SIEMENS

Communicating with PLCs (Basic Panels) 1

SIMATIC

WinCC WinCC Basic V13 SP2 – Communication

System Manual

Online help printout

03/2017
Online help printout

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
Division Digital Factory
Postfach 48 48
90026 NÜRNBERG
GERMANY

Document order number: Online help printout
℗ 05/2017 Subject to change

Table of contents

1 Communicating with PLCs (Basic Panels)....................9

1.1 Basics of communication (Basic Panels)....................9
1.1.1 Communication between devices (Basic Panels)....................9
1.1.2 Devices and networks in the automation system (Basic Panels)....................10
1.1.3 Data exchange using tags (Basic Panels)....................13
1.1.4 Data exchange using area pointers (Basic Panels)....................14
1.1.5 Communication drivers (Basic Panels)....................14

1.2 Editors for communication (Basic Panels)....................15
1.2.1 "Devices & networks" editor (Basic Panels)....................15
1.2.2 Network view (Basic Panels)....................16
1.2.3 Network data (Basic Panels)....................19
1.2.4 Diagnostics of online connections (Basic Panels)....................21
1.2.5 Device view (Basic Panels)....................23
1.2.6 Topology view (Basic Panels)....................25
1.2.7 Inspector window (Basic Panels)....................27
1.2.8 Hardware catalog (Basic Panels)....................29
1.2.9 Information on hardware components (Basic Panels)....................31

1.3 Networks and connections (Basic Panels)....................32
1.3.1 SIMATIC communication networks (Basic Panels)....................32
1.3.1.1 Communication networks (Basic Panels)....................32
1.3.1.2 PROFINET and Ethernet (Basic Panels)....................34
1.3.1.3 PROFIBUS (Basic Panels)....................34
1.3.1.4 MPI (Basic Panels)....................35
1.3.1.5 PPI (Basic Panels)....................36
1.3.2 Configuring networks and connections (Basic Panels)....................37
1.3.2.1 Networking devices (Basic Panels)....................37
1.3.2.2 Configuring an integrated connection in the "Devices & Networks" editor (Basic Panels)....................38
1.3.2.3 Special considerations of the "Devices & Networks" editor (Basic Panels)....................40
1.3.2.4 Configuring a non-integrated connection in the "Connections" editor (Basic Panels)....................41
1.3.2.5 Configuring a routed connection in the "Devices & Networks" editor (Basic Panels)....................44

1.4 Data exchange (Basic Panels)....................47
1.4.1 Data exchange using tags (Basic Panels)....................47
1.4.1.1 Basics of tags (Basic Panels)....................47
1.4.1.2 Overview of HMI tag tables (Basic Panels)....................48
1.4.1.3 External tags (Basic Panels)....................49
1.4.1.4 Addressing external tags (Basic Panels)....................51
1.4.1.5 Internal Tags (Basic Panels)....................53
1.4.2 Data exchange using area pointers (Basic Panels)....................54
1.4.2.1 Basic information on area pointers (Basic Panels)....................54
1.4.2.2 Area pointers for connections (Basic Panels)....................55
1.4.2.3 Use of area pointers (Basic Panels)....................56
1.4.2.4 Accessing data areas (Basic Panels)....................58
1.4.2.5 Configuring area pointers (Basic Panels)....................58

1.5 Device dependency (Basic Panels)....................61

1.5.1 Basic Panel (Basic Panels)....61
1.5.1.1 Communication drivers (Basic Panels)....61
1.5.1.2 Interfaces of the Basic Panels (Basic Panels)....65
1.5.1.3 Area pointers for Basic Panels (Basic Panels)....66

1.6 Communicating with SIMATIC S7 1500 (Basic Panels)....67
1.6.1 Communication with SIMATIC S7 1500 (Basic Panels)....67
1.6.2 Communication via PROFINET (Basic Panels)....68
1.6.2.1 Configuring an HMI connection (Basic Panels)....68
1.6.2.2 PROFINET parameters (Basic Panels)....71
1.6.2.3 Setting port options (Basic Panels)....81
1.6.3 Communication via PROFIBUS (Basic Panels)....84
1.6.3.1 Configuring an HMI connection (Basic Panels)....84
1.6.3.2 PROFIBUS parameters (Basic Panels)....88
1.6.4 Data exchange (Basic Panels)....93
1.6.4.1 Area pointers (Basic Panels)....93
1.6.4.2 Trends (Basic Panels)....107
1.6.4.3 Alarms (Basic Panels)....109
1.6.5 Performance features of communication (Basic Panels)....112
1.6.5.1 SIMATIC S7-1500 device dependency (Basic Panels)....112
1.6.5.2 Valid data types for SIMATIC S7 1500 (Basic Panels)....113
1.6.6 Configuring connections in the "Connections" editor (Basic Panels)....114
1.6.6.1 Connection parameters (Basic Panels)....114
1.6.6.2 Ethernet parameters (Basic Panels)....116
1.6.6.3 PROFIBUS parameters (Basic Panels)....117

1.7 Communicating with SIMATIC S7 1200 (Basic Panels)....118
1.7.1 Communication with SIMATIC S7 1200 (Basic Panels)....118
1.7.2 Communication via PROFINET (Basic Panels)....119
1.7.2.1 Communication via PROFINET (Basic Panels)....119
1.7.2.2 Configuring an HMI connection via PROFINET (Basic Panels)....120
1.7.2.3 PROFINET parameters (Basic Panels)....123
1.7.2.4 Setting port options (Basic Panels)....134
1.7.3 Communication via PROFIBUS (Basic Panels)....137
1.7.3.1 Communication via PROFIBUS (Basic Panels)....137
1.7.3.2 Configuring an HMI connection via PROFIBUS (Basic Panels)....138
1.7.3.3 PROFIBUS parameters (Basic Panels)....140
1.7.4 Data exchange (Basic Panels)....147
1.7.4.1 Data exchange using area pointers (Basic Panels)....147
1.7.4.2 Trends (Basic Panels)....164
1.7.4.3 Alarms (Basic Panels)....164
1.7.5 Performance features of communication (Basic Panels)....167
1.7.5.1 S7-1200 device dependency (Basic Panels)....167
1.7.5.2 Valid data types for SIMATIC S7-1200 (Basic Panels)....168
1.7.6 Creating connections in the "Connections" editor (Basic Panels)....169
1.7.6.1 Creating a PROFINET connection (Basic Panels)....169
1.7.6.2 Creating a PROFIBUS DP connection (Basic Panels)....172
1.7.6.3 Parameters for the connection (Basic Panels)....174

1.8 Communicating with SIMATIC S7 300/400 (Basic Panels)....177
1.8.1 Communication with SIMATIC S7 300/400 (Basic Panels)....177
1.8.2 Communication via PROFINET (Basic Panels)....178
1.8.2.1 Communication via PROFINET (Basic Panels)....178
1.8.2.2 Configuring an HMI connection via PROFINET (Basic Panels)....179

1.8.2.3 PROFINET parameters (Basic Panels)....181
1.8.2.4 Setting port options (Basic Panels)....188
1.8.3 Communication via PROFIBUS (Basic Panels)....191
1.8.3.1 Communication via PROFIBUS (Basic Panels)....191
1.8.3.2 Configuring an HMI connection via PROFIBUS (Basic Panels)....192
1.8.3.3 PROFIBUS parameters (Basic Panels)....194
1.8.4 Communication via MPI (Basic Panels)....199
1.8.4.1 Communication via MPI (Basic Panels)....199
1.8.4.2 Configuring an HMI connection via MPI (Basic Panels)....199
1.8.4.3 MPI parameters (Basic Panels)....201
1.8.5 Data exchange (Basic Panels)....207
1.8.5.1 Data exchange using area pointers (Basic Panels)....207
1.8.5.2 Trends (Basic Panels)....221
1.8.5.3 Alarms (Basic Panels)....221
1.8.6 Performance features of communication (Basic Panels)....224
1.8.6.1 S7-300/400 device dependency (Basic Panels)....224
1.8.6.2 Permitted data types for SIMATIC S7 300/400 (Basic Panels)....225
1.8.7 Creating connections in the "Connections" editor (Basic Panels)....225
1.8.7.1 Creating a PROFINET connection (Basic Panels)....225
1.8.7.2 Creating a PROFIBUS connection (Basic Panels)....228
1.8.7.3 Creating an MPI connection (Basic Panels)....229
1.8.7.4 Parameters for the connection (Basic Panels)....231

1.9 Communicating with the SIMATIC S7-1500 Software Controller (Basic Panels)....237
1.9.1 Communication with SIMATIC S7-1500 Software Controller (Basic Panels)....237
1.9.2 Communication via PROFINET (Basic Panels)....238
1.9.2.1 Configuring an HMI connection (Basic Panels)....238
1.9.2.2 PROFINET parameters (Basic Panels)....240
1.9.2.3 Setting port options (Basic Panels)....246
1.9.3 Performance features of communication (Basic Panels)....249
1.9.3.1 WinAC 1500 device dependency (Basic Panels)....249
1.9.4 Configuring time synchronization (Basic Panels)....251
1.9.4.1 Time synchronization (Basic Panels)....251
1.9.4.2 Time synchronization restrictions (Basic Panels)....251
1.9.4.3 Configuring time synchronization for integrated connections....252
1.9.4.4 Configuring time synchronization for non-integrated connections (Basic Panels)....253

1.10 Communicating with SIMATIC ET 200 CPU (Basic Panels)....255
1.10.1 Communication with SIMATIC ET 200 CPU (Basic Panels)....255
1.10.2 Communication via PROFINET (Basic Panels)....255
1.10.2.1 Communication via PROFINET (Basic Panels)....255
1.10.2.2 Configuring an HMI connection via PROFINET (Basic Panels)....256
1.10.2.3 PROFINET parameters (Basic Panels)....258
1.10.2.4 Setting port options (Basic Panels)....265
1.10.3 Communication via PROFIBUS (Basic Panels)....269
1.10.3.1 Communication via PROFIBUS (Basic Panels)....269
1.10.3.2 Configuring an HMI connection via PROFIBUS (Basic Panels)....269
1.10.3.3 PROFIBUS parameters (Basic Panels)....271
1.10.4 Performance features of communication (Basic Panels)....276
1.10.4.1 Device dependencies of SIMATIC ET 200 CPU (Basic Panels)....276
1.10.4.2 Valid data types for SIMATIC ET 200 CPU (Basic Panels)....282
1.10.5 Creating connections in the "Connections" editor (Basic Panels)....282
1.10.5.1 Creating a PROFINET connection (Basic Panels)....282

1.10.5.2 Creating a PROFIBUS DP connection (Basic Panels)....284
1.10.5.3 Connection parameters (Basic Panels)....287

1.11 Communicating with SIMATIC S7 200 (Basic Panels)....290
1.11.1 Communication with SIMATIC S7 200 (Basic Panels)....290
1.11.2 Creating a connection to SIMATIC S7 200 (Basic Panels)....290
1.11.3 Parameters for the connection (Basic Panels)....292
1.11.3.1 Cyclic operation (Basic Panels)....292
1.11.3.2 Parameters for the connection (SIMATIC S7 200) (Basic Panels)....293
1.11.3.3 Ethernet parameters (Basic Panels)....295
1.11.3.4 PROFIBUS parameters (Basic Panels)....296
1.11.3.5 MPI parameters (Basic Panels)....298
1.11.3.6 PPI parameters (Basic Panels)....299
1.11.4 Data exchange (Basic Panels)....300
1.11.4.1 Data exchange using area pointers (Basic Panels)....300
1.11.4.2 Trends (Basic Panels)....314
1.11.4.3 Alarms (Basic Panels)....315
1.11.5 Performance features of communication (Basic Panels)....317
1.11.5.1 Device dependency S7 200 (Basic Panels)....317
1.11.5.2 Permitted data types for SIMATIC S7 200 (Basic Panels)....318

1.12 Communicating with SIMATIC LOGO! (Basic Panels)....319
1.12.1 Communication with SIMATIC LOGO! (Basic Panels)....319
1.12.2 Creating a connection to SIMATIC LOGO! (Basic Panels)....320
1.12.3 Connection parameters (Basic Panels)....321
1.12.3.1 Connection parameters (Basic Panels)....321
1.12.3.2 Ethernet parameters (Basic Panels)....323
1.12.3.3 Cyclic operation (Basic Panels)....324
1.12.4 Data exchange (Basic Panels)....325
1.12.4.1 Trends (Basic Panels)....325
1.12.4.2 Alarms (Basic Panels)....327
1.12.5 Performance features of communication (Basic Panels)....329
1.12.5.1 SIMATIC LOGO! device dependency (Basic Panels)....329
1.12.5.2 Valid data types for SIMATIC LOGO! (Basic Panels)....331

1.13 Communication with other PLCs (Basic Panels)....331
1.13.1 Communication with other PLCs (Basic Panels)....331
1.13.2 Distinctive features when configuring (Basic Panels)....332
1.13.3 Communication drivers (Basic Panels)....333
1.13.3.1 Allen-Bradley (Basic Panels)....333
1.13.3.2 Mitsubishi (Basic Panels)....375
1.13.3.3 Modicon Modbus (Basic Panels)....403
1.13.3.4 Omron (Basic Panels)....432
1.13.3.5 Communication Service Packages (Basic Panels)....449
1.13.4 Data exchange using area pointers (Basic Panels)....451
1.13.4.1 General information on area pointers (Basic Panels)....451
1.13.4.2 Accessing data areas (Basic Panels)....453
1.13.4.3 "Screen number" area pointer (Basic Panels)....453
1.13.4.4 "Date/time" area pointer (Basic Panels)....454
1.13.4.5 "Date/time PLC" area pointer (Basic Panels)....455
1.13.4.6 "Coordination" area pointer (Basic Panels)....456
1.13.4.7 "Project ID" area pointer (Basic Panels)....457
1.13.4.8 "PLC job" area pointer (Basic Panels)....457
1.13.4.9 "Data record" area pointer (Basic Panels)....460

Index..467

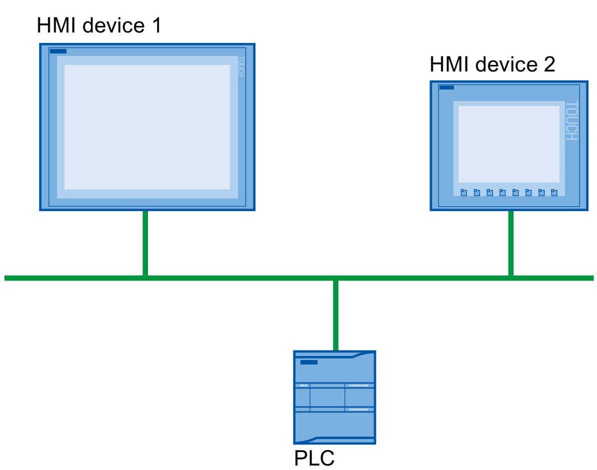
Communicating with PLCs (Basic Panels) 1

1.1 Basics of communication (Basic Panels)

1.1.1 Communication between devices (Basic Panels)

Communication

Data exchange between devices is known as communication. The devices can be interconnected directly or via a network. The interconnected devices in communication are referred to as communication partners.



Data transferred between the communication partners may serve different purposes:

- Display processes
- Operate processes
- Output alarms
- Administer process parameters and machine parameters

Communication partners

Communication between the following devices is described in more detail in this section:

- PLC
  The PLC controls a process by means of a user program.
- HMI device
  You use the HMI device to operate and monitor the process.

Basic information for all communication

The basis for all types of communication is a network configuration. In a network configuration, you specify the connection that exists between the configured devices.

With the network configuration, you also ensure the necessary prerequisites for communication, in other words:

- Every device in a network is assigned a unique address.
- The devices carry out communication with consistent transmission characteristics.

Automation system

The following characteristics describe an automation system:

- The PLC and HMI device are interconnected
- The network between the PLC and HMI device is configured

Communication between HMI devices

The HTTP protocol is available for communication between HMI devices.

For more detailed information, refer to the documentation on the SIMATIC HMI HTTP protocol.

Communication via a uniform and vendor-neutral interface

With OPC (Openess Productivity Collaboration), WinCC has a uniform and manufacturer-neutral software interface. This interface enables standardized data exchange between industrial, office, and manufacturing applications.

For more detailed information, refer to the documentation for OPC.

1.1.2 Devices and networks in the automation system (Basic Panels)

Introduction

To set up an automation system, you must configure, parameterize, and interconnect the individual devices.

You insert PLCs and HMI devices into the project in the same way. Likewise, you configure the two devices in the same way.

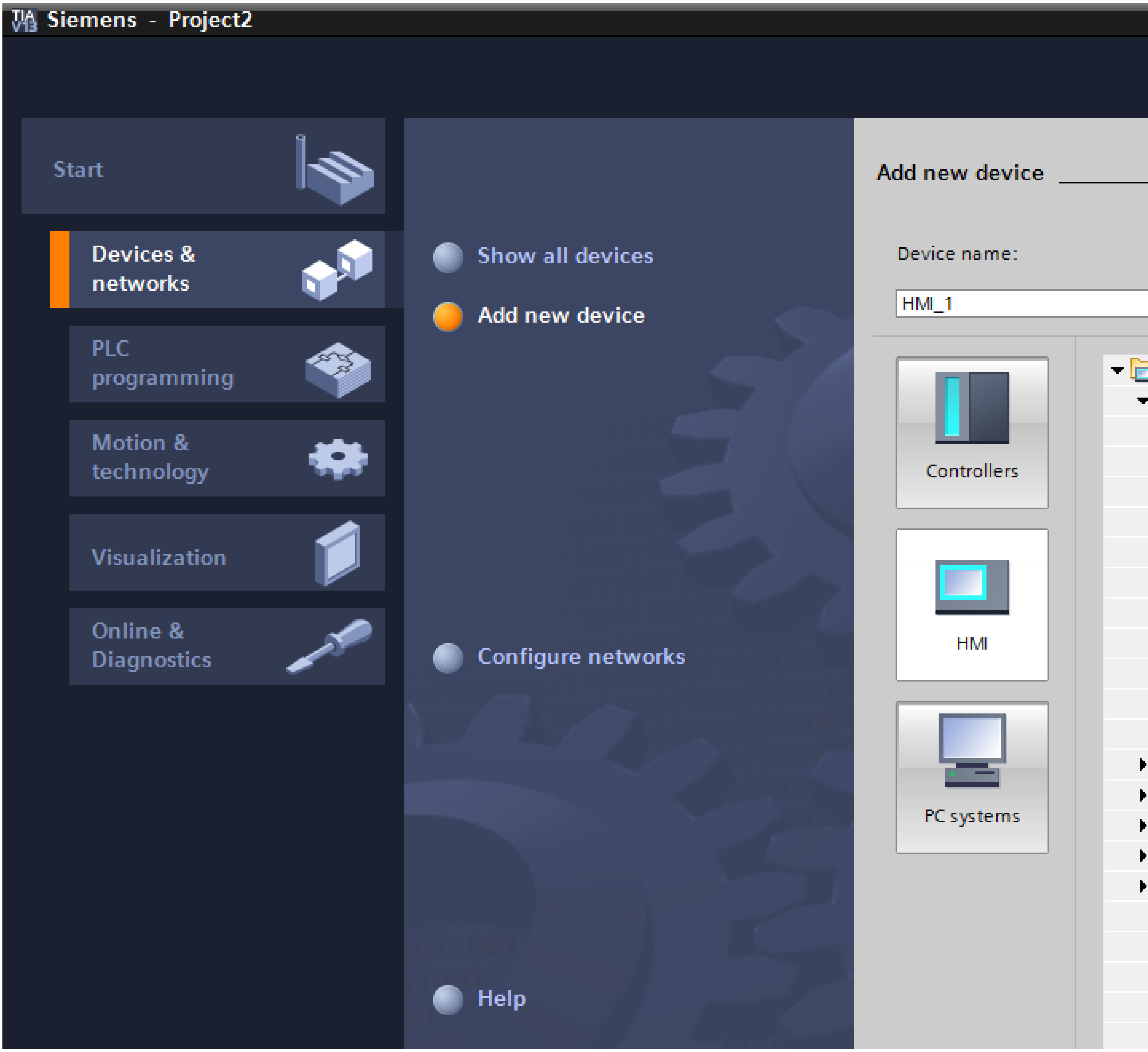
Automation system setup:

1. Insert PLC into the project.
2. Insert HMI device into the project.
3. Network the devices together.
4. Interconnect the devices.

