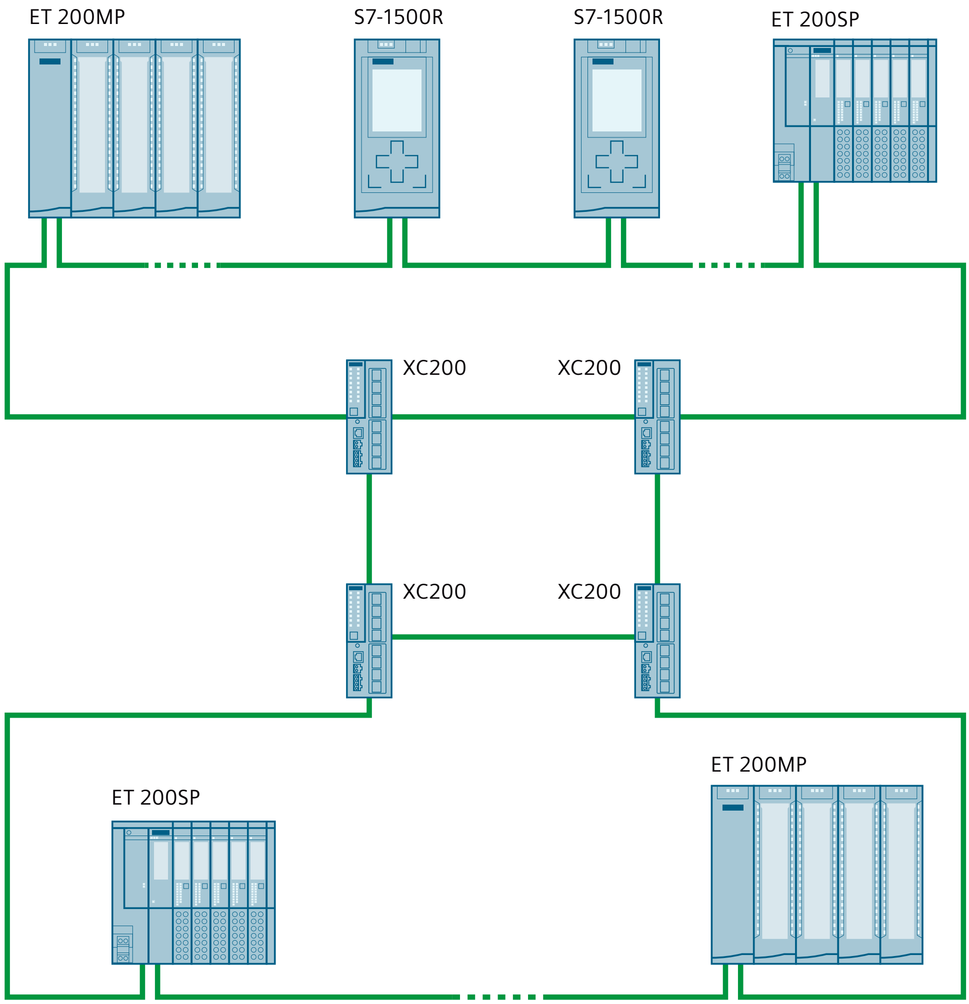


Figure 7-20 Example: Redundant connection of S7-1500R in 2 rings with MRP interconnection

Topology with two rings, using the example of S7-1500H

The figure below shows the redundant coupling, using the example of S7-1500H in two rings.