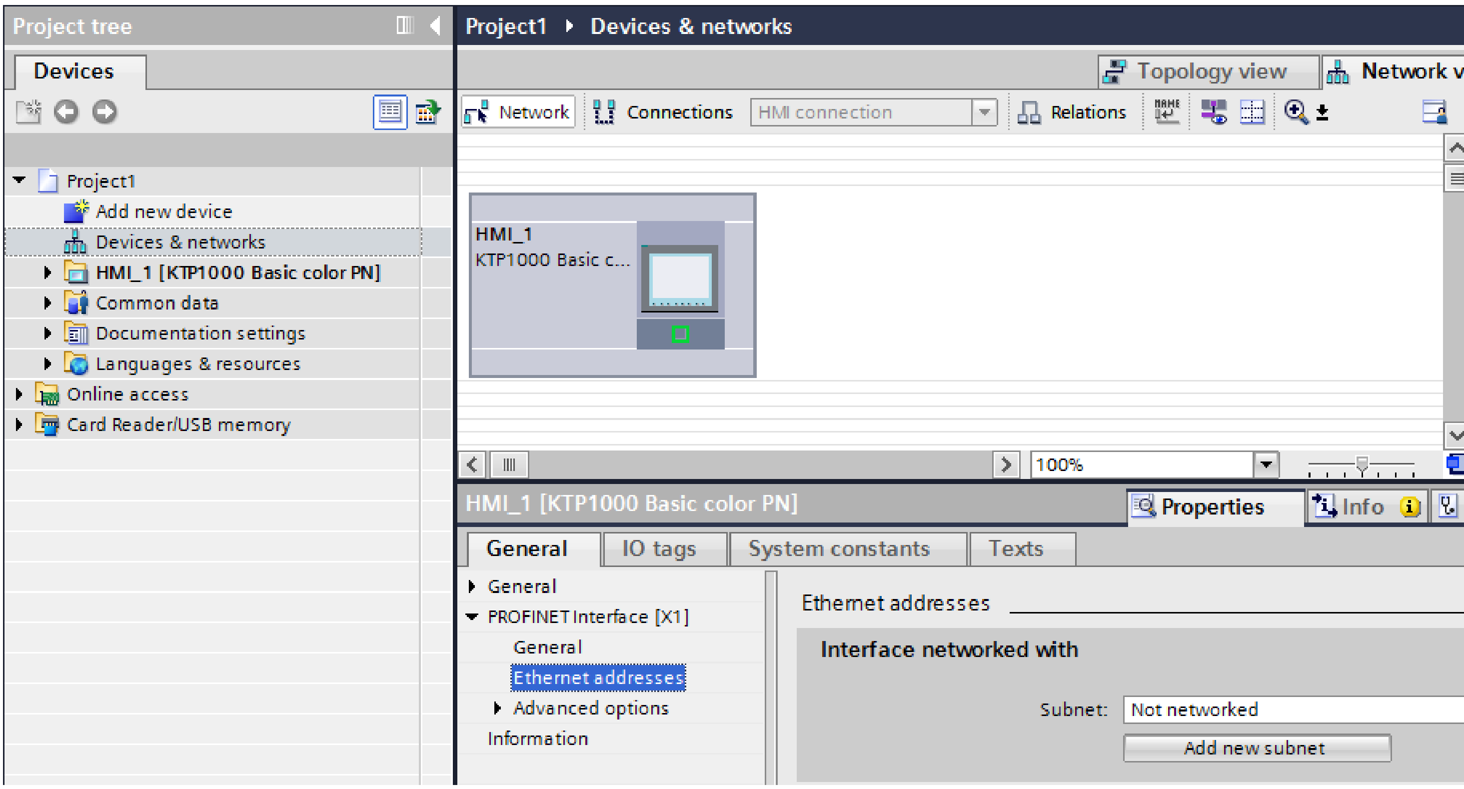
Inserting devices

If you have created a project, you can add a device in the portal view or project view.

- Portal view

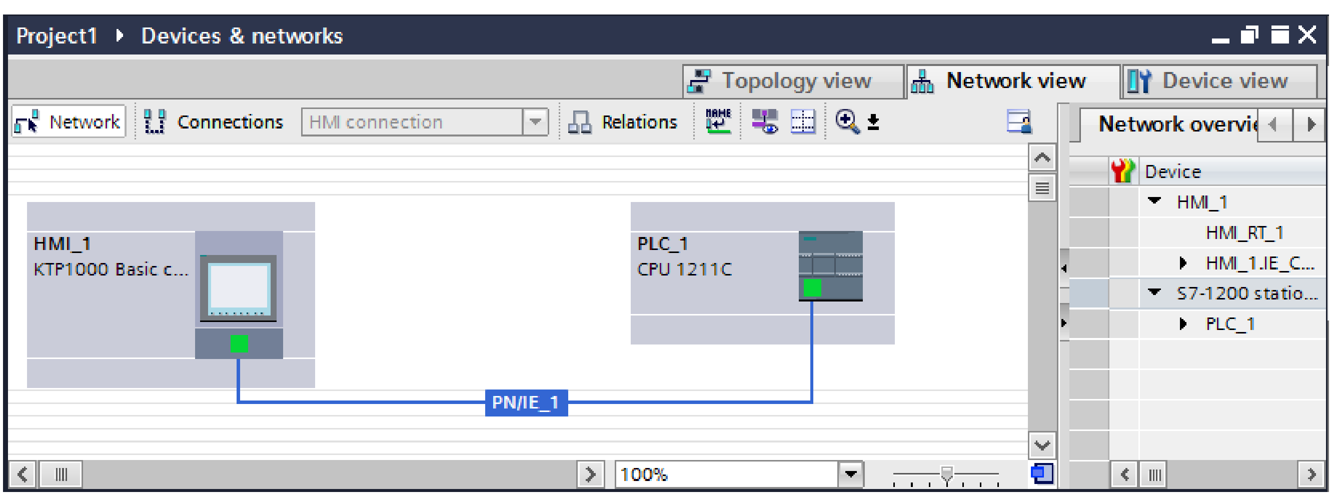


- Project view



Networking devices

You can network the interfaces of the communication-capable devices conveniently in the "Devices & Networks" editor. In the networking step, you configure the physical device connections.

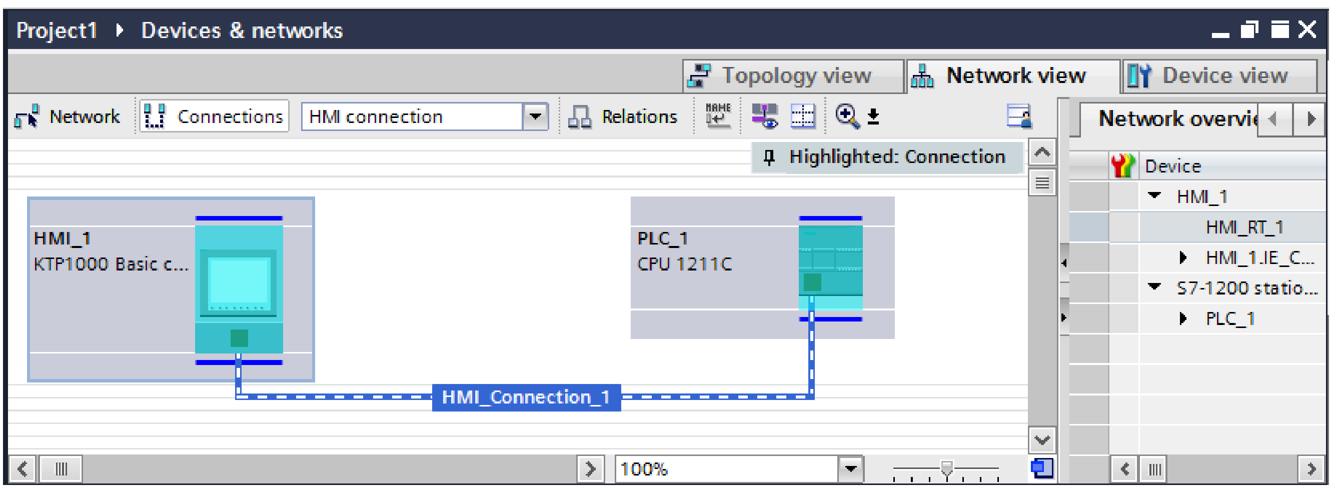


The tabular network overview supplements the graphical network view with the following additional functions:

- You obtain detailed information on the structure and parameter settings of the devices.
- Using the "Subnet" column, you can connect communication-capable components to subnets that have been created.

Connecting devices

After you network the devices together, you configure the connection. You configure the "HMI connection" connection type for communication with the HMI device.



1.1.3 Data exchange using tags (Basic Panels)

Communication using tags

Process values are forwarded in runtime using tags. Process values are data which is stored in the memory of one of the connected automation systems. They represent the status of a plant in the form of temperatures, fill levels or switching states, for example. Define external tags for processing the process values in WinCC.

WinCC works with two types of tag:

- External tags
- Internal tags

Working with tags

See the chapter "Working with tags" for further information about configuring tags.

1.1.4 Data exchange using area pointers (Basic Panels)

Communication using area pointers

Area pointers are parameter fields. WinCC receives information from these parameter fields in Runtime during the course of the project. This information contains data on the location and size of data areas in the PLC.

During communication, the PLC and the HMI device alternately access these data areas for read and write operations.

Based on the evaluation of data stored in the data areas, the PLC and HMI device initiate mutually defined actions.

The area pointers are managed centrally in the "Connections" editor. Area pointers are used to exchange data from specific user data areas.

You use the following area pointers in WinCC:

- Data record
- Date/time
- Coordination
- Job mailbox
- Date/time PLC
- Project ID
- Screen number

The availability of the various area pointers is determined by the HMI device used.

1.1.5 Communication drivers (Basic Panels)

Communication drivers

A communication driver is a software component that establishes a connection between a PLC and an HMI device. The communication driver thus enables the assignment of process values to HMI tags.

The interface as well as the profile and transmission speed can be chosen, depending on the HMI device used and the connected communication partner.

1.2 Editors for communication (Basic Panels)

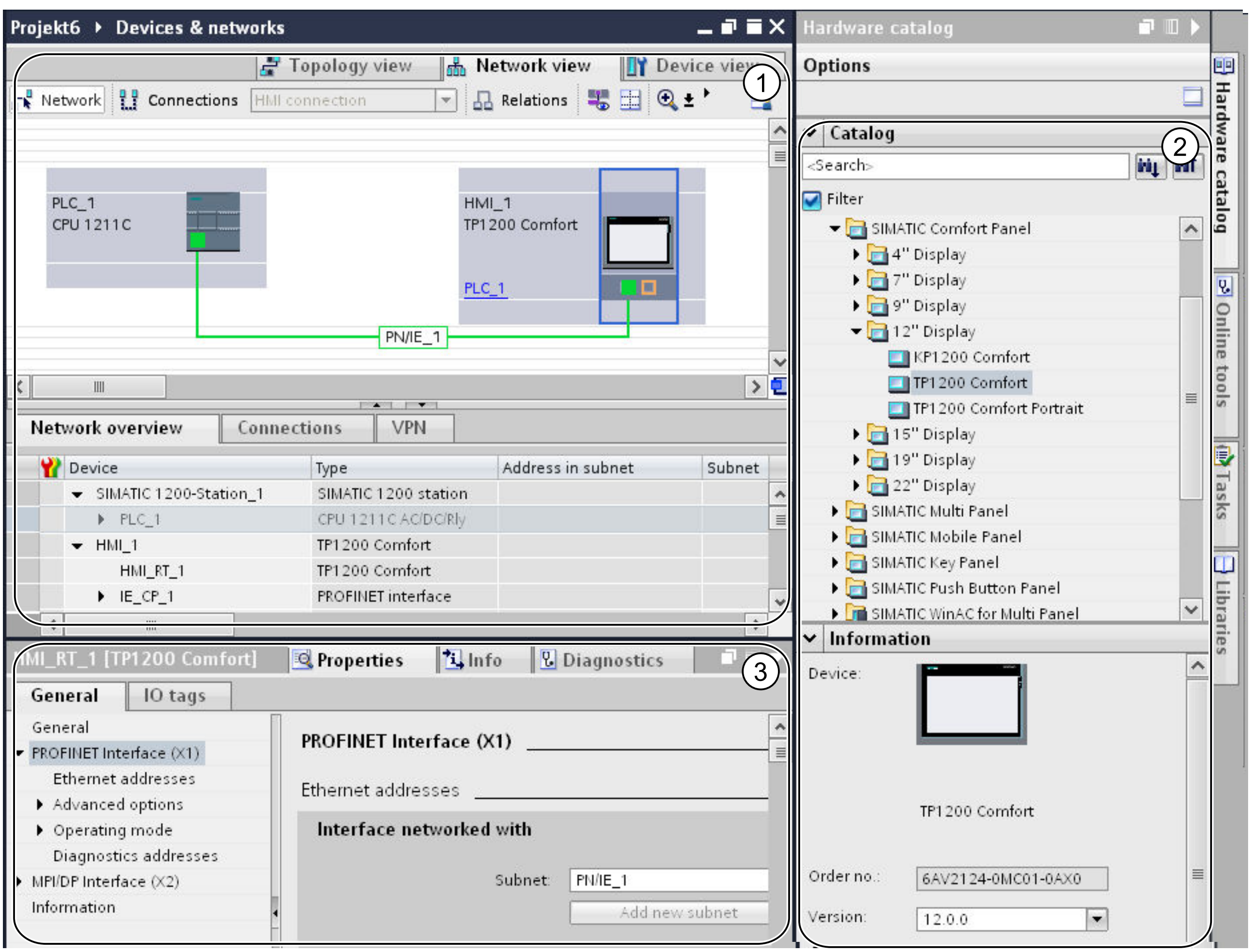
1.2.1 "Devices & networks" editor (Basic Panels)

Function of the hardware and network editor

The "Devices & networks" editor is the development environment for networking, configuring and assigning parameters to devices and modules.

Configuration

The "Devices & networks" editor consists of the following components:



1	Device view, network view, topology view
2	Hardware catalog
3	Inspector window

The "Devices & networks" editor provides you with three different views of your project. You can switch between these three views at any time depending on whether you want to produce and edit individual devices and modules, entire networks and device configurations or the topological structure of your project.

The inspector window contains information on the object currently marked. Here you can change the settings for the object marked.

Drag the devices and modules you need for your automation system from the hardware catalog to the network, device or topology view.

1.2.2 Network view (Basic Panels)

Introduction

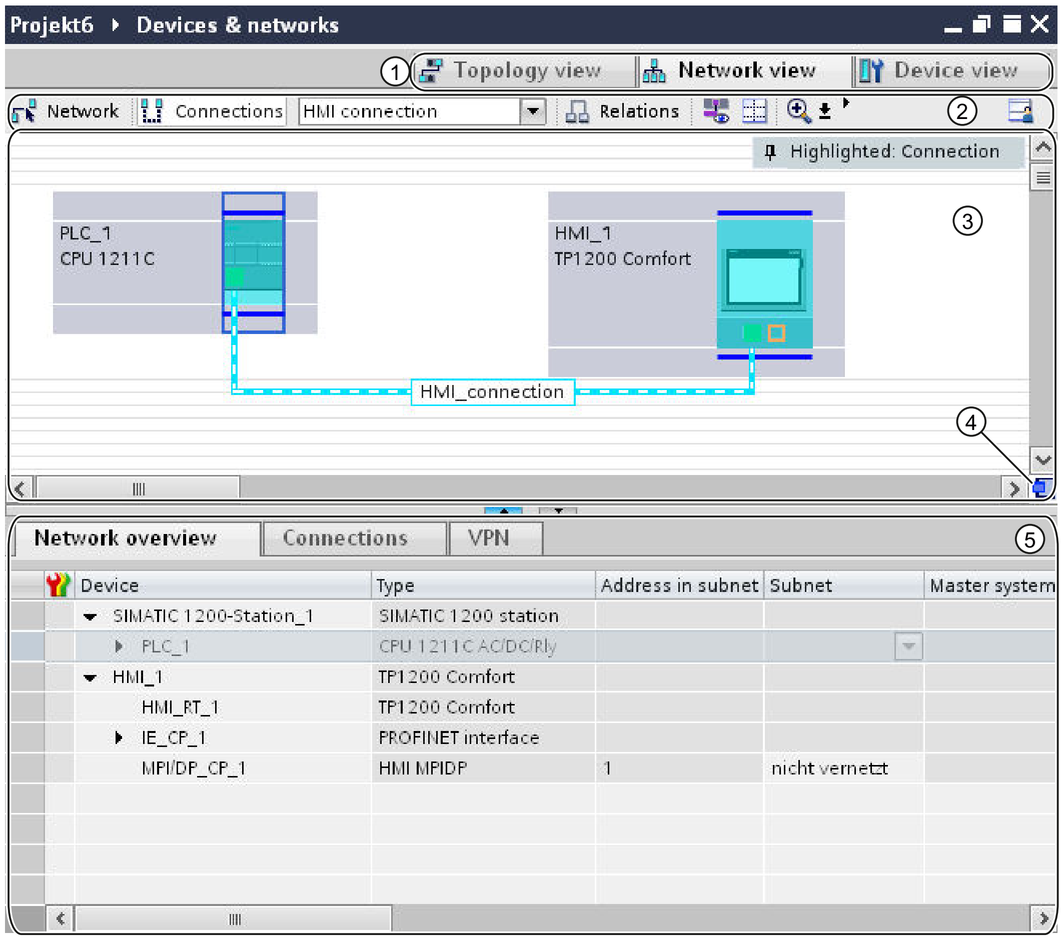
The network view is a working area of the hardware and network editor.

You undertake the following tasks in network view:

- Configuring and assign device parameters
- Networking devices with one another

Structure

The following diagram shows the components of the network view:



1	Changeover switch: network view/device view/topology view
2	Toolbar of network view
3	Graphic area of network view
4	Overview navigation
5	Table area of network view

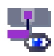
You can use your mouse to change the spacing between the graphic and table areas of the network view.

To do this, click on the upper edge of the table view and expand or contract this by moving the mouse with the mouse button held down.

You can use the two small arrow keys to minimize, maximize or select the latest table structure of the table view with just one click.

Toolbar

The toolbar provides the following functions:

Icon	Meaning
	Mode to network devices.
	Mode to create connections. You can select the connection type using the adjacent drop-down list.
	Mode to create relations.
	Show interface addresses.
	Adjust the zoom setting. You can select the zoom setting or enter it directly in the adjacent drop-down list. You can also zoom in or zoom out the view in steps using the zoom symbol or draw a frame around an area to be zoomed in.
	Show page breaks Enables the page break preview. Dashed lines will be displayed at the positions where the pages will break when printed.
	Remember layout Saves the current table view. The layout, width and visibility of columns in the table view is stored.

Graphic area

The graphic area of the network view displays any network-related devices, networks, connections and relations. In this area, you add devices from the hardware catalog, connect them with each other via their interfaces and configure the communication settings.

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

The table area of the network view includes various tables for the devices, connections and communication settings present:

- Network overview
- Connections
- I/O communication

You can use the shortcut menu of the title bar of the table to adjust the tabular display.

1.2.3 Network data (Basic Panels)

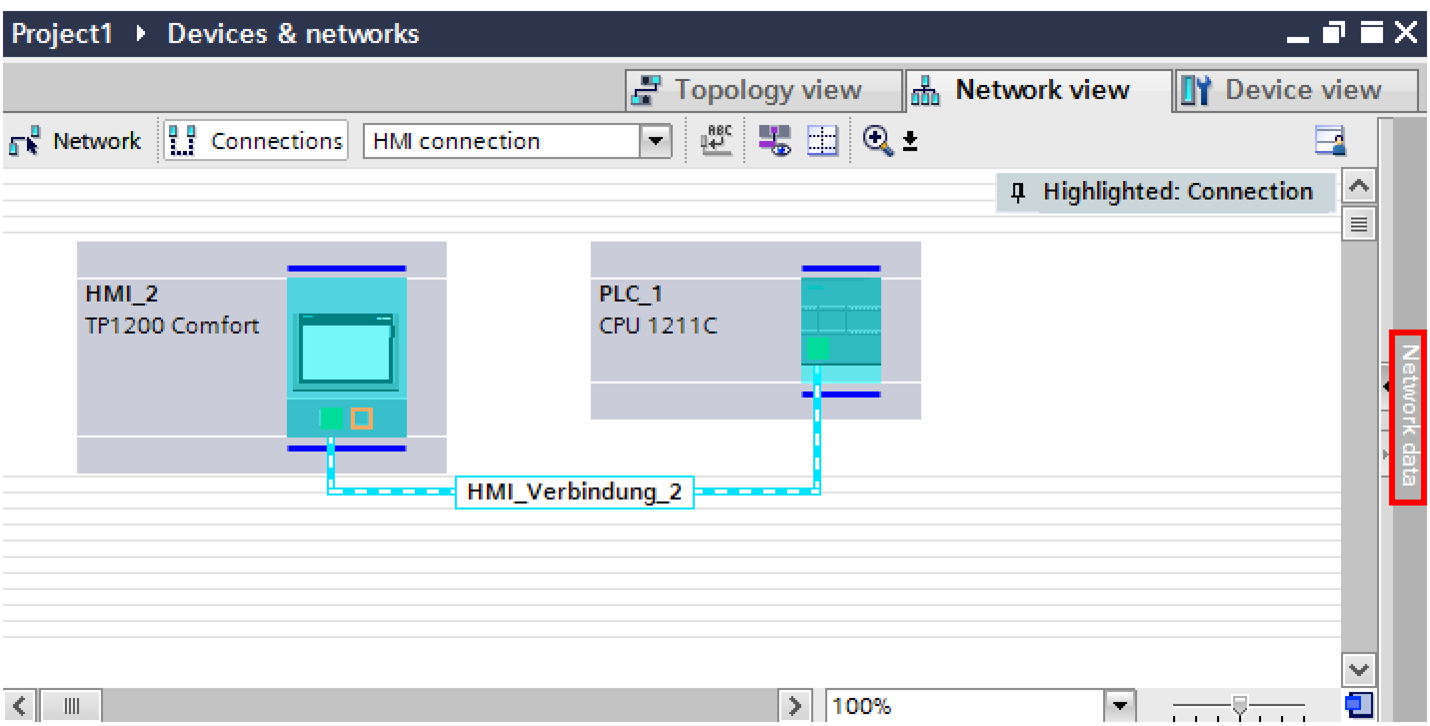
Introduction

The "Network view" editor also gives you a tabular view of the "Network data" in addition to the graphic network view.

You have the following selections in the "Network data" editor:

- Network overview
- Connections
- VPN
- I/O communication

You open the "Network data" below the graphic network overview.



Basic functions

The network data are displayed in tabular form and support the following basic functions for editing a table:

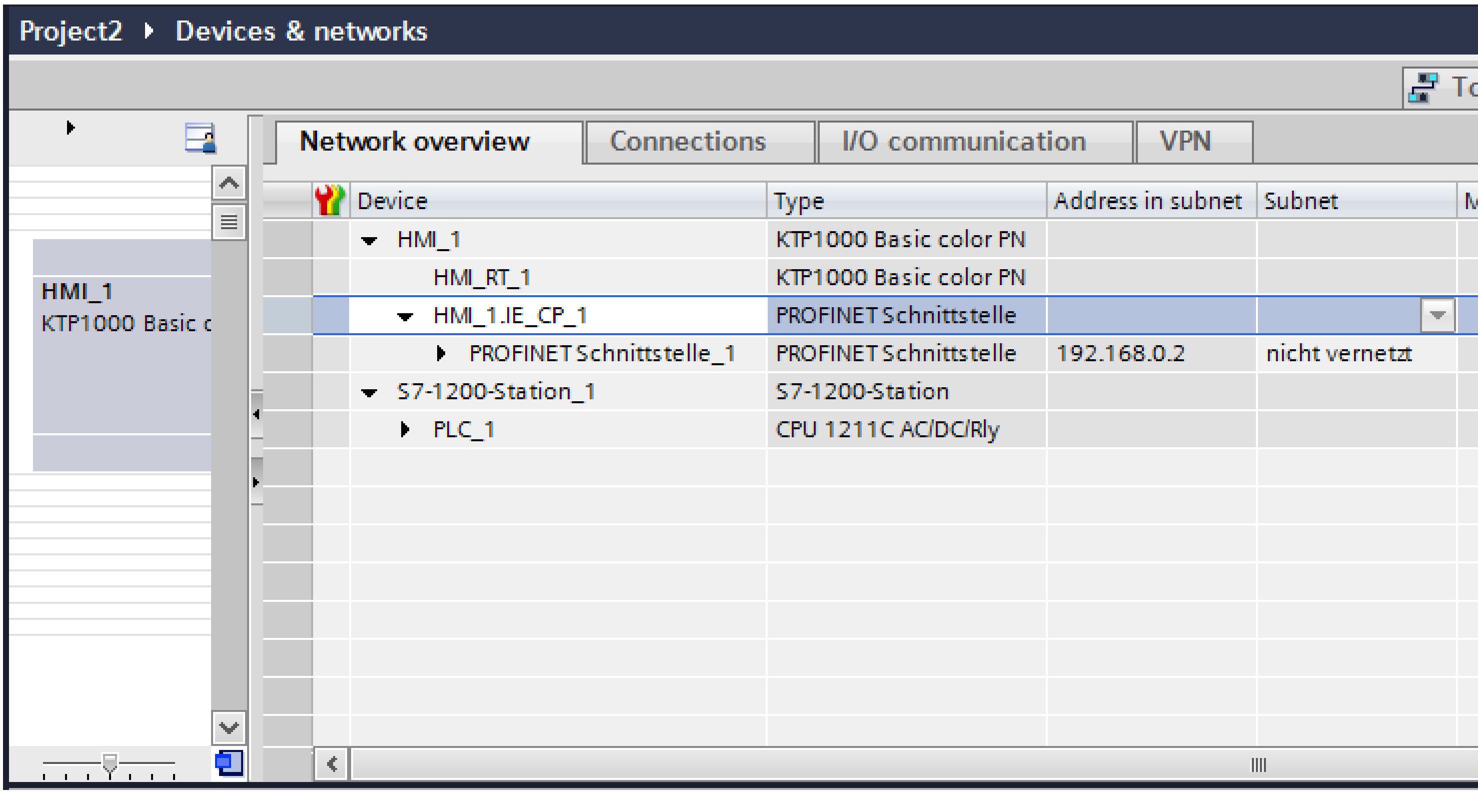
- Displaying and hiding table columns
  Note: The columns of relevance to configuration cannot be hidden.
- Optimizing column width
- Sorting table
- Displaying the meaning of a column, a row or cell using tooltips.

Network overview

The tabular network overview adds the following functions to the graphic network view:

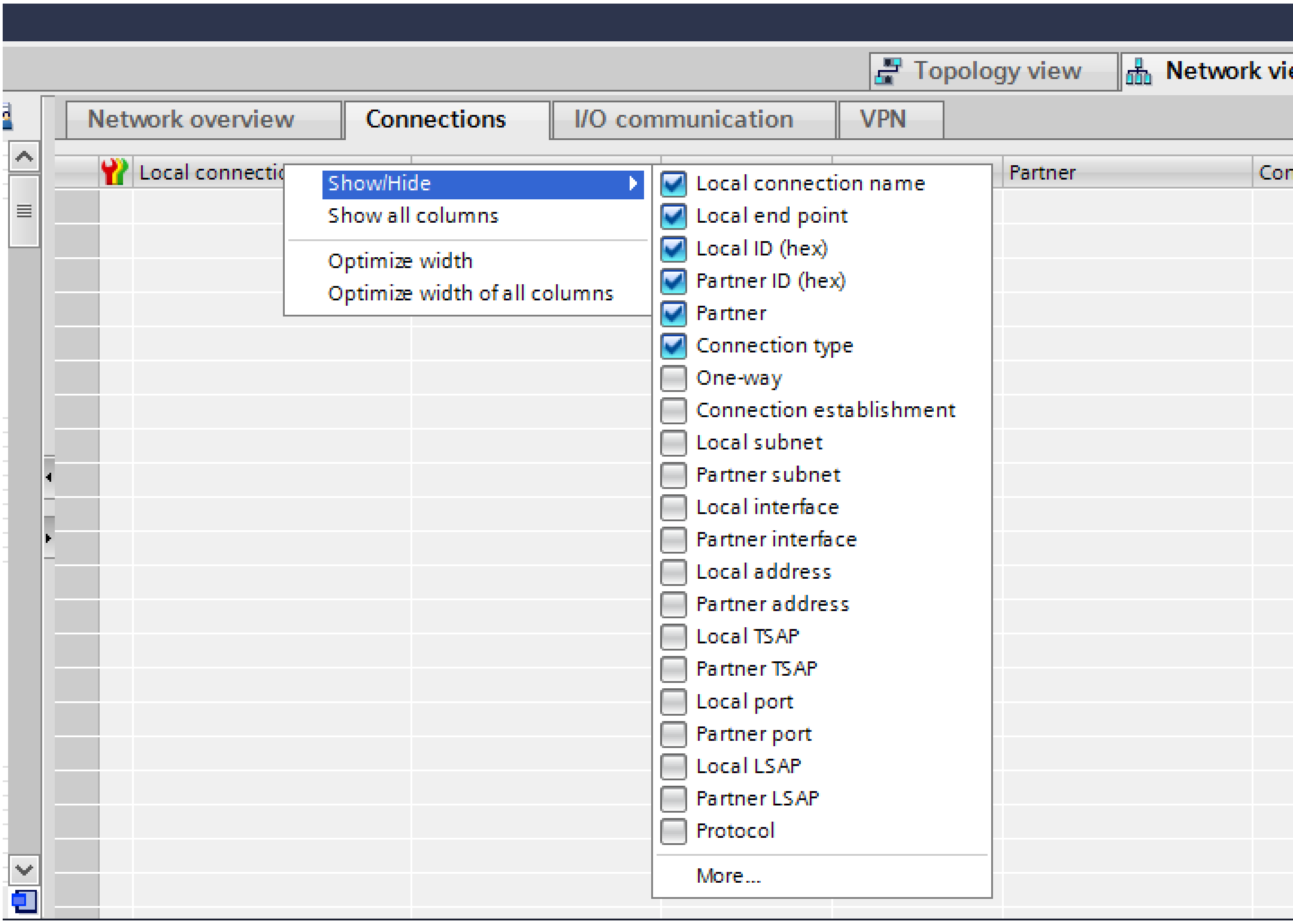
- You obtain detailed information on the structure and parameter settings of the devices.
- Using the "Subnet" column, you can connect components capable of communication with created subnets.

The



Connections

You will find additional network data under "Connections".



1.2.4 Diagnostics of online connections (Basic Panels)

Diagnostics of online connections

You can read out the diagnostics data of existing connections in the TIA Portal.

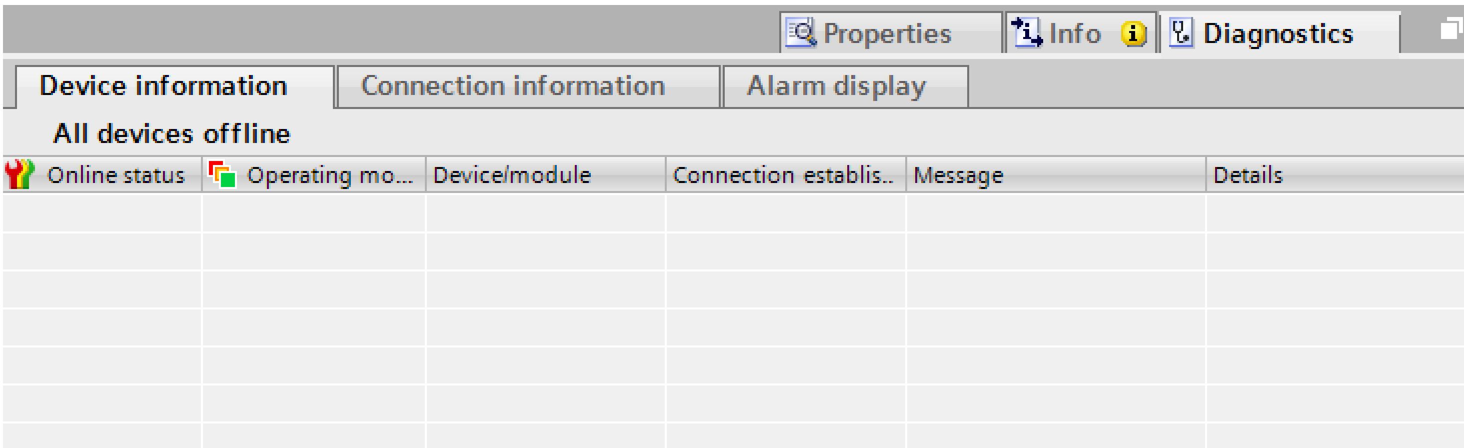
The "Diagnostics" function shows the connection data in tabular form in the Inspector window.

Requirements

- Devices must be in "Online" mode.

Device information

The diagnostics data of all devices in "Online" mode is displayed in the "Diagnostics > Device information" Inspector window.



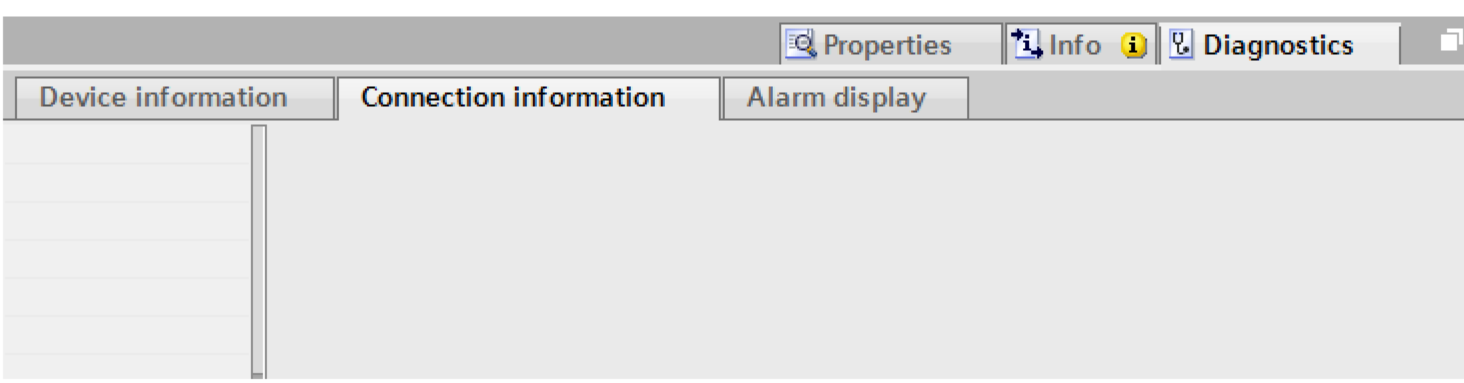
The following data is displayed:

- Online status
- Operating mode
- Device / module
- Alarm
- Details
- Help

Connection information

Use the "Connection information" function to display the diagnostics data of the connection selected in the "Devices&Networks" editor.

A graphic displays the communication partners of the connection and by which communication driver they are connected with each other.