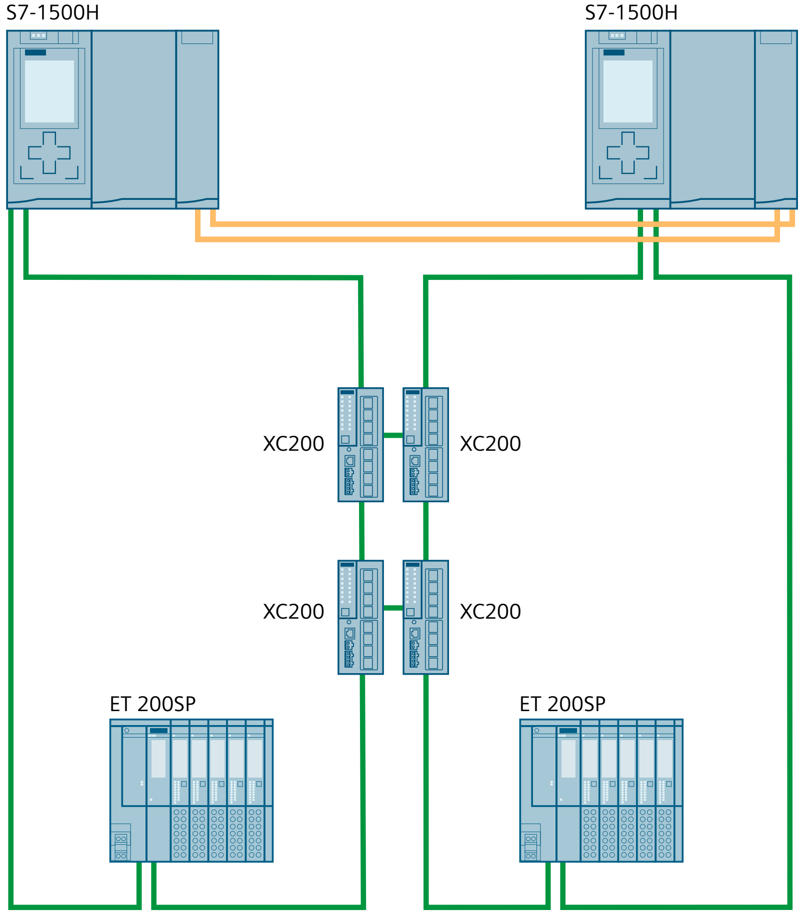


Figure 7-21 Example: Redundant connection of S7-1500H in two rings with MRP interconnection

Topology with multiple rings, using the example of S7-1500H

The figure below shows the redundant coupling, using the example of S7-1500H in four rings.

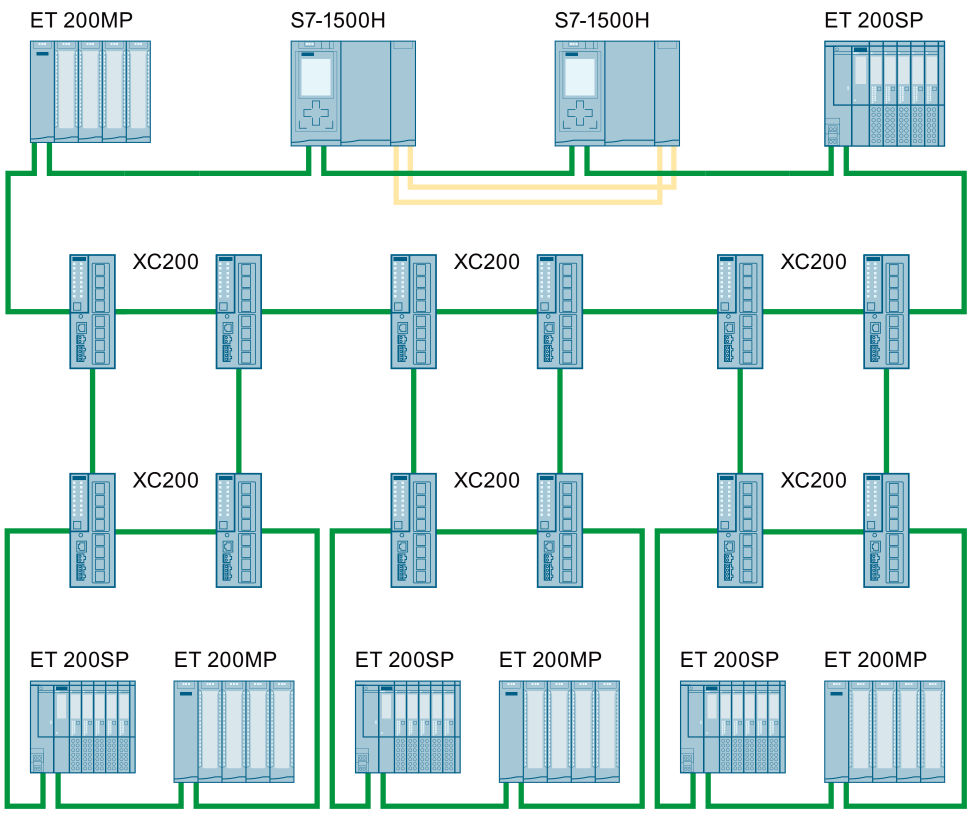


Figure 7-22 Example: Redundant coupling, using the example of S7-1500H in four rings

Topology with R1 devices

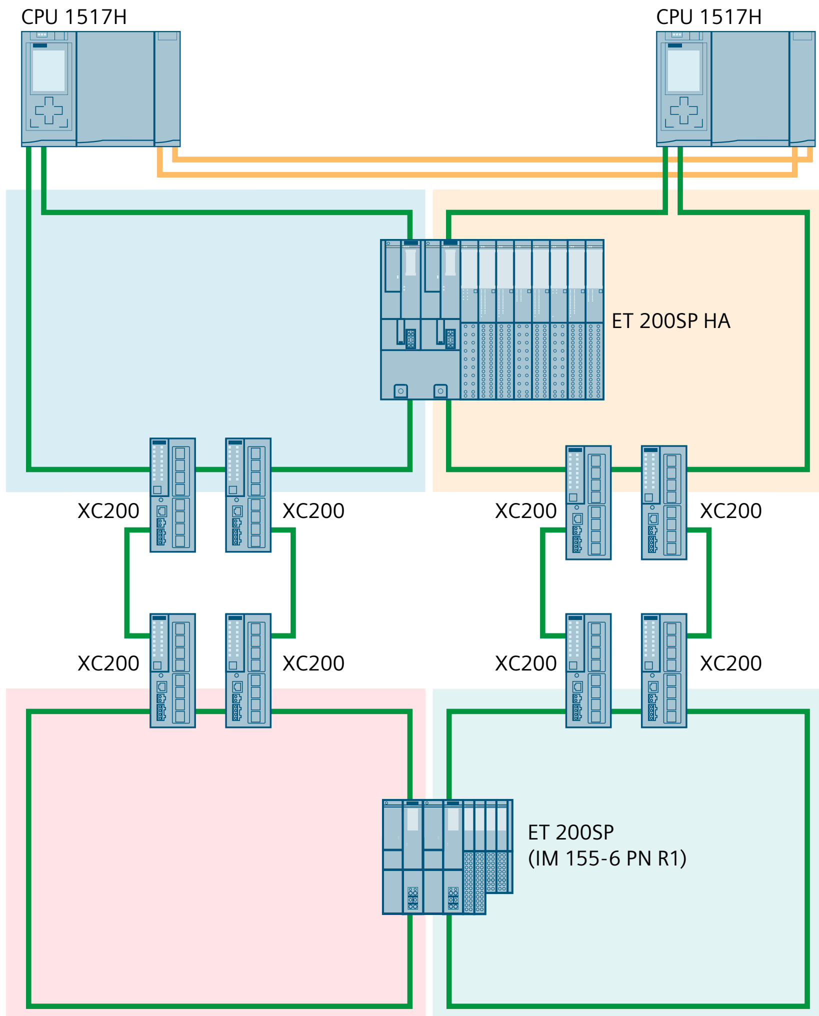


Figure 7-23 Example: S7-1500H configuration with R1 devices and switches with MRP interconnection in the PROFINET ring

Additional information

You can find all information that generally applies to MRP interconnection and thus also to redundant S7-1500R/H systems with MRP interconnection, in the section MRP interconnection (Page 193), for example, constraints, principle of operation, configuration and times.

Tool for setting the watchdog timer

For the correct setting of the watchdog timer, you can use the "S7-1500R/H AddIn (https://support.industry.siemens.com/cs/de/en/view/109769093)" which is available on the Internet and can also be used for MRP interconnections.

Glossary

API

API (**A**pplication **P**rocess **I**dentifier) is a parameter, the value of which specifies the IO data processing process (application).