The following data is displayed:

- End point
- Interface
- Subnet
- Address
- TSAP
- Number of HMI resources

1.2.5 Device view (Basic Panels)

Introduction

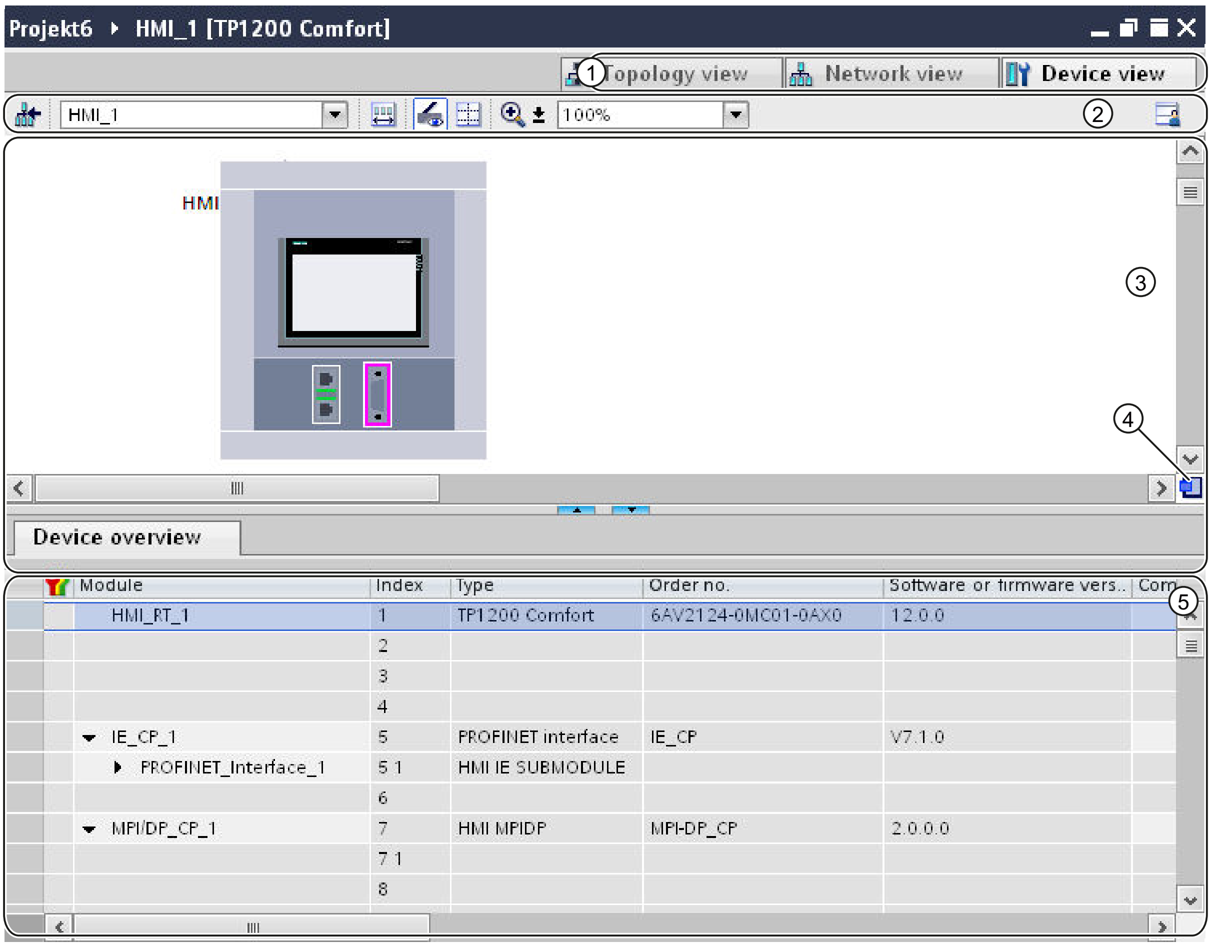
The device view is a working area of the hardware and network editor.

You undertake the following tasks in the device view:

- Configuring and assign device parameters
- Configuring and assign module parameters

Structure

The following diagram shows the components of the device view:



1 Changeover switch: network view/device view/topology view
2 Toolbar of device view
3 Graphic area of the device view
4 Overview navigation
5 Table area of device view

You can use your mouse to change the spacing between the graphic and table areas of the device view.

To do this, click on the upper edge of the table view and expand or contract this by moving the mouse with the mouse button held down. You can use the two small arrow keys to minimize, maximize or select the latest table structure of the table view with just one click.

Toolbar

The toolbar provides the following functions:

Icon	Meaning
	Switches to the network view. Note: The device view can switch between the existing devices using the drop-down list.
	Show the area of unplugged modules.
	Show module labels.
	Adjust the zoom setting. Select the zoom setting or enter it directly in the adjacent drop-down list. You can use the Zoom button to zoom in or out incrementally or to drag a frame around an area to be enlarged. You can read the address descriptions of the I/O channels of signal modules at a zoom level setting of 200% or higher.
	Show page breaks Enables the page break preview. Dashed lines will be displayed at the positions where the pages will break when printed.
	Remember layout Saves the current table view. The layout, width and visibility of columns in the table view is stored.

Graphic area

The graphic area of the device view displays hardware components and if necessary the associated modules that are assigned to each other via one or more racks. In the case of devices with racks, you have the option of installing additional hardware objects from the hardware catalog into the slots on the racks.

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

The table area of the device view gives you an overview of the modules used and the most important technical and organizational data.

You can use the shortcut menu of the title bar of the table to adjust the tabular display.

1.2.6 Topology view (Basic Panels)

Introduction

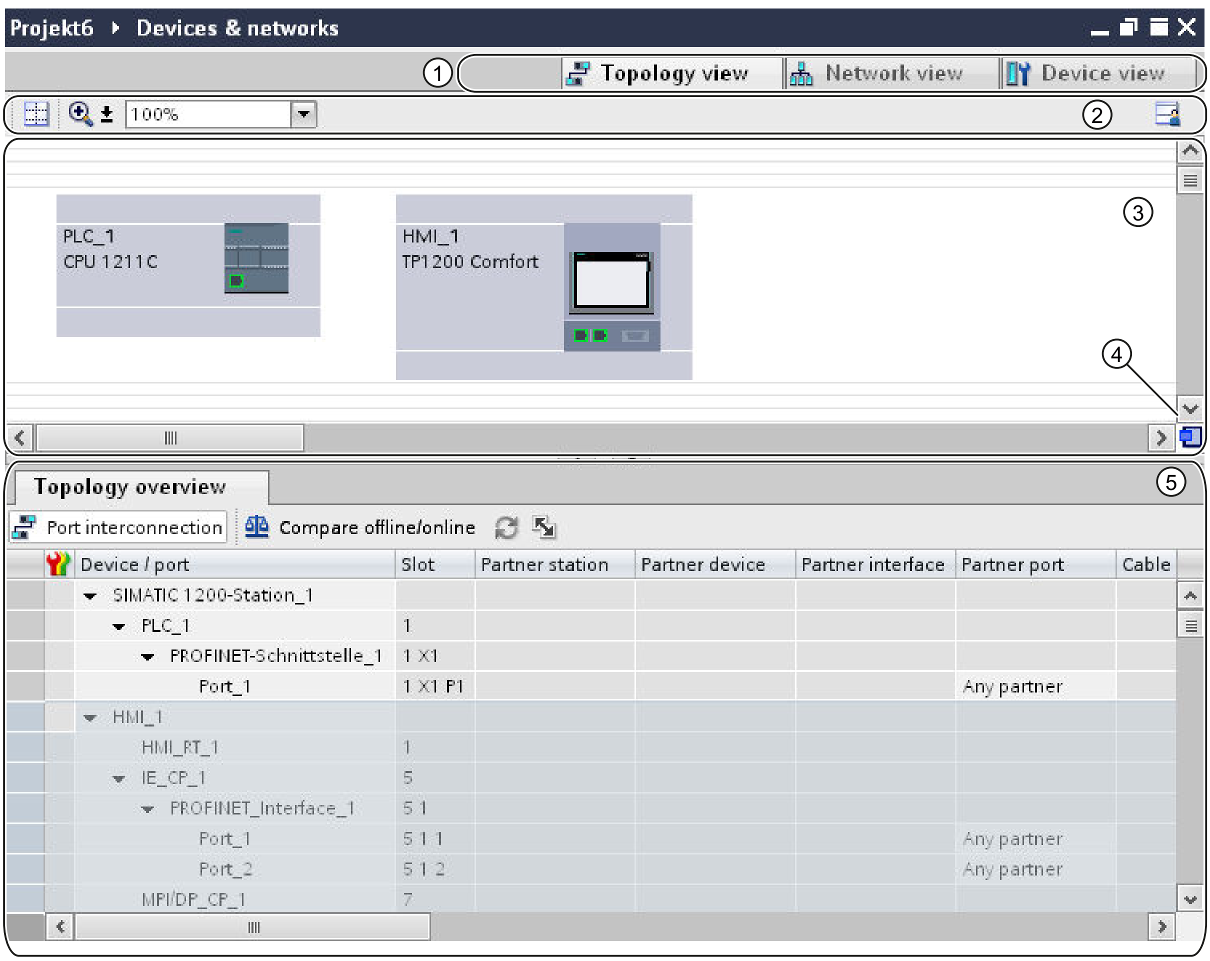
The topology view is a working area of the hardware and network editor.

You undertake the following tasks in topology view:

- Displaying the Ethernet topology
- Configuring the Ethernet topology
- Identify and minimize differences between the desired and actual topology

Structure

The following figure provides an overview of the topology view.



1	Changeover switch: device view / network view / topology view
2	Topology view toolbar
3	Graphic area of the topology view
4	Overview navigation
5	Table area of the topology view

You can use your mouse to change the spacing between the graphic and table areas of the topology view.

To do this, click on the upper edge of the table view and expand or contract this by moving the mouse with the mouse button held down. You can use the two small arrow keys to minimize, maximize or select the latest table structure of the table view with just one click.

Toolbar

The toolbar provides the following functions:

Icon	Meaning
	Adjust the zoom setting. You can select the zoom setting via the adjacent drop-down list or enter it directly. You can also zoom in or zoom out the view in steps using the zoom symbol or draw a frame around an area to be zoomed in.
	Show page breaks Enables the page break preview. Dashed lines will be displayed at the positions where the pages will break when printed.
	Remember layout Saves the current table view. The layout, width and visibility of columns in the table view is stored.

Graphic area

The graphic area of the topology view displays Ethernet modules with their appropriate ports and port connections. Here you can add additional hardware objects with Ethernet interfaces.

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

This displays the Ethernet or PROFINET modules with their appropriate ports and port connections in a table. This table corresponds to the network overview table in the network view.

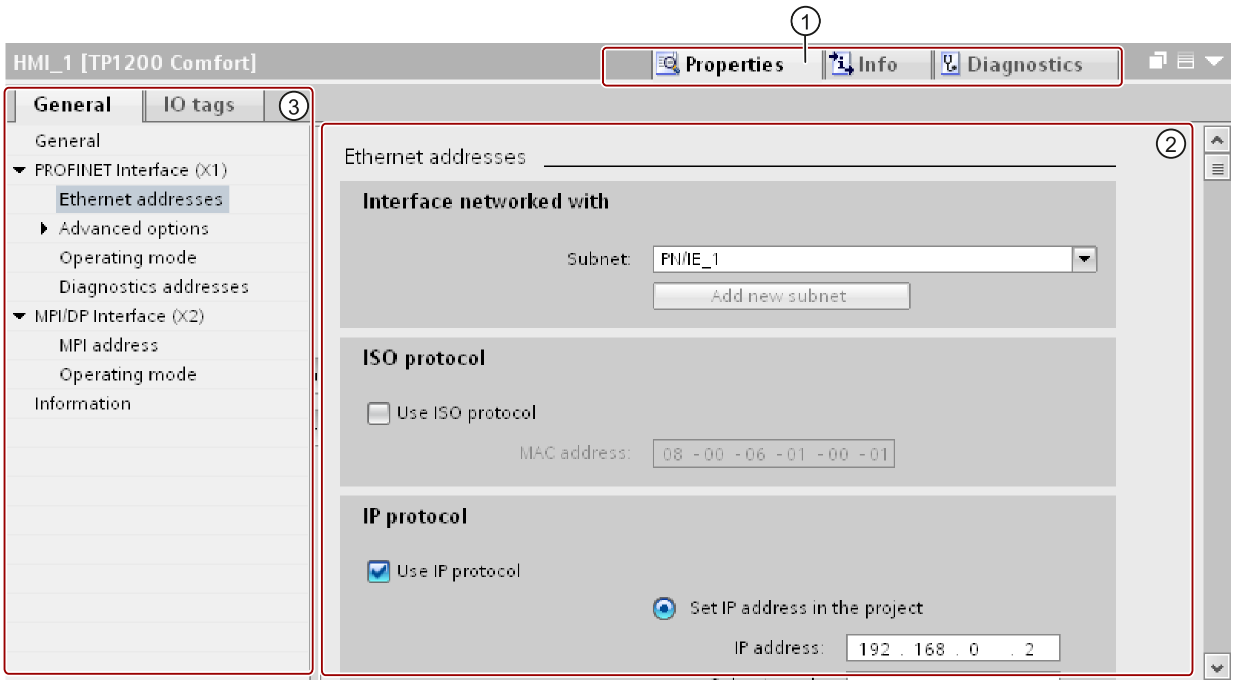
You can use the shortcut menu of the title bar of the table to adjust the tabular display.

1.2.7 Inspector window (Basic Panels)

The properties and parameters shown for the object selected can be edited in the inspector window.

Structure

The inspector window consists of the following components:



1	Switch between various information and work areas
2	Navigation between various pieces of information and parameters
3	Display showing the selected information and parameters

Function

The information and parameters in the inspector window are split into different types of information:

- Properties
- Info
- Diagnostics

To display the corresponding information and parameters, click in the area you want. The "Properties" area is the most important one for configuring an automation system. This area is displayed by default.

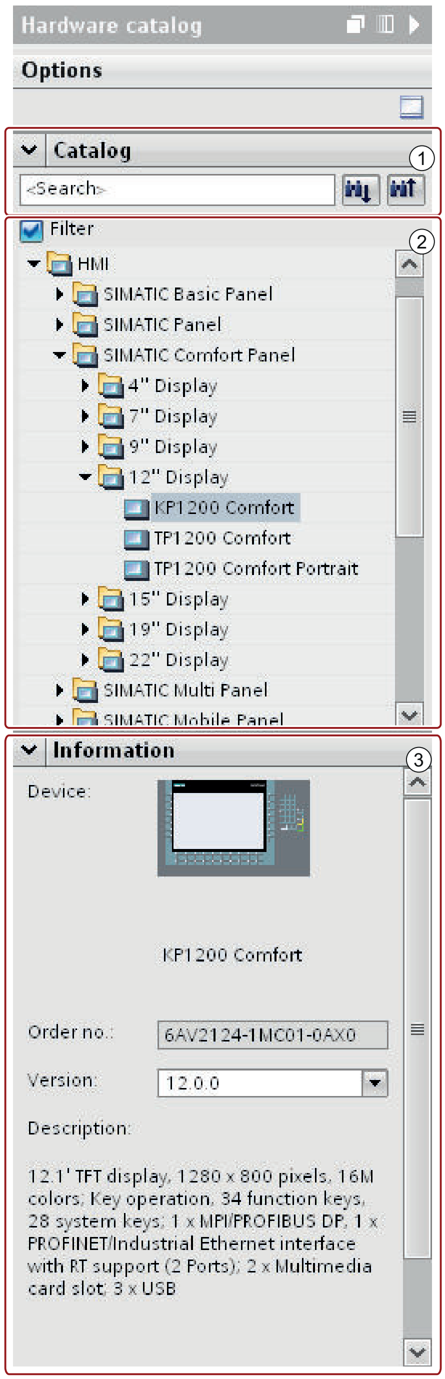
The left pane of the inspector window is used for area navigation. Information and parameters are arranged there in groups. If you click on the arrow symbol to the left of the group name, you can expand the group if sub-groups are available. If you select a group or sub-group, the corresponding information and parameters are displayed in the right pane of the inspector window and can be edited there too.

1.2.8 Hardware catalog (Basic Panels)

The "Hardware catalog" task card gives you easy access to a wide range of hardware components.

Structure

The "Hardware catalog" task card consists of the following panes:



1	"Catalog" pane, search and filter function
2	"Catalog" pane, component selection
3	"Information" pane

Search and filter function

The search and filter functions of the "Catalog" pane make it easy to search for particular hardware components. You can limit the display of the hardware components to certain criteria using the filter function. For example, you can limit the display to objects that you can also place within the current context or which contain certain functions.

Objects that can be used in the current context include, for example, interconnectable objects in the network view or only modules compatible with the device in the device view.

Component selection

The component selection in the "Catalog" pane contains the installed hardware components in a tree structure. You can move the devices or modules you want from the catalog to the graphic work area of the device or network view.

Installed hardware components without a license are grayed out. You cannot use non-licensed hardware components.

Hardware components belonging to various components groups thematically are partially implemented as linked objects. When you click on such linked hardware components, a catalog tree opens in which you can find the appropriate hardware components.

Information

The "Information" pane contains detailed information on the object selected from the catalog:

- Schematic representation
- Name
- Order number
- Version number
- Description

1.2.9 Information on hardware components (Basic Panels)

In the hardware catalog, you can display information on selected hardware components in the "Information" pane. You can also display further information on the selected hardware components using the shortcut menu.

Access to further information

If you select a hardware object in the hardware catalog and open the shortcut menu, you not only have the "Copy" function available but also three options for accessing information on Service & Support:

- Information regarding product support
- FAQs
- Manuals

The required information is displayed in the work area of the hardware and network editor.

Note

You can only access Service & Support when you are connected to the Internet and the function is enabled. By default, the function is disabled.

To enable the function, refer to the instructions in the section "Enabling product support".

Information regarding product support

Here, you have access to general information on hardware and software components. The order number of the selected hardware object is entered automatically in the search mask. You can, however, also search for other hardware and software components.

FAQs

Here, you have access to "Frequently Asked Questions" (FAQs). You can view various entries on hardware and software questions. Using a detailed search mask, you can filter the required topics.

Manuals

Here, you have access to the manuals of the various hardware components. This is particularly useful if the configuration, addressing or parameter assignment you are planning requires more detailed knowledge of the hardware you are using.

1.3 Networks and connections (Basic Panels)

1.3.1 SIMATIC communication networks (Basic Panels)

1.3.1.1 Communication networks (Basic Panels)

Overview

Communication networks are a central component of modern automation solutions. Industrial networks have to fulfill special requirements, for example:

- Coupling of automation systems as well as simple sensors, actuators, and computers.
- The information has to be correct and has to be transferred at the right moment.
- Robust against electromagnetic disturbances, mechanical stresses and soiling
- Flexible adaptation to the production requirements

Industrial networks belong to the LANs (Local Area Networks) and allow communication within a limited area.