The PROFINET standard IEC 61158 assigns profiles to specific APIs (PROFIdrive, PROFIsave), which are defined by the PROFINET user organization.

The standard API is 0.

Application

An application is a program that runs directly on the MS-DOS/Windows operating system. An application on the PG is STEP 7, for example.

AR

The AR (**A**pplication **R**elation) comprises all the communication relations between IO controller and IO device (e.g. IO data, data records, interrupts).

AR is also an addressing level for diagnostic data records.

Asset

An asset is a component of a machine or plant This can be either hardware or software/firmware.

Automation system

Programmable logic controller for the open-loop and closed-loop control of process chains of the process engineering industry and manufacturing technology. The automation system consists of different components and integrated system functions according to the automation task.

Backplane bus

The backplane bus is a serial data bus for module intercommunication and the distribution of the necessary power to the modules. Bus connectors interconnect the modules.

Backup AR

The backup AR is the system redundancy AR between an IO device with system redundancy S2 and the backup CPU of the redundant S7-1500R/H system.

Backup CPU

If the R/H system is in RUN-Redundant system state, the primary CPU controls the process. The backup CPU synchronously executes the user program and can take over the process control on failure of the primary CPU.

Bus

A bus is a transfer medium that interconnects several devices. Data transmission can be performed electrically or via optical fibers, either in series or in parallel.

Communications Cycle and Reservation of Transmission Bandwidth

PROFINET IO is a scalable real-time communications system based on the Layer 2 protocol for Fast Ethernet. With the RT transmission method, two real-time-support performance levels are available for time-critical process data and IRT for high-accuracy and also isochronous processes .

Communications processor

Communication processors are modules used for point-to-point and bus connections.

Configuring

Selecting and putting together individual components of an automation system or installing necessary software and adapting it for a specific application (for example, by configuring the modules).

Consistent data

Data which belongs together in terms of content and must not be separated is known as consistent data.

For example, the values of analog modules must always be handled as a whole, that is, the value of an analog module must not be corrupted as a result of reading out at two different points in time.

CP

→ Communications processor

CPU

Central **P**rocessing **U**nit - Central module of the S7 automation system with a control and arithmetic unit, memory, operating system and interface for programming device.

DCP

DCP (**D**iscovery and Basic **C**onfiguration **P**rotocol). Enables the assignment of device parameters (for example, the IP address) using manufacturer-specific configuration/programming tools.

Default router

The default router is used when data has to be forwarded via TCP/IP to a partner located outside the local network.

Determinism

Determinism means that a system responds in a predictable (deterministic) manner.

Device

In the PROFINET environment, "device" is the generic term for:

- Automation systems (for example, PLC, PC)
- Distributed I/O systems
- Field devices (for example, PLC, PC, hydraulic devices, pneumatic devices)
- Active network components (for example, switches, routers)
- Gateways to PROFIBUS, AS interface or other fieldbus systems

Device name (PROFINET device name)

In the PROFINET IO environment, the device name is a unique identifier for the PROFINET IO interface of a device.

DHCP

DHCP (**D**ynamic **H**ost **C**onfiguration **P**rotocol). Protocol that can be used to assign IP addresses (and other important startup parameters in the Internet environment).

Diagnostics

→ System diagnostics

Diagnostics buffer

The diagnostics buffer represents a backup memory in the CPU, used to store diagnostics events in their order of occurrence.

Diagnostics error interrupt

Modules capable of diagnostics operations report detected system errors to the CPU by means of diagnostics interrupts.
→ *See also CPU*

Direct data exchange

In the case of direct data exchange, an S7-1500 CPU provides cyclic user data from the I/O area to one or more partners.
The direct data exchange function enables deterministic, isochronous I/O communication between multiple S7-1500 CPUs.

DP master

A master which behaves in accordance with EN 50170, Part 3, is known as a DP master.
→ *See also Master*

DP slave

A slave operated on PROFIBUS with PROFIBUS DP protocol and in accordance with EN 50170, Part 3 is referred to as DP slave.
→ *See also Slave*

DPV1

The designation DPV1 refers to the functional extension of acyclic services (to include new interrupts, for example) provided by the DP protocol. The DPV1 functionality is integrated in IEC 61158/EN 50170, Volume 2, PROFIBUS.