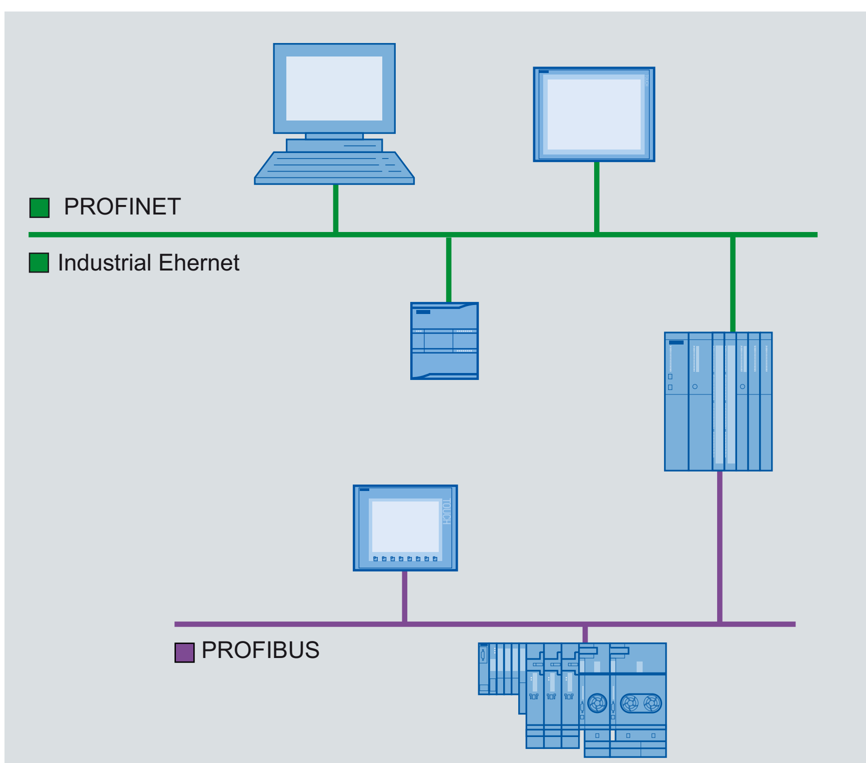
Industrial networks fulfill the following communication functions:

- Process and field communication of the automation systems including sensors and actuators
- Data communication between automation systems
- IT communication for integrating the modern information technology

Overview of the networks

This section examines the following networks:

- **Industrial Ethernet**
  The industrial network standard for all levels
- **PROFINET**
  The open Industrial Ethernet standard for automation
- **PROFIBUS**
  The international standard for the field area and market leader at the field busses
- **MPI**
  The integrated interface of the SIMATIC products
- **PPI**
  The integrated interface specially for the S7-200



1.3.1.2 PROFINET and Ethernet (Basic Panels)

Industrial Ethernet

Industrial Ethernet, which is based on IEEE 802.3, enables you to connect your automation system to your office networks. Industrial Ethernet provides IT services that you can use to access production data from the office environment.

Ethernet network

An Ethernet network allows you to interconnect all devices that are connected to the network via an integrated Ethernet interface or a communication module. This enables connection of multiple HMI devices to one SIMATIC S7 PLC and multiple SIMATIC S7 PLCs to one HMI device. The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used. Additional information is available in the documentation for the respective HMI device.

PROFINET

PROFINET is an open standard for industrial automation defined by IEEE 61158 and based on Industrial Ethernet. PROFINET makes use of IT standards all the way to the field level and enables plant-wide engineering.

With PROFINET, you can realize high-performance automation solutions for applications with stringent real-time requirements.

1.3.1.3 PROFIBUS (Basic Panels)

PROFIBUS DP

PROFIBUS DP (distributed I/O) is used to connect the following devices:

- PLCs, PCs, HMI devices
- Distributed I/O devices, e.g., SIMATIC ET 200
- Valves
- Drives

PROFIBUS DP's fast response times make it ideally suited for the manufacturing industry.

Its basic functionality includes cyclic exchange of process data between the master and PROFIBUS DP slaves, as well as diagnostics.

PROFIBUS network

You can connect an HMI device within the PROFIBUS network to any SIMATIC S7 module that has an integrated PROFIBUS or PROFIBUS DP interface. You can thereby connect multiple HMI devices to one SIMATIC S7 PLC and multiple SIMATIC S7 PLCs to one HMI device.

The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used. Additional information is available in the documentation for the respective HMI device.

You configure the SIMATIC S7-200 PLC as a passive device in the network. You connect the SIMATIC S7-200 using the DP connector or a PROFIBUS communication module.

1.3.1.4 MPI (Basic Panels)

MPI

MPI (Multi-Point Interface) is the integrated interface for SIMATIC products:

- PLCs
- HMI devices
- Programming device/PC

Small subnets with the following characteristics are set up with MPI:

- Short distances
- Few devices
- Small data quantities

MPI network

You connect the HMI device to the MPI interface of the SIMATIC S7 PLC. This enables connection of multiple HMI devices to one SIMATIC S7 PLC and multiple SIMATIC S7 PLCs to one HMI device. The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used. Additional information is available in the documentation for the respective HMI device.

Network architectures

MPI is based on the PROFIBUS standard (IEC 61158 and EN 50170) and supports the following bus topologies:

- Line
- Star
- Tree

An MPI subnet contains a maximum of 127 devices and consists of multiple segments. Each segment contains a maximum of 32 devices and is limited by terminating resistors. Repeaters are used to connect segments. The maximum cable length without a repeater is 50 m.

1.3.1.5 PPI (Basic Panels)

Introduction

PPI (point-to-point interface) is an integrated interface that was developed specially for the SIMATIC S7-200. A PPI network typically connects S7-200 PLCs. However, other SIMATIC PLCs (e.g., S7-300 and S7-400) or HMI devices can communicate with a SIMATIC S7-200 in the PPI network.

PPI network

A PPI connection is a point-to-point connection. The HMI device is the master. The SIMATIC S7-200 is the slave.

You can connect a maximum of one SIMATIC S7-200 to an HMI device. You use the serial connector of the CPU to connect the HMI device. You can connect multiple HMI devices to one SIMATIC S7-200. From the perspective of the SIMATIC S7-200, only one connection at a time is possible.

Note

The PPI network can contain a maximum of four masters in addition to the HMI device. For performance reasons, do not configure more than four devices at a time as a master in the PPI network.

Network architectures

PPI is based on the PROFIBUS standard (IEC 61158 and EN 50170) and supports the following bus topologies:

- Line
- Star

Multi-master networks with a maximum of 32 masters are set up with PPI:

- An unlimited number of masters can communicate with each slave.
- A slave can be assigned to multiple masters.

The RS 485 repeater can be used to extend the PPI network. Modems can also be connected to the PPI network.

1.3.2 Configuring networks and connections (Basic Panels)

1.3.2.1 Networking devices (Basic Panels)

Introduction

The "Devices & Networks" editor is provided for configuring connections. You can network devices in the editor. You can also configure and assign parameters to devices and interfaces. You then configure the required connections between the networked devices.

In the "Devices & Networks" editor you configure HMI connections with the PLCs:

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

You configure the HMI connections to other PLCs in the "Connections" editor of the respective HMI device.

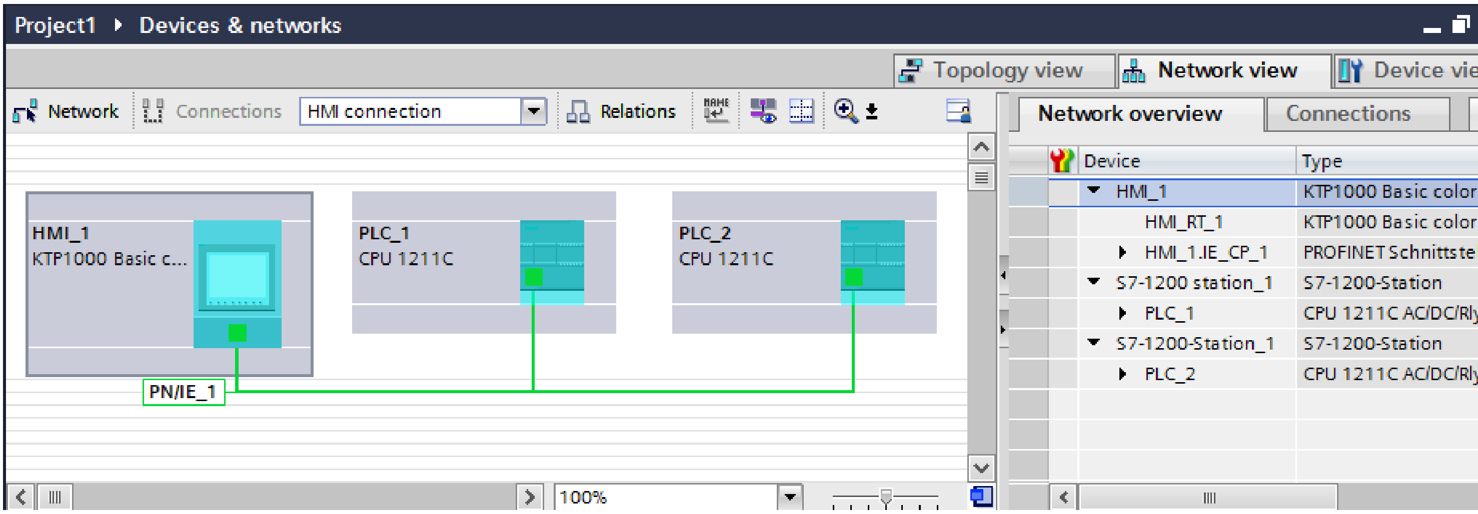
Networking devices

The network view of the "Devices & Networks" editor includes a graphical area and a tabular area. You can use the graphical area to network the devices in the project with drag-and-drop. The tabular area provides an overview of the devices and their components.

You can network the following PLCs together with HMI devices in the "Devices & Networks" area.

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

All other PLCs are available in the TIA Portal and are configured "not integrated". You configure "not integrated" connections in the "Connections" editor of the HMI device.



With the networking step, you configure the physical connection of the communication partners. The networking of devices is depicted by lines that are colored according to the interface.

1.3.2.2 Configuring an integrated connection in the "Devices & Networks" editor (Basic Panels)

Introduction

You configure an HMI connection between the HMI device and a SIMATIC S7 PLC in the "Devices & Networks" editor. This HMI connection is the direct connection between the communication partners that you have created in a project.

Integrated connections

Connections of devices within a project are referred to as integrated connections. In the case of integrated connections, you can directly configure addresses of PLC tags.

Note

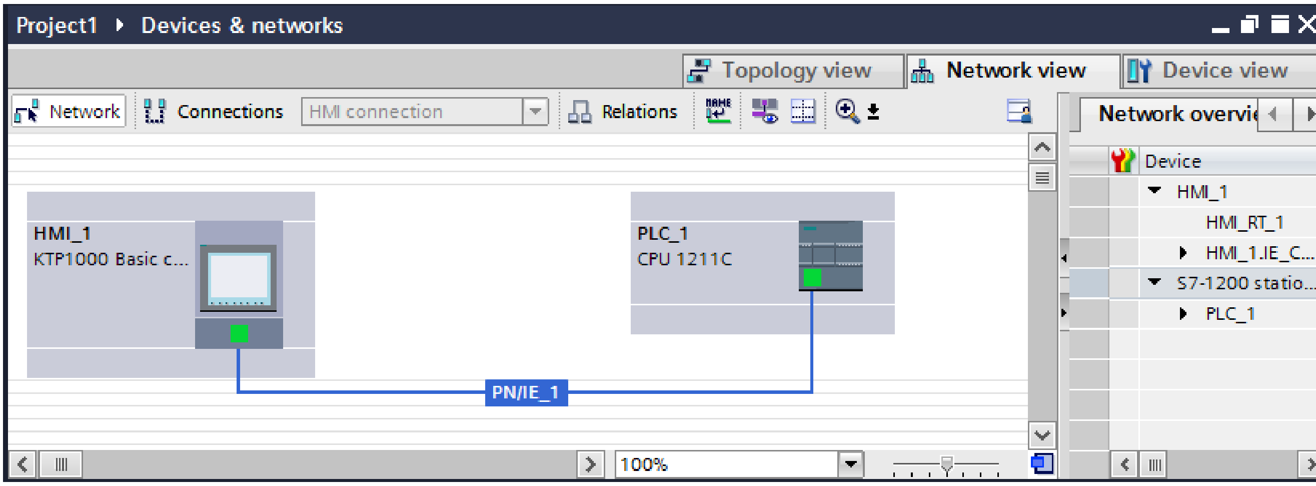
An HMI connection can be configured in the "Devices & Networks" editor for the following PLCs only:

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

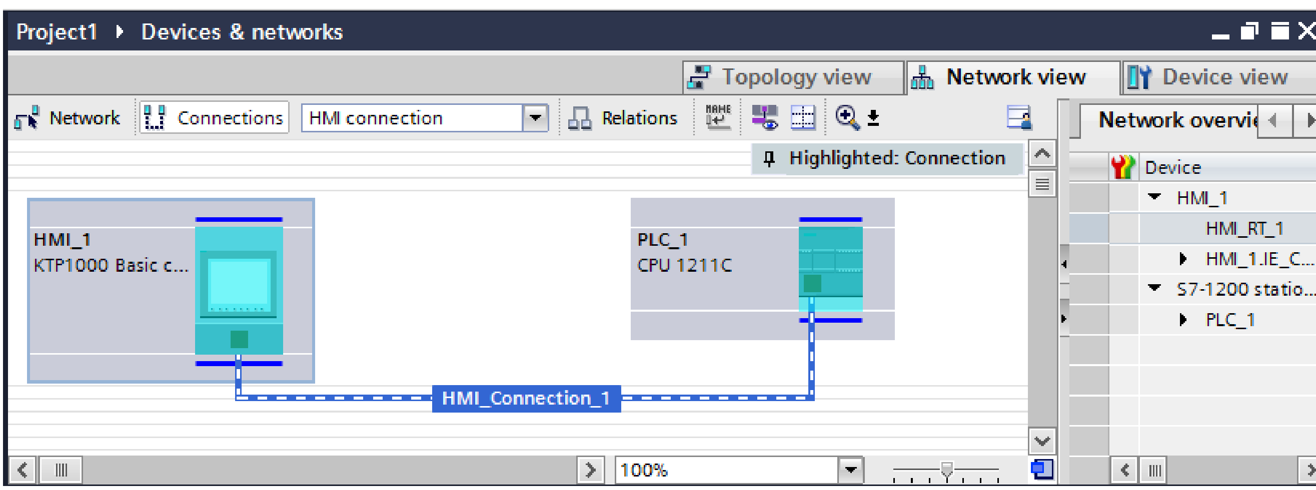
You configure the HMI connections to all other PLCs in the "Connections" editor of the HMI device.

Configuring an HMI connection in the "Devices & Networks" editor

1. Insert an HMI device and a SIMATIC S7 PLC in your project.



2. Switch to "Connections" mode.
3. Select the "HMI connection" connection type.
4. Use a drag-and-drop operation to interconnect the two PROFINET interfaces.



5. Change the IP address and subnet mask address parameters according to the requirements of your project.

1.3.2.3 Special considerations of the "Devices & Networks" editor (Basic Panels)

Introduction

If you are configuring or have already configured networks or HMI connections, the "Devices & Networks" editor supports you with the following functions:

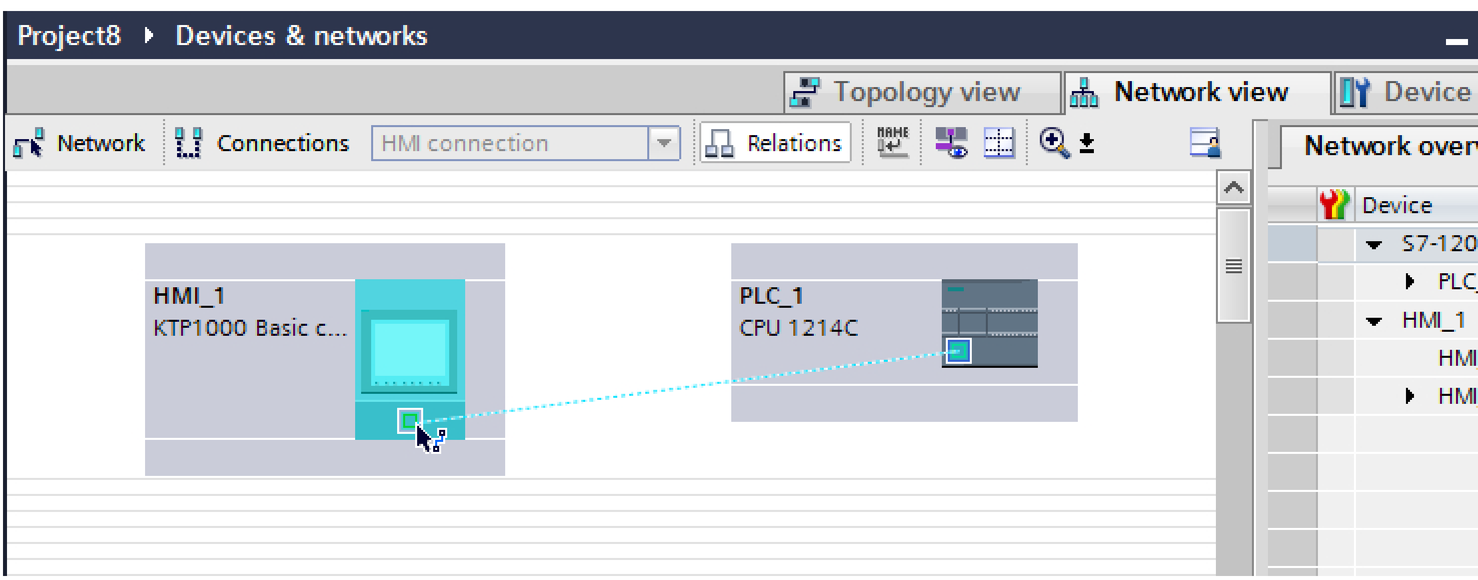
- Highlighting of communication partners
- Highlighting of HMI connections
- Automatic creation of subnets

Highlighting of communication partners

All communication partners for which an HMI connection is possible are highlighted in turquoise if you have selected the "HMI connection" type.

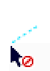
Starting from the interface of a device create an HMI connection to the device of another device using a drag-and-drop operation. During the drag-and-drop operation all potential communication partners are highlighted in turquoise.

Use the ESC key to stop connecting interfaces using a drag-and-drop operation.