Duplex

Half duplex: One channel is available for alternating exchange of information.

Full duplex: Two channels are available for simultaneous exchange of information in both directions.

Ethernet

Ethernet is an international standard technology for local area networks (LAN) based on frames. It defines types of cables and signaling for the physical layer and packet formats and protocols for media access control.

GSD file

As a Generic Station Description, this file contains all properties of a PROFINET device that are necessary for its configuration in XML format.

H-Sync Forwarding

H-Sync Forwarding enables a PROFINET device with MRP to forward synchronization data (synchronization frames) of a S7-1500R redundant system only within the PROFINET ring.

In addition, H-Sync Forwarding forwards the synchronization data even during reconfiguration of the PROFINET ring. H-Sync Forwarding avoids a cycle time increase if the PROFINET ring is interrupted.

S7-1500R: H-Sync Forwarding is recommended for all PROFINET devices with only two ports in the PROFINET ring. All PROFINET devices with more than two ports (e.g. switch) in the PROFINET ring must support H-Sync forwarding.

S7-1500H: H-Sync Forwarding is not relevant for redundant S7-1500H systems.

I-device

The "I-device" (intelligent IO device) functionality of a CPU facilitates data exchange with an IO controller, for example, as intelligent preprocessing unit of sub-processes. In its role as an IO device, the I-device is accordingly integrated into a "higher-level" IO controller.

Industrial Ethernet

Industrial Ethernet is a guideline for installing an Ethernet in an industrial environment. The biggest difference from standard Ethernet is the mechanical current carrying capacity and noise immunity of the individual components.

Interrupt

An interrupt is an event that causes the operating system of an S7-CPU to automatically call an assigned organization block (interrupt OB) in which the user can program the desired reaction.

Interrupt, diagnostics

→ Diagnostics error interrupt

IP address

To allow a PROFINET device to be addressed as a device on Industrial Ethernet, this device also requires an IP address that is unique within the network. Example: An IPv4 address consists of 4 decimal numbers with the value range 0 to 255. The decimal numbers are separated by periods.

The IP address is made up of the following components:

- Address of the network
- Address of the device (generally called the host or network node).

IRT

IRT is a synchronized communication protocol for cyclic exchange of IRT data between PROFINET devices. A reserved bandwidth is available in the send cycle for IRT data. The reserved bandwidth ensures that the IRT data can be transferred at reserved synchronized intervals, without being influenced also by higher other network loads (such as TCP/IP communication, or additional real-time communication).

LAN

Local **A**rea **N**etwork - a local network to which multiple computers within a company are connected. The LAN therefore has a limited geographical span and is only available to the company or institution.

Line depth

Designates the number of external switches or integrated switches interconnected in series.

MAC address

Worldwide unique device identification for all Ethernet devices. It is assigned by the manufacturer and has a 3-byte vendor ID and 3-byte device ID as a consecutive number.

Every PROFINET device is assigned a worldwide unique device identifier at the factory. This 6-byte long device identifier is the MAC address.

Maintenance Required / Maintenance Demanded

A "Maintenance required" system message can be defined for different wear parameters and an inspection of a component can be recommended, for example, when a certain number of operating hours has elapsed.

The "Maintenance demanded" alarm is sent when the part involved needs to be replaced within a foreseeable period.

(Example printer: The maintenance demanded alarm is sent when the ink/printer cartridges have to be replaced immediately.)

Master

Higher-level, active participant in the communication/on a PROFIBUS subnet. It has rights to access the bus (token), sends data and requests it.

Media redundancy

The so-called **M**edia **R**edundancy **P**rotocol (MRP) enables the configuration of redundant networks. Redundant transmission links (ring topology) ensure that an alternating communication path is made available if a transmission link fails. The PROFINET devices that are a part of this redundant network form an MRP domain.

Micro Memory Card (MMC)

Micro Memory Cards are memory media for CPUs and CPs. Their only difference to the SIMATIC memory card is the smaller size. S7-1500 and ET 200SP-CPUs use SIMATIC memory cards.