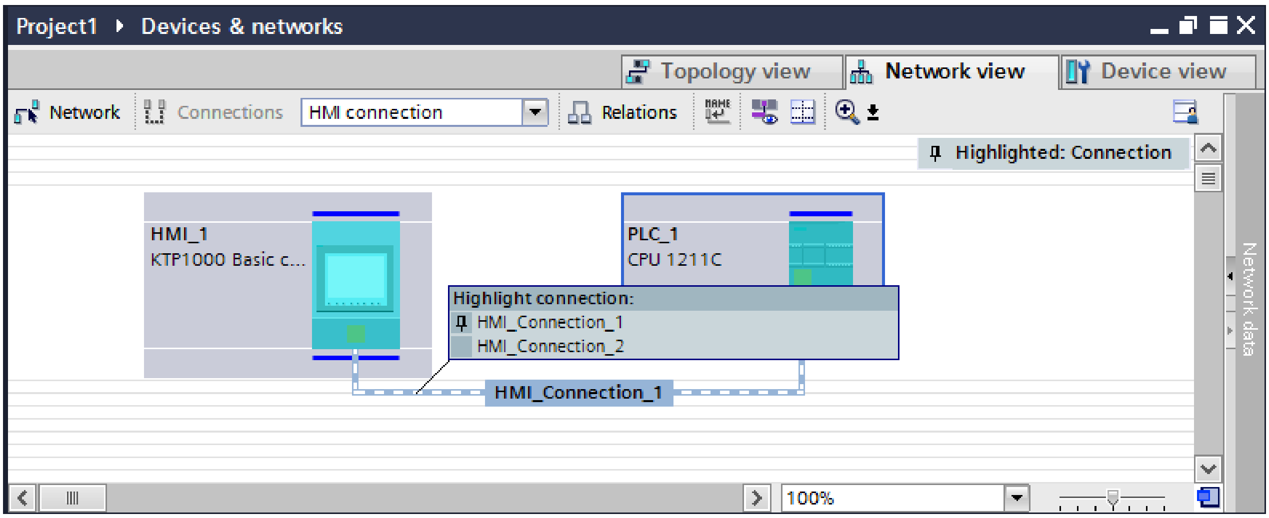
When the mouse pointer is moved over the interface of a device, the following icons indicate whether a connection is possible:

 A connection is possible.

 A connection is not possible.

Highlighting of HMI connections

A turquoise highlighting of the connection indicates that a HMI connection was created. If several HMI connections are created, you can select one of the already created HMI connections in a dialog.



Then you can configure the parameters of the selected HMI connection and the communication partners in the inspector window.

Subnets

Subnetworks are automatically created and used under the following conditions:

- If both communication partners are not already interconnected in different networks
- If a free interface is available to both communication partners.
- If a subnetwork already exists then that subnetwork is automatically used for the HMI connection.

1.3.2.4 Configuring a non-integrated connection in the "Connections" editor (Basic Panels)

Introduction

You use the "Connections" editor of the HMI device to configure a connection between an HMI device and a PLC that cannot be configured in the "Devices & Networks" editor.

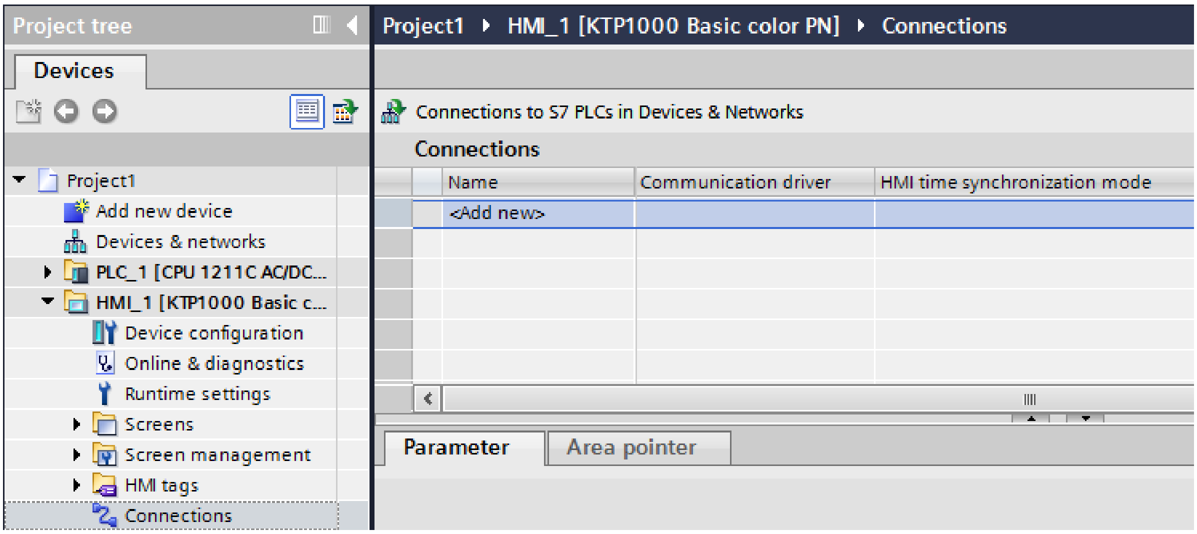
These connections are referred to as non-integrated connections.

Requirements

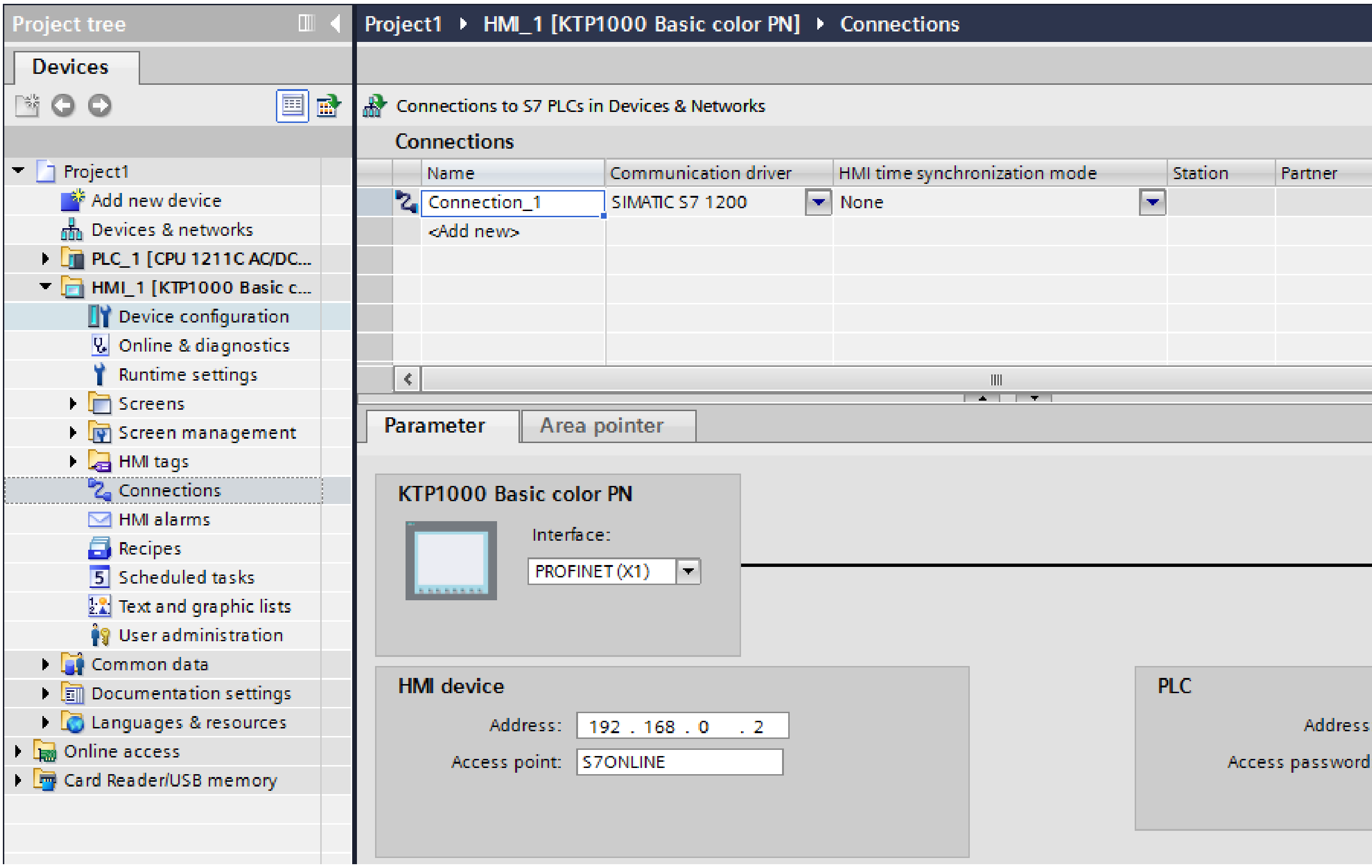
- A project is open.
- An HMI device has been created.

Configuring a connection in the "Connections" editor

1. Open the "Connections" editor of the HMI device.
2. Create a new connection.

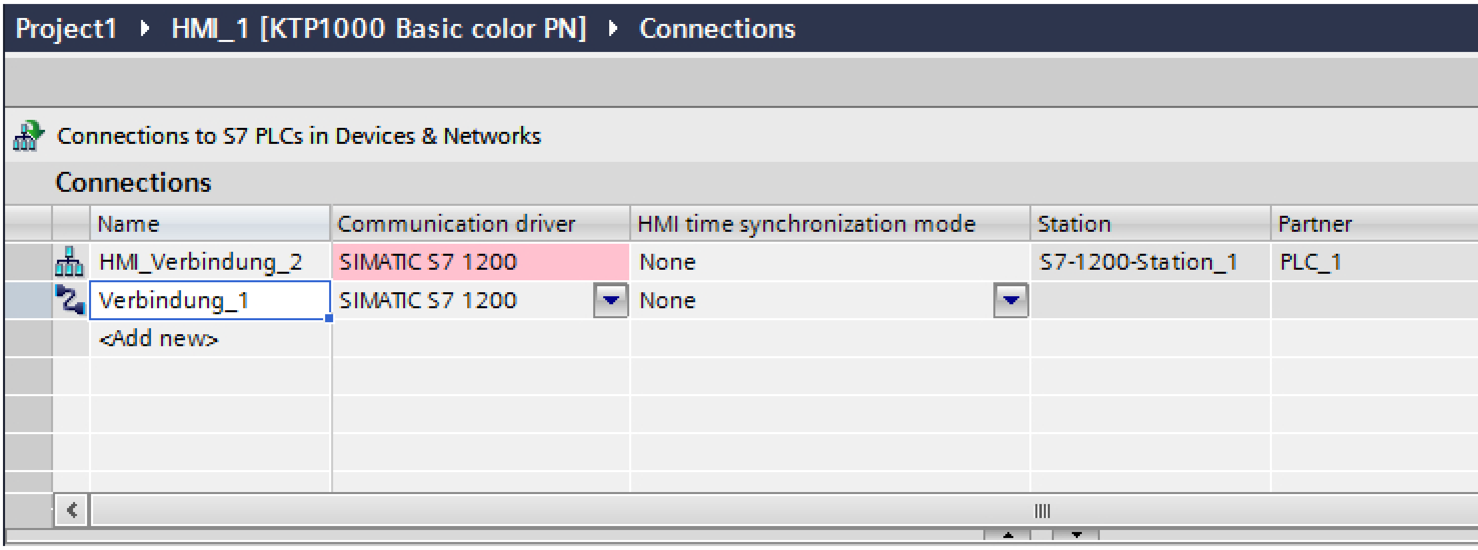


3. Select the communication driver.
4. Set the connection parameters.



Integrated connections in the "Connections" editor

If you have already configured the integrated connections of the HMI device in the "Devices & Networks" editor, they are also displayed in the "Connections" editor.



Meaning of the icons used:

 Integrated connection

 Non-integrated connection

1.3.2.5 Configuring a routed connection in the "Devices & Networks" editor (Basic Panels)

Introduction

You can configure a routed HMI connection to a PLC in another subnet in the "Devices & Networks" editor.

Note

A routed HMI connection can only be configured for the following PLCs:

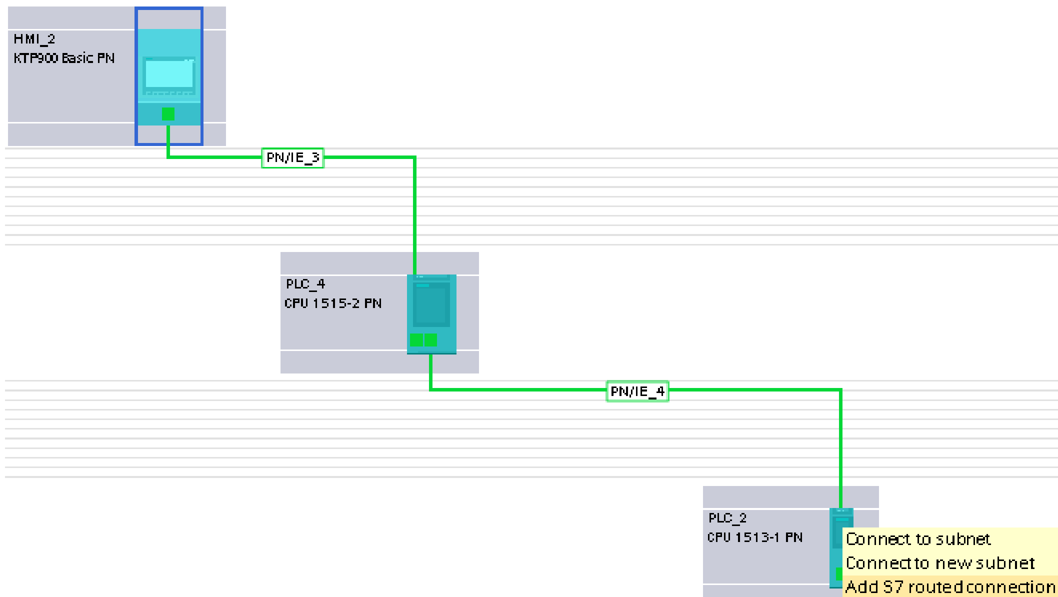
- SIMATIC S7 1500
- SIMATIC S7 1200

Requirement

- An HMI device has been created.
- PLCs have been set up in different networks
- The network view in the "Devices & Networks" editor is open.

Configuring a routed HMI connection

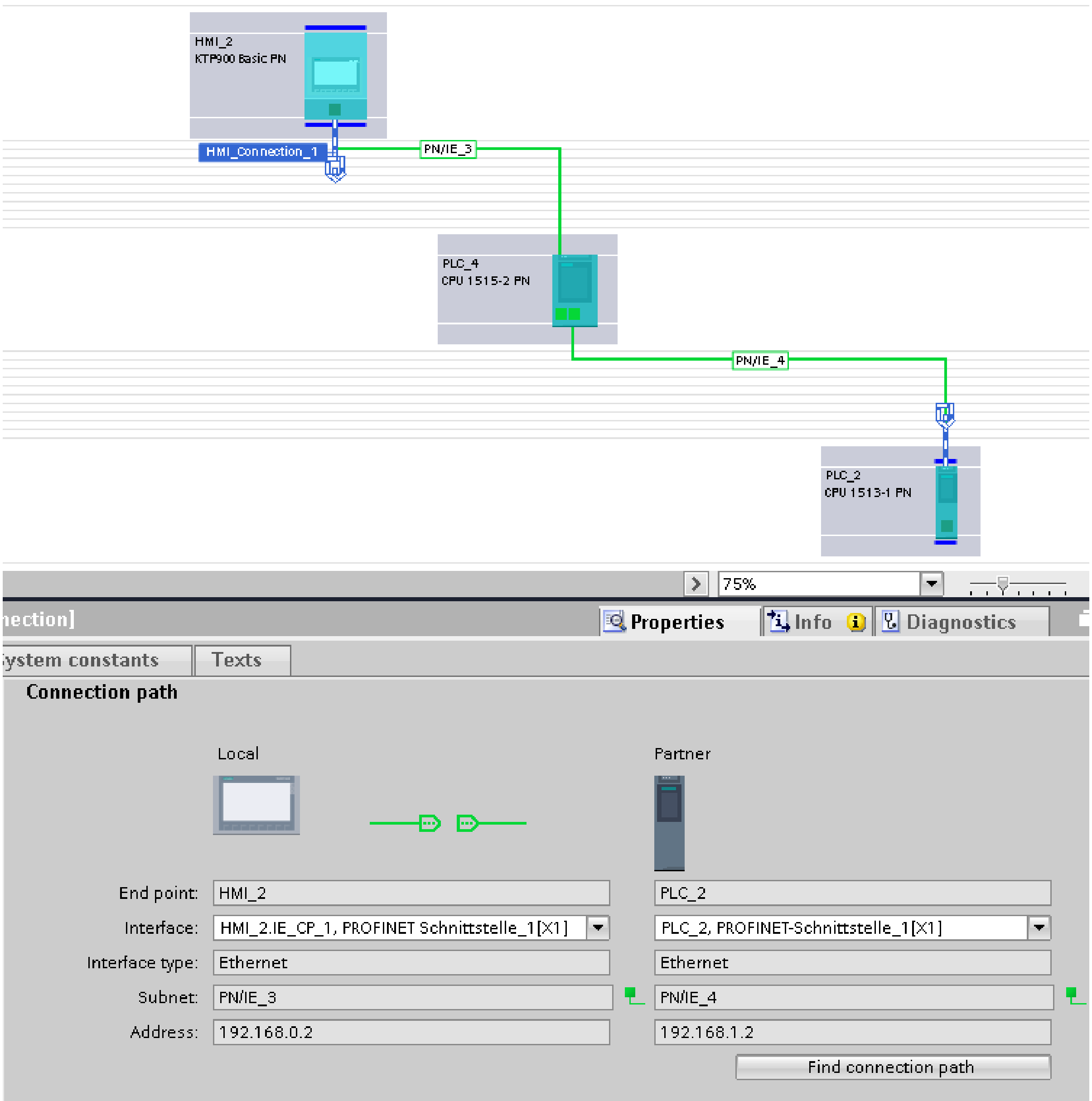
1. Switch to "Connections" mode.
2. Select the "HMI connection" connection type.
3. Drag-and-drop the HMI device to the PLC in the other subnet.
   A dialog opens.



4. Select "Add routed connection".

Result

The routed HMI connection has now been created. If you change the HMI device type or the configuration of the PLC, you will need to adapt the routed HMI connection again.



1.4 Data exchange (Basic Panels)

1.4.1 Data exchange using tags (Basic Panels)

1.4.1.1 Basics of tags (Basic Panels)

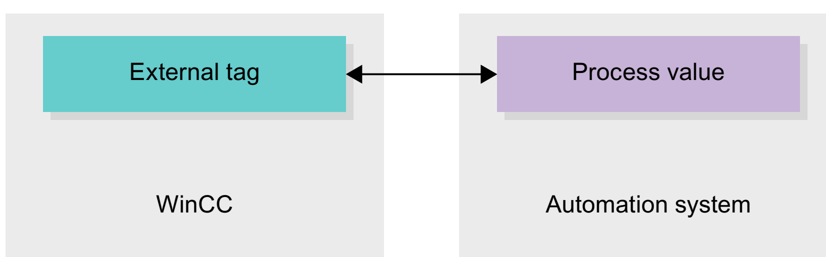
Introduction

Process values are forwarded in runtime using tags. Process values are data which is stored in the memory of one of the connected automation systems. They represent the status of a plant in the form of temperatures, fill levels or switching states, for example. Define external tags for processing the process values in WinCC.

WinCC works with two types of tag:

- External tags
- Internal tags

The external tags form the link between WinCC and the automation systems. The values of external tags correspond to the process values from the memory of an automation system. The value of an external tag is determined by reading the process value from the memory of the automation system. It is also possible to rewrite a process value in the memory of the automation system.



Internal tags do not have a process link and only convey values within the WinCC. The tag values are only available as long as runtime is running.

Tags in WinCC

For external tags, the properties of the tag are used to define the connection that the WinCC uses to communicate with the automation system and form of data exchange.

Tags that are not supplied with values by the process - the internal tags - are not connected to the automation system. In the tag's "Connection" property, this is identified by the "Internal tag" entry.

You can create tags in different tag tables for greater clarity. You then directly access the individual tag tables in the "HMI tags" node in the project tree. The tags from all tag tables can be displayed with the help of the table "Show all tags".

With structures you bundle a number of different tags that form one logical unit. Structures are project-associated data and are available for all HMI devices of the project. You use the "Types" editor in the project library to create and edit a structure.

1.4.1.2 Overview of HMI tag tables (Basic Panels)

Introduction

HMI tag tables contain the definitions of the HMI tags that apply across all devices. A tag table is created automatically for each HMI device created in the project.

In the project tree there is an "HMI tags" folder for each HMI device. The following tables can be contained in this folder:

- Default tag table
- User-defined tag tables
- Table of all tags

In the project tree you can create additional tag tables in the HMI tags folder and use these to sort and group tags and constants. You can move tags to a different tag table using a drag-and-drop operation or with the help of the "Tag table" field. Activate the "Tags table" field using the shortcut menu of the column headings.

Default tag table

There is one default tag table for each HMI device of the project. It cannot be deleted or moved. The default tag table contains HMI tags and, depending on the HMI device, also system tags. You can declare all HMI tags in the standard tags table or, as necessary, additional user-defined tables of tags.

User-defined tag tables

You can create multiple user-defined tag tables for each HMI device in order to group tags according to your requirements. You can rename, gather into groups, or delete user-defined tag tables. To group tag tables, create additional subfolders in the HMI tags folder.

All tags

The "All tags" table shows an overview of all HMI tags and system tags of the HMI device in question. This table cannot be deleted, renamed or moved. This table also contains the "Tags table" column, which indicates the tag table of where a tag is included. Using the "Tags table" field, the assignment of a tag to a tags table can be changed.

With devices for Runtime Professional, the table "All tags" contain an additional tab "System tags". The system tags are created by the system and used for internal management of the project. The names of the system tags begin with the "@" character. System tags cannot be deleted or renamed. You can evaluate the value of a system tag, but cannot modify it.

Additional tables

The following tables are also available in an HMI tag table:

- Discrete alarms
- Analog alarms
- Logging tags

With the help of these tables you configure alarms and logging tags for the currently selected HMI tag.

Discrete alarms table

In the "Discrete alarms" table, you configure discrete alarms to the HMI tag selected in the HMI tag table. When you configure a discrete alarm, multiple selection in the HMI tag table is not possible. You configure the discrete alarms for each HMI tag separately.

Analog alarms table

In the "Analog alarms" table, you configure analog alarms to the HMI tag selected in the HMI tag table. When you configure an analog alarm, multiple selection in the HMI tag table is not possible. You configure the analog alarms for each HMI tag separately.

Logging tags table

In the "Logging tags" table, you configure logging tags to the HMI tag selected in the HMI tag table. When you configure a logging tag, multiple selection in the HMI tag table is not possible. You configure the logging tags for each HMI tag separately. The "Logging tags" table is only available if the HMI device used supports logging.
If WinCC Runtime Professional is used, you can also assign several log tags to a tag. With the other HMI devices, you can only assign one log tag to a tag.

1.4.1.3 External tags (Basic Panels)

Introduction

External tags allow communication (data exchange) between the components of an automation system, for example, between an HMI device and a PLC.

Principle

An external tag is the image of a defined memory location in the PLC. You have read and write access to this storage location from both the HMI device and from the PLC.

Since external tags are the image of a storage location in the PLC, the applicable data types depend on the PLC which is connected to the HMI device.

If you write a PLC control program in STEP 7, the PLC tags created in the control program will be added to the PLC tag table. If you want to connect an external tag to a PLC tag, access the PLC tags directly via the PLC tag table and connect them to the external tag.

Data types

All the data types which are available at the connected PLC are available at an external tag in WinCC. Information about data types which are available for connection to other PLCs can be found in the documentation about the respective communication drivers.

See "AUTOHOTSPOT" for additional information.

Note

As well as external tags, area pointers are also available for communication between the HMI device and PLC. You can set up and enable the area indicators in the "Connections" editor.

Central tag management in STEP 7

You can connect also connect DB instances of user-defined PLC data types (UDT) to the HMI tags.

The PLC data type and the corresponding DB instances are created and updated centrally in STEP 7. In WinCC, you can use the following sources as PLC tag (DB instances):

- Data block elements that use a UDT as data type
- Data block instances of a UDT

The data type is taken from STEP 7 and is not converted to an HMI data type. The access mode is always "Symbolic access". Depending on the release for WinCC in STEP 7, elements and structured elements of the PLC data type are applied to WinCC. Elements of a structured UDT are applied and displayed in the PLC tag table if the instance-specific properties "Visible in HMI" and "Accessible from HMI" have been set.

Note

Accessing PLC data types

Access to PLC data types is available only in conjunction with SIMATIC S7-1200 or S7-1500.

Synchronization with PLC tags

A variety of options for synchronizing external tags with the PLC tags are available in the Runtime settings under "Settings for tags".

When you perform synchronization, you have the option of automatically applying the tag names of the PLC to external tags and reconnecting the existing tags.

The generated tag name is derived from the position of the data value in the hierarchical structure of the data block.

Update of tag values

For external tags, the current tag values are transmitted in Runtime via the communication connection between WinCC and the connected automation systems and then saved in the Runtime memory. Next, the tag value will be updated to the set cycle time. For use in the Runtime project, WinCC accesses tag values in the Runtime memory that were read from the PLC at the previous scan cycle checkpoint. As a result, the value in the PLC can already change whilst the value from the Runtime memory is being processed.

Note

PLC array elements in conjunction with S7-1200 or S7-1500

The index of the PLC array elements can begin with any number. In WinCC, indexing always starts with 0.

A PLC tag "Array [1..3] of Int", for example, is mapped to "Array [0..2] of Int" in WinCC.

When you access an array in a script, pay attention to the correct indexing.

1.4.1.4 Addressing external tags (Basic Panels)

Introduction

The options for addressing external tags depend on the type of connection between WinCC and the PLC in question. A distinction must be made between the following connection types:

- Integrated connection
  Connections of devices which are within a project and were created with the "Devices & Networks" editor are referred to as integrated connections.
- Non-integrated connection
  Connections of devices which were created with the "Connections" editor are referred to as non-integrated connections. It is not necessary that all of the devices be within a single project.

The connection type can also be recognized by its icon.

	Integrated connection
	Non-integrated connection

You can find additional information in the section "Basics of communication (Page 9)".

Addressing with integrated connections

An integrated connection offers the advantage that you can address a tag both symbolically and absolutely.

For symbolic addressing, you select the PLC tag via its name and connect it to the HMI tag. The valid data type for the HMI tag is automatically selected by the system. You have to distinguish between the following cases when you address elements in data blocks:

Symbolic addressing of data blocks with optimized access and standard access:

During the symbolic addressing of a data block with optimized access and standard access, the address of an element in the data block is dynamically assigned and is automatically adopted in the HMI tag in the event of a change. You do not need to compile the connected data block or the WinCC project for this step.
For data blocks with optimized access, only symbolic addressing is available.

For symbolic addressing of elements in a data block, you only need to recompile and reload the WinCC project in case of the following changes:

- If the name or the data type of the linked data block element or global PLC tag has changed.
- If the name or the data type of the higher level structure node of a linked element in the data block element or global PLC tag has changed.
- If the name of the connected data block has changed.

Symbolic addressing is currently available with the following PLCs:

- SIMATIC S7 1200
- SIMATIC S7 1500

Symbolic addressing is also available if you have an integrated link.

You can also use absolute addressing with an integrated connection. You have to use absolute addressing for PLC tags from a SIMATIC S7 300/400 PLC. If you have connected an HMI tag with a PLC tag and the address of the PLC tag changes, you only have to recompile the control program to update the new address in WinCC. Then you recompile the WinCC project and load it onto the HMI device.

In WinCC, symbolic addressing is the default method. To change the default setting, select the menu command "Options > Settings". Select "Visualization > Tags" in the "Settings" dialog. If required, disable the "Symbolic access" option.

The availability of an integrated connection depends on the PLC used. The following table shows the availability:

PLC	Integrated connection	Comments
S7 300/400	Yes	The linking of tags is not checked in Runtime. If the tag address changes in the PLC and the HMI device is not compiled again and loaded, the change is not registered in runtime.
S7 1200	Yes	A validity check of the tag connection is performed in runtime during symbolic addressing. If an address is changed in the PLC, the change is registered and an error message is issued. In the case of absolute addressing, the following behavior applies to the S7 300/400.
S7-1500	Yes	A validity check of the tag connection is performed in runtime during symbolic addressing. If an address is changed in the PLC, the change is registered and an error message is issued. In the case of absolute addressing, the following behavior applies to the S7 300/400.

Create an integrated connection in the "Devices & Networks" editor. If the PLC is contained in the project and integrated connections are supported, you can then also have the connection created automatically. To do this, when configuring the HMI tag, simply select an existing PLC tag to which you want to connect the HMI tag. The integrated connection is then automatically created by the system.

Addressing with non-integrated connections

In the case of a project with a non-integrated connection, you always configure a tag connection with absolute addressing. Select the valid data type yourself. If the address of a PLC tag changes in a project with a non-integrated connection during the course of the project, you also have to make the change in WinCC. The tag connection cannot be checked for validity in Runtime, an error message is not issued.

A non-integrated connection is available for all supported PLCs.

Symbolic addressing is not available in a non-integrated connection.

With a non-integrated connection, the control program does not need to be part of the WinCC project. You can perform the configuration of the PLC and the WinCC project independently of each other. For configuration in WinCC, only the addresses used in the PLC and their function have to be known.

See also

Basics of communication (Page 9)

1.4.1.5 Internal Tags (Basic Panels)

Introduction

Internal tags do not have any connection to the PLC. Internal tags transport values within the HMI device. The tag values are only available as long as runtime is running.

Principle

Internal tags are stored in the memory of the HMI device. Therefore, only this HMI device has read and write access to the internal tags. You can create internal tags to perform local calculations, for example.

You can use the HMI data types for internal tags. Availability depends on the HMI device being used.

The following HMI data types are available:

HMI data type	Data format
Array	One-dimensional array
Bool	Binary tag
DateTime	Date/time format
DInt	Signed 32-bit value
Int	Signed 16-bit value
LReal	Floating-point number 64-bit IEEE 754
Real	Floating-point number 32-bit IEEE 754

HMI data type	Data format
SInt	Signed 8-bit value
UDnt	Unsigned 32-bit value
UInt	Unsigned 16-bit value
USInt	Unsigned 8-bit value
WString	Text tag, 16-bit character set

1.4.2 Data exchange using area pointers (Basic Panels)

1.4.2.1 Basic information on area pointers (Basic Panels)

Introduction

You use an area pointer to access a data area in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations. The PLC and the HMI device trigger defined interactions based on the evaluation of stored data.

For example, area pointers are required for the following data:

- Recipes
- Job mailboxes
- Sign-of-life monitoring

Area pointer

The following area pointers are supported:

Area pointer

Area pointers can be configured for connections.

- Data record
- Date/time
- Coordination
- Job mailbox

Global area pointers of the HMI device

Global area pointers can be configured to only one connection for each project.

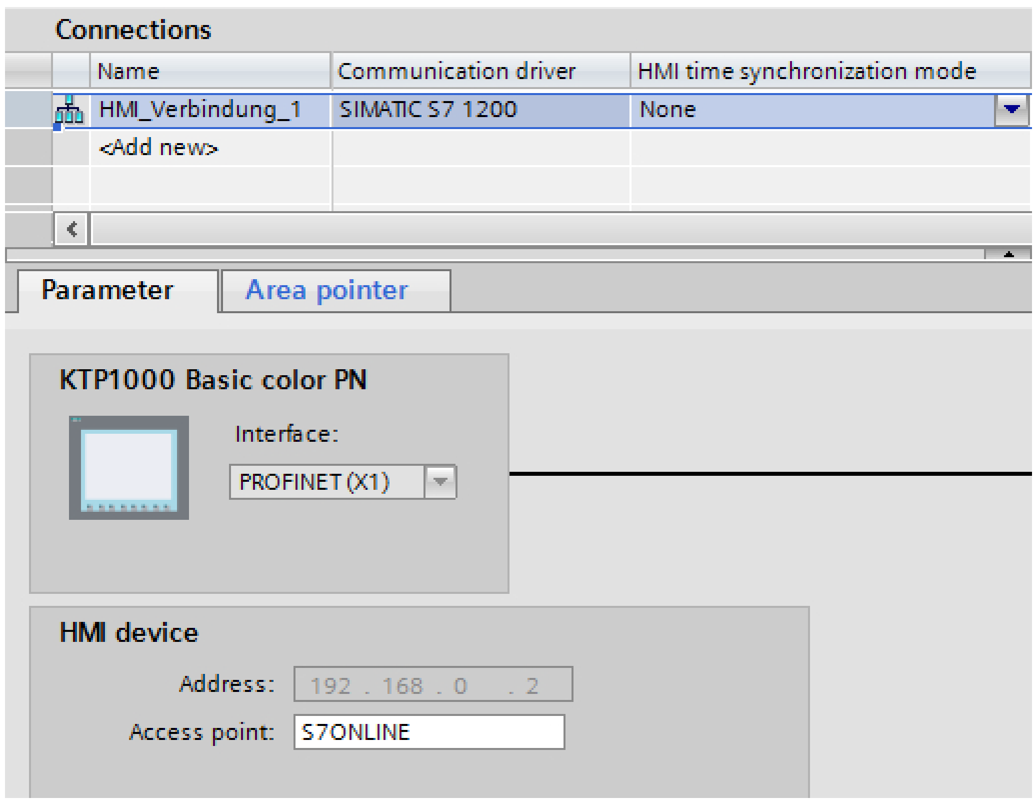
- Screen number
- Date/time PLC
- Project ID

1.4.2.2 Area pointers for connections (Basic Panels)

Introduction

Using the "Area pointer" tab of the "Connections" editor, you can configure the usage of the available area pointers.

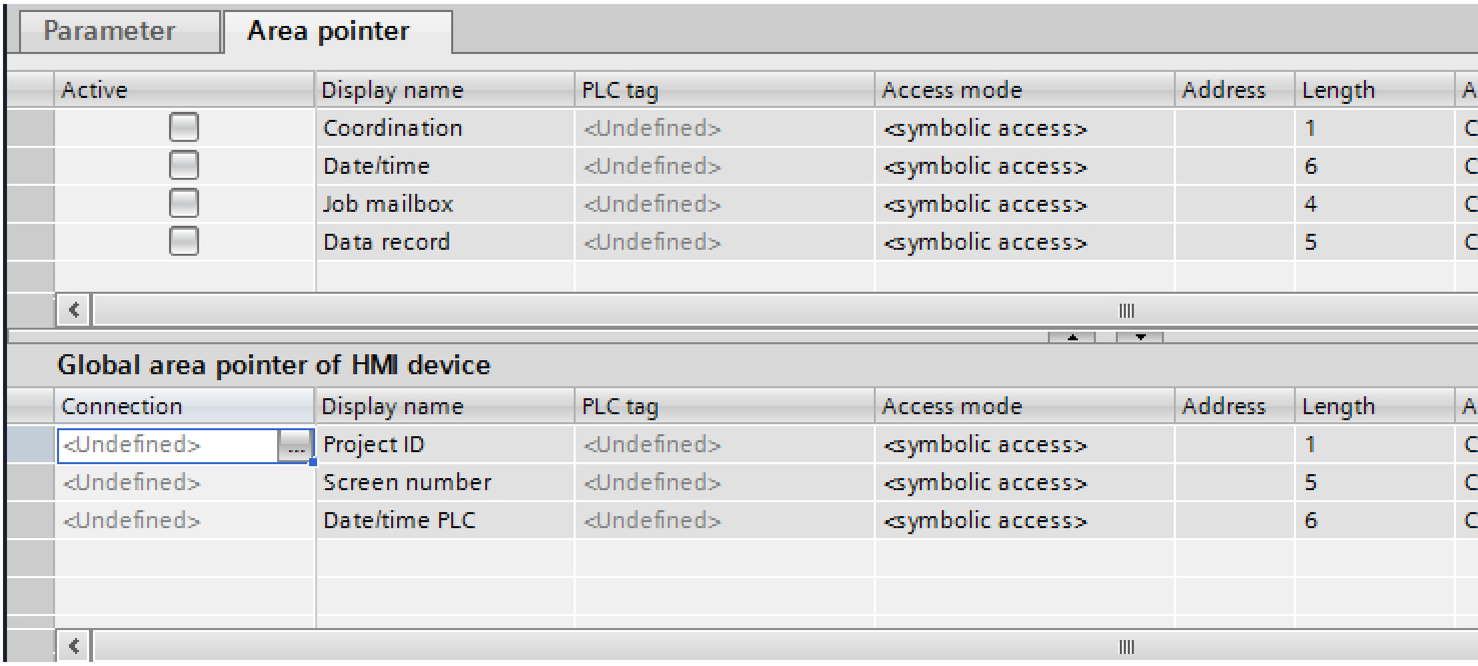
To configure the area pointers, open the "Connections" editor and open the "Area pointer" tab.



Structure

The "Area pointer" tab contains two tables of area pointers. The top part of the table contains the area pointers you can create and enable separately for each available connection.