MPI

The multipoint interface (**M**ulti **P**oint **I**nterface,MPI) represents the programming device interface of SIMATIC S7. It enables multiple devices (programming devices, text-based displays, OPs) to be operated simultaneously by one or more CPUs. Each device is identified by its unique (MPI) address.

MRP

MRP (**M**edia **R**edundancy **P**rotocol) describes the ring redundancy according to IEC 61158 and IEC 62439.

MRP interconnection

The MRP interconnection procedure is an extension of MRP. MRP interconnection enables the redundant coupling of two or more rings with MRP in PROFINET networks.

MRPD

MRPD (**M**edia **R**edundancy with **P**lanned **D**uplication of Frames) based on IRT and MRP. To realize media redundancy with short update times, the PROFINET devices participating in the ring send their data in both directions. The devices receive this data at both ring ports so that there is no reconfiguration time.

Network

A network consists of one or more interconnected subnets with any number of devices. Several networks can exist alongside each other.

OB

→ Organization block

OPC

OPC (OLE for Process Control) refers to a standard interface for communication in automation technology.

Operating states

Operating states describe the behavior of a single CPU at a specific time. The CPUs of the SIMATIC standard systems feature the STOP, STARTUP and RUN operating states.
The primary CPU of the S7-1500R/H redundant system has the operating states STOP, STARTUP, RUN, RUN-Syncup and RUN-Redundant. The backup CPU has the operating states STOP, SYNCUP and RUN-Redundant.

Organization block

Organization blocks (OBs) form the interface between the CPU operating system and the user program. The order in which the user program is executed is defined in the organization blocks.

Parameter

1. Tag of a STEP 7 code block:
2. Tag used to set one or several characteristics of a module

In delivery state, every module has practical basic settings, which can be modified by configuration in STEP 7.
There are static and dynamic parameters

PG

→ Programming device

PLC

→ Programmable logic controller

Primary AR

The primary AR is the system redundancy AR between an IO device with system redundancy S2 and the primary CPU of the redundant S7-1500R/H system.

Primary CPU

If at least one R/H CPU is in RUN state, the primary CPU is the controlling CPU. The primary CPU controls the process (productive data).
The user program is executed identically in the primary CPU and the backup CPU.

Prioritized startup

Prioritized startup denotes the PROFINET functionality for accelerating the startup of IO devices operated on a PROFINET IO system with RT and IRT communication. It reduces the time that the correspondingly configured IO devices require in order to return to cyclic user data exchange in the following cases:

- After the power supply has returned
- After a station has returned
- After activation of IO devices

Process image (I/O)

The CPU transfers the values from the input and output modules to this memory area. At the start of the cyclic program, the CPU transfers the process image output as a signal state to the output modules. The CPU then transfers the signal states of the input modules to the process image of the inputs. Then the CPU executes the user program.

PROFIBUS

Process **Fi**eld **Bus** - European Fieldbus standard.

PROFIBUS device

A PROFIBUS device has at least one PROFIBUS interface with an electrical (RS485) or optical (polymer optical fiber, POF) interface.

PROFIBUS DP

A PROFIBUS with DP protocol that complies with EN 50170. DP stands for distributed I/O (fast, real-time capable, cyclic data exchange). From the perspective of the user program, the distributed I/Os are addressed in exactly the same way as the centralized I/Os.

PROFIBUS Users Organization

Technical committee dedicated to the definition and development of the PROFIBUS and PROFINETstandard. Additional information can be found on the Internet (Page %getreference).

PROFIenergy

Function for energy saving in the process, for example, during pause times by briefly switching off the entire system via standardized PROFIenergy commands.

PROFINET

Open component-based industrial communications system based on Ethernet for distributed automation systems. Communication technology promoted by the PROFIBUS Users Organization.

PROFINET device

A PROFINET device always has a PROFINET interface (electrical, optical, wireless). A lot of devices also have a PROFIBUS DP interface to connect PROFIBUS devices.

PROFINET IO

Communication concept for the realization of modular, distributed applications within the scope of PROFINET.
PROFINET IO is based on switched Ethernet with full-duplex operation and a bandwidth of 100 Mbps.

PROFINET IO Controller

Device used to address the connected IO devices. This means that the IO controller exchanges input and output signals with assigned field devices. The IO controller is often the controller on which the automation program runs.

PROFINET IO Device

A distributed field device that is assigned to one of the IO controllers (e.g. remote IO, valve terminals, frequency converters, switches)

PROFINET IO System

PROFINET IO controller with assigned PROFINET IO devices.

PROFINET ring

Structure of a network. Common structures include:

- Linear bus topology
- Ring topology
- Star topology
- Tree topology

Programmable logic controller

Programmable logic controllers (PLCs) are electronic controllers whose function is stored as a program in the control unit. The structure and wiring of the device does not therefore depend on the controller's function. A programmable logic controller is structured like a computer. It consists of a CPU with memory, input/output modules and an internal bus system. The IOs and the programming language are oriented to the requirements of the control technology.

Programming device

Programming devices are essentially compact and portable PCs which are suitable for industrial applications. They are identified by a special hardware and software configuration for programmable logic controllers.

Proxy

The PROFINET device with proxy functionality is the substitute for a PROFIBUS device on Ethernet. The proxy functionality allows a PROFIBUS device to communicate not only with its master but also with all devices on PROFINET.

With PROFINET, existing PROFIBUS systems can be integrated into the PROFINET communication with the aid of an IE/PB link, for example. The IE/PB Link PN IO then handles communication via PROFINET on behalf of the PROFIBUS components.

In this way, you can link both DPV0 and DPV1 slaves to PROFINET.

Real-time

Real-time means that a system processes external events within a defined time.

Real-time communication

Group error for RT and IRT.

PROFINET uses its own real-time channel (RT) rather than TCP/IP for communication of time-critical IO user data.

Redundant systems

Redundant systems are characterized by the fact that important automation components are present multiple times (redundantly). Process control is maintained if a redundant component fails.

Remote Procedure Call (RPC)

A protocol for exchanging parameter assignment and diagnostics data between the IO controller and IO device.

Router

A router interconnects two subnets. A router works in a similar way to a switch. With a router, however, you can also specify which communication devices may communicate via the router and which may not. The communication devices on various sides of a router can only communicate with one another if you have explicitly enabled communication between these devices via the router. Real-time data cannot be exchanged beyond subnet boundaries.

RT

PROFINET IO with **R**eal **T**ime communication (RT) is the optimal transmission method for time-critical applications in factory automation. PROFINET IO frames are prioritized over standard frames in accordance with IEEE802.1Q. This ensures the required determinism in the automation technology.

Security

Generic term for all the measures taken to protect against

- Loss of confidentiality due to unauthorized access to data
- Loss of integrity due to manipulation of data
- Loss of availability due to the destruction of data

Send clock

Period between two consecutive intervals for IRT or RT communication. The send clock is the shortest possible transmit interval for exchanging data.

SIMATIC

The term denotes Siemens AG products and systems for industrial automation.

SIMATIC NET

Siemens Industrial Communication division for Networks and Network Components.

SIMATIC PC Station

A "PC station" is a PC with communication modules and software components within a SIMATIC automation solution.

SIMATIC Memory Card (SMC)

→ Micro Memory Card (MMC)

Slave

A slave can only exchange data after being requested to do so by the master.
→ *See also DP slave*

SNMP

The network management protocol SNMP (**S**imple **N**etwork **M**anagement **P**rotocol) uses the wireless UDP transport protocol. It consists of two network components, similar to the client/server model. The SNMP manager monitors the network nodes and the SNMP agents collect the various network-specific information in the individual network nodes and stores it in a structured form in the MIB (**M**anagement **I**nformation **B**ase). This information allows a network management system to run detailed network diagnostics.

STEP 7

STEP 7 is an Engineering System and contains programming software for the creation of user programs for SIMATIC S7 controllers.

Subnet

All the devices interconnected by switches are nodes of the same network or subnet. All the devices in a subnet can communicate directly with each other.

All devices in the same subnet have the same subnet mask.

A subnet is physically restricted by a router.

Subnet mask

The bits set in the subnet mask decide the part of the IP address that contains the address of the network.

In general, the following applies:

- The network address is obtained from the AND operation of the IP address and subnet mask.
- The device address is obtained from the AND NOT operation of the IP address and subnet mask.

Switch

Network components used to connect several terminal devices or network segments in a local network (LAN).

Switched S1 device

The "Switched S1 device" function of the CPU enables operation of standard IO devices on the S7-1500R/H redundant system.

PROFINET communication runs on an AR between the primary CPU and the standard IO device. When replacing the primary CPU, the standard IO device is briefly disconnected from the S7-1500R/H redundant system until the new primary CPU has set up an AR to the standard IO device.

Sync domain

All PROFINET devices to be synchronized with IRT via PROFINET IO must belong to a sync domain.

The sync domain consists of exactly one sync master and at least one sync slave. An IO controller or switch generally handles the role of the sync master.

Non-synchronized PROFINET devices are not part of a sync domain.

System diagnostics

System diagnostics refers to the detection, evaluation, and signaling of errors that occur within the automation system, for example, programming errors or module failures. System errors can be indicated by LEDs or in STEP 7.

System redundancy AR

In a redundant system, an IO device with system redundancy S2 has a system redundancy AR with each of the two CPUs (IO controllers).

System states

The system states of the redundant S7-1500R/H system result from the operating states of the primary and backup CPUs. The concept of the system state is used to obtain a simplified expression that characterizes the simultaneously occurring operating states of the two CPUs. The following system states are available for the redundant S7-1500R/H system: STOP, STARTUP, RUN-Solo, SYNCUP and RUN-Redundant.

TCP/IP

The Ethernet itself is only a transport system for data - similar to a highway, which is a transport system for people and goods. The actual data transport is performed by so-called protocols - similar to cars and trucks, which transport people and goods on the highway.

The two basic protocols TCP (**T**ransmission **C**ontrol **P**rotocol) and IP (**I**nternet **P**rotocol) - TCP/IP for short - perform the following tasks:

1. The data is broken down into packets at the sender.
2. The packets are transported over the Ethernet to the correct recipient.
3. At the recipient, the packets are reassembled in the correct order.
4. Corrupt packets continue to be sent until they are received correctly.

Most higher-level protocols use TCP/IP to perform their duties. **H**yper **T**ext **T**ransfer **P**rotocol (HTTP), for example transfers documents written in **H**yper **T**ext **M**arkup **L**anguage (HTML) in the **W**orld **W**ide **W**eb (WWW). This technique is what enables you to view web pages in your Internet browser in the first place.

Topology configuration

All the interconnected ports of the PROFINET devices in the STEP 7 project and their relationships to each other.

Twisted-pair

Fast Ethernet via twisted-pair cables is based on the IEEE 802.3u standard (100 Base-TX). The transmission medium is a shielded 2x2 twisted-pair cable with an impedance of 100 Ω (AWG 22). The transmission characteristics of this cable must meet the requirements of category 5.

The maximum length of the connection between the terminal and the network component must not exceed 100 m. The connectors are designed according to the 100Base-TX standard with the RJ45 connector system.

Update time

An IO device / IO controller in the PROFINET IO system is supplied cyclically with new data from the IO controller / IO device within this time interval. The update time can be configured separately for each IO device and determines the interval at which data is sent from the IO controller to the IO device (outputs) as well as from the IO device to the IO controller (inputs).

User program

In SIMATIC, we distinguish between the CPU operating system and user programs. The user program contains all instructions, declarations and data by which a plant or process can be controlled. The user program is assigned to a programmable module (for example, CPU, CM) and can be structured in smaller units.

WAN

A network that extends beyond LAN boundaries and enables, for example, intercontinental network communication. Legal rights do not belong to the user but to the provider of the communication network.

XML

XML (E**x**tensible **M**arkup **L**anguage) is a flexible, easy to understand and easy to learn data description language. Information is exchanged using readable XML documents. These include continuous text supplemented by structure information.

Index

R

S