The "Global area pointers of HMI device" table contains the area pointers which are created only once in the project and can be used for only one connection.

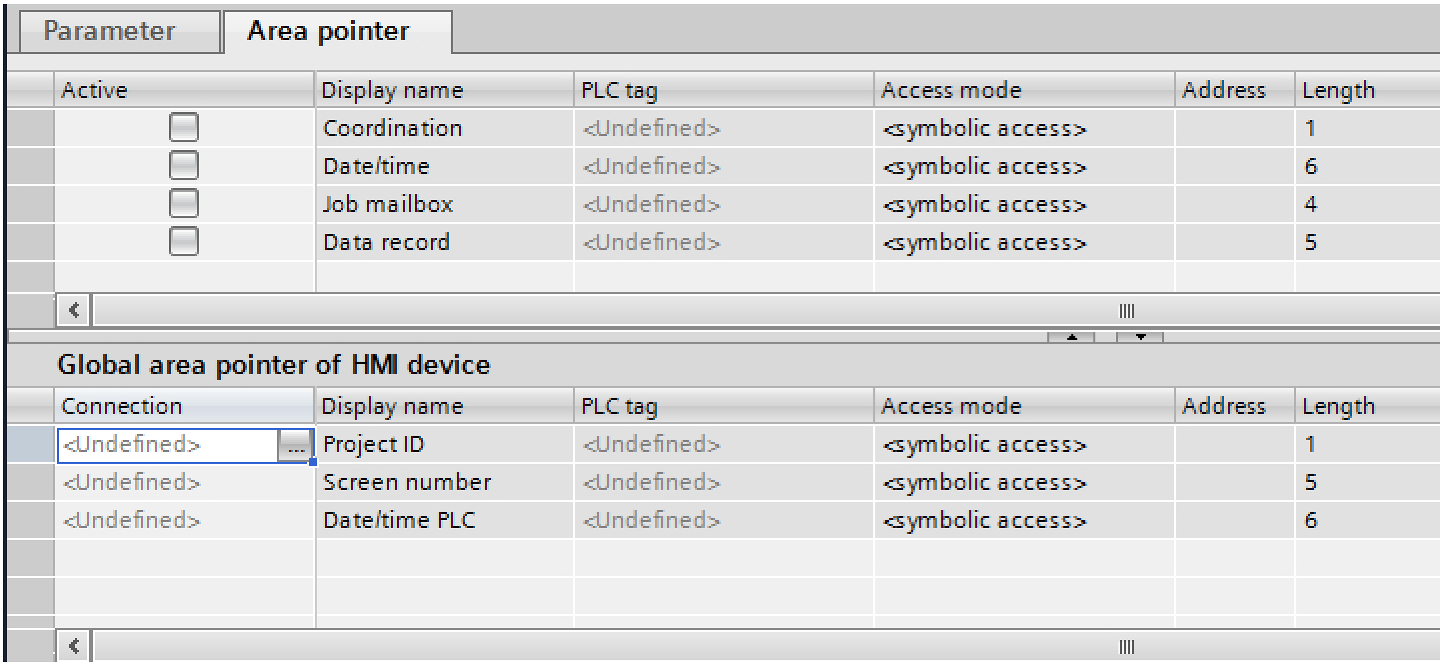


1.4.2.3 Use of area pointers (Basic Panels)

"Area pointer" tab

Before you use an area pointer, you enable it under "Connections > Area pointer". You then assign the area pointer parameters.

You assign the following parameters in the "Area pointer" tab:



- Active
  Enables the area pointer.
- Pointer name
  Name of the area pointer specified by WinCC.
- PLC tag
  Here you select the PLC tag or the tag array that you have configured as the data area for the area pointer.
- Access mode
  Here you can select from the following access modes:
  - Symbolic access
  - Absolute access
- Address
  If you selected "Symbolic access", no address is entered in this field.
  If you selected "Absolute access", enter the address of a tag in the "Address" field.
- Length
  WinCC specifies the length of the area pointer.
- Acquisition cycle
  You specify the acquisition cycle in this field for area pointers that are read by the HMI device. Note that a very short acquisition time may have a negative impact on HMI device performance.
- Comment
  Enter a comment, for example, to describe the purpose of the area pointer.

1.4.2.4 Accessing data areas (Basic Panels)

Accessing data areas

The following table shows how HMI devices and PLCs access individual data areas for read (R) or write (W) operations.

Data area	Required for	HMI device	PLC
Screen number	Evaluation by the PLC in order to determine the active screen.	W	R
Data record	Transfer of data records with synchronization	R/W	R/W
Date/time	Transfer of the date and time from the HMI device to the PLC	W	R
Date/time PLC	Transfer of the date and time from the PLC to the HMI device	R	W
Coordination	Requesting the HMI device status in the PLC program	W	R
Project ID	Runtime checks for consistency between the WinCC project ID and the project in the PLC	R	W
Job mailbox	Triggering of HMI device functions by the PLC program	R/W	R/W

1.4.2.5 Configuring area pointers (Basic Panels)

Configuration of area pointers (Basic Panels)

Introduction

You use an area pointer to access a data area in the PLC. The data area is stored in the PLC.

Prior to configuring area pointers

Before you use the area pointer, you must enable and parameterize it under "Connections > Area Pointer".

Global data block

To access the data area in the PLC, you have to create a global data block in the PLC program. The following example shows the use of a data block.

Length of area pointers

For area pointers with a length >= 1, you set up the data area as a tag array in a global data block or instance data block.

You also have the option to use a PLC tag for area pointers with a length = 1.

The configuration of the tags in a data block is dependent on the length of the area pointer you want to use. The unit for the length of an area pointer is a 16-bit word.

If, for example, you want to use an area pointer with a length of "5", you must create an array with 5 array elements of the data type UINT in the data block.

Alternative procedure

Alternatively, you can also use the absolute access mode to access area pointers. Absolute access mode only works on standard PLC data blocks.

Parameterizing a global data block (Basic Panels)

Introduction

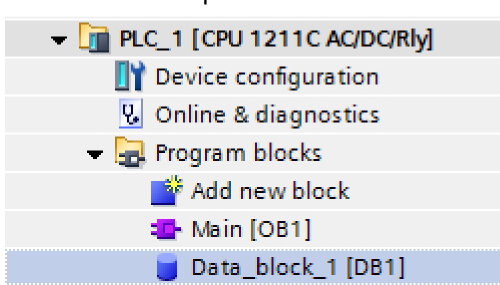
To access the data area in the PLC, a global data block for the area pointer must be parameterized in the PLC program.

Requirements

- A PLC is created in the project.
- A connection is configured between the PLC and the HMI device.
- The PLC program contains a global data block.

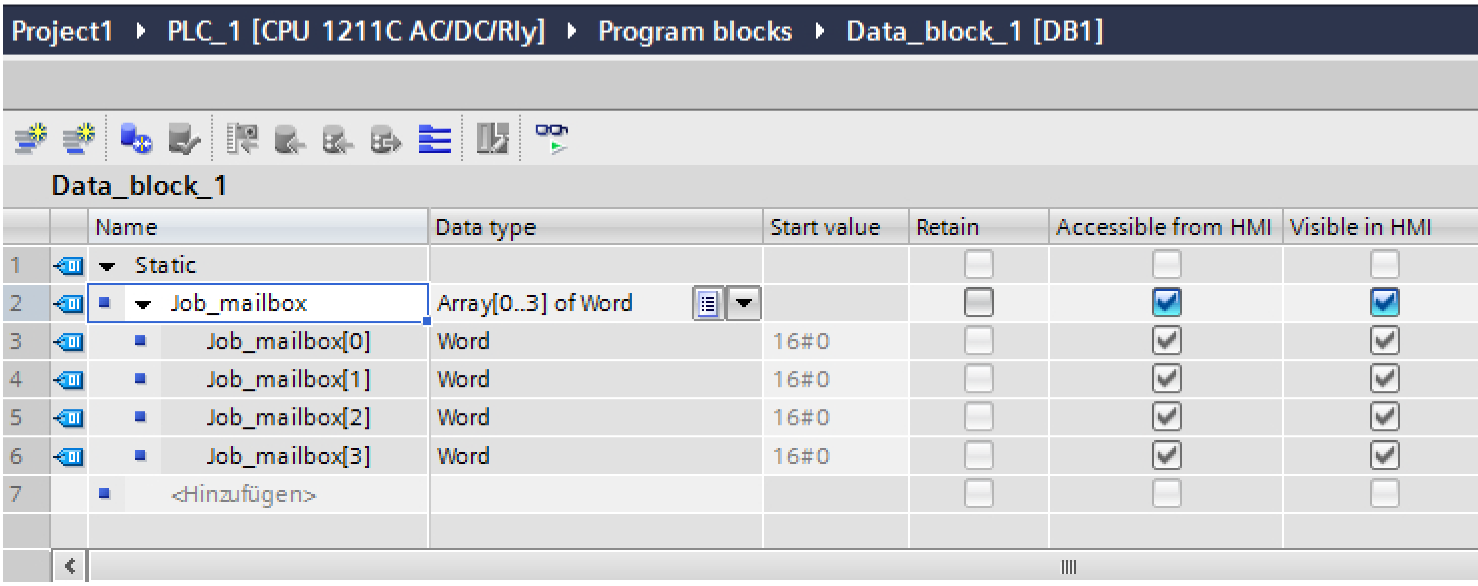
Procedure

1. Open "PLC > Program blocks" in the project tree.
2. Double-click the global data block you created previously.
   The data block opens.



3. Enter a tag name in the "Name" column.
4. Select "Array[lo .. hi] of type" as the data type in the "Data type" column.
5. Replace the "lo" entry by the low value for the dimension of the array.
6. Replace the "hi" entry by the high value for the dimension of the array.
   Example: If you configure an area pointer with the length "4", enter the value "0" for "lo" and the value "3" for "hi" inside the brackets.

7. Replace the "type" designation with the "word" data type.
   The full data type for an array of 4 tags appears as follows: "Array[0 .. 3] of word".
   The tag array is created after the entry is confirmed.
8. Click "Compile".
   The project is compiled.



Configuring an area pointer for a connection (Basic Panels)

Introduction

After you have parameterized the global data block, you now create the area pointer for the connection.

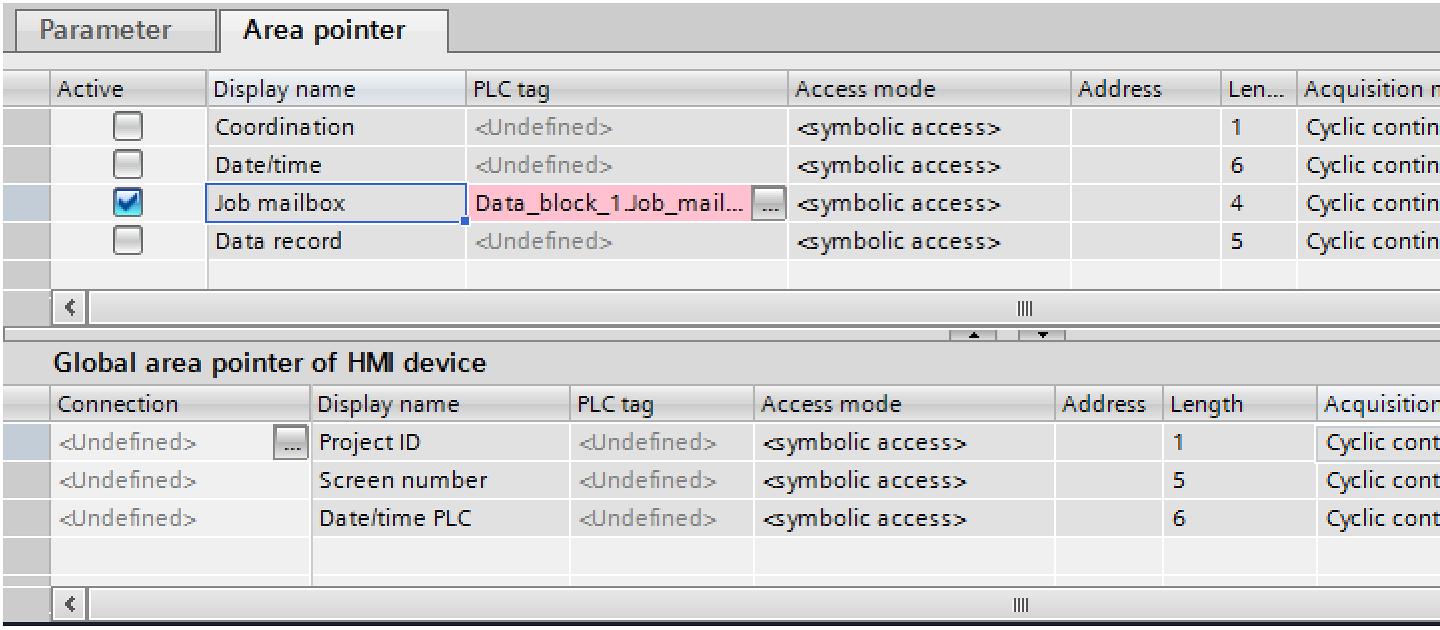
Requirements

- The global data block has been parameterized in the PLC program.

Procedure

1. Open "HMI >Connections" in the project tree.
2. Click the "Area pointer" tab.
3. Enable the required area pointer.
   You enable a global area pointer by selecting the connection in the "Connection" field.
4. Click the navigation button in the "PLC tag" field.
   The object list opens.

5. Navigate to the data block in the object list, and select the tag in the right window.
   You do not need an array tag to configure an area pointer with the length of "1".



6. Select the "Word" data type when creating the tag in the data block.

If required, set additional parameters, such as the acquisition cycle, during configuration.

Result

The area pointer is enabled and connected to the PLC tag in the global data block.

1.5 Device dependency (Basic Panels)

1.5.1 Basic Panel (Basic Panels)

1.5.1.1 Communication drivers (Basic Panels)

SIMATIC communication drivers (Basic Panels)

Device dependency of the Basic Panels

Various communication drivers are configured for Basic Panels depending on the communication.

The following tables show the communication drivers released for integrated and non-integrated communication.

There are different versions of the HMI devices within the integrated communication.

Communication drivers for integrated communication (V11.0)

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)	SIMATIC S7-1500	SIMATIC S7-300/400
KP300 Basic	Yes	Yes	Yes	Yes	Yes	No	Yes
KP400 Basic	Yes	Yes	Yes	Yes	Yes	No	Yes
KTP400 Basic PN	Yes	Yes	Yes	Yes	Yes	No	Yes
KTP600 Basic DP	Yes	Yes	Yes	Yes	Yes	No	Yes
KTP600 Basic PN	Yes	Yes	Yes	Yes	Yes	No	Yes
KTP1000 Basic DP	Yes	Yes	Yes	Yes	Yes	No	Yes
KTP1000 Basic PN	Yes	Yes	Yes	Yes	Yes	No	Yes
TP1500 Basic PN	Yes	Yes	Yes	Yes	Yes	No	Yes

Communication drivers for integrated communication (V12.0)

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)	SIMATIC S7-1500	SIMATIC S7-300/400
KP300 Basic	No	Yes	Yes	Yes	Yes	Yes	Yes
KP400 Basic	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP400 Basic PN	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP600 Basic DP	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP600 Basic PN	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP1000 Basic DP	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP1000 Basic PN	No	Yes	Yes	Yes	Yes	Yes	Yes
TP1500 Basic PN	No	Yes	Yes	Yes	Yes	Yes	Yes

Communication drivers for integrated communication (V13.0)

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)	SIMATIC S7-1500	SIMATIC S7-300/400
KTP400 Basic	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP700 Basic	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP900 Basic	No	Yes	Yes	Yes	Yes	Yes	Yes
KTP1200 Basic	No	Yes	Yes	Yes	Yes	Yes	Yes

Communication drivers for non-integrated communication

HMI devices	SIMATIC S7-1200	SIMATIC S7-1500	SIMATIC S7-300/400	SIMATIC S7-200	SIMATIC LOGO!	SIMATIC HTTP protocol
KP300 Basic	Yes	Yes	Yes	Yes	Yes	No
KP400 Basic	Yes	Yes	Yes	Yes	Yes	No
KTP400 Basic PN	Yes	Yes	Yes	Yes	Yes	No
KTP600 Basic DP	Yes	Yes	Yes	Yes	Yes	No
KTP600 Basic PN	Yes	Yes	Yes	Yes	Yes	No
KTP700 Basic PN	Yes	Yes	Yes	Yes	Yes	No
KTP900 Basic PN	Yes	Yes	Yes	Yes	Yes	No
KTP1000 Basic DP	Yes	Yes	Yes	Yes	Yes	No
KTP1000 Basic PN	Yes	Yes	Yes	Yes	Yes	No
TP1500 Basic PN	Yes	Yes	Yes	Yes	Yes	No

Other communication drivers (Basic Panels)

Device dependency of the Basic Panels

The following table shows which communication drivers you can configure with the various Basic Panels.

Communication drivers

HMI devices	OPC	SIMATIC HTTP protocol	Allen-Bradley EtherNet/IP	Allen-Bradley DF1	Mitsubishi MC TCP/IP	Mitsubishi FX	Modicon Modbus TCP/IP	Modicon Modbus RTU	Omron Host Link
KP300 Basic	No	No	Yes	No	Yes	No	Yes	No	No
KP400 Basic	No	No	Yes	No	Yes	No	Yes	No	No
KTP400 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No
KTP600 Basic DP	No	No	No	Yes [2)]	No	Yes	No	Yes [1)]	Yes
KTP600 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No
KTP700 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No
KTP700 Basic DP	No	No	No	Yes [2)]	No	Yes	No	Yes [1)]	Yes
KTP900 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No
KTP1000 Basic DP	No	No	No	Yes [2)]	No	Yes	No	Yes [1)]	Yes
KTP1000 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No
KTP1200 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No
KTP1200 Basic DP	No	No	No	Yes [2)]	No	Yes	No	Yes [1)]	Yes
TP1500 Basic PN	No	No	Yes	No	Yes	No	Yes	No	No

1) only with RS 422-RS232 converter
Order number of the converter: 6AV6 671-8XE00-0AX0

2) Direct communication with PLC 5 or KF2 module, otherwise only approved with RS422-RS232 converter (option).
Order number of the converter: 6AV6 671-8XE00-0AX0

1.5.1.2 Interfaces of the Basic Panels (Basic Panels)

Device dependency of the Basic Panels

The following table shows which HMI device interfaces are available for the communication driver protocols.

Table 1-1 Basic Panels

	KP300 Basic **KP400 Basic** **KTP400 Basic PN** **KTP600 Basic PN** **KTP700 Basic PN** **KTP900 Basic PN** **KTP1200 Basic PN** **KTP1000 Basic PN** **TP1500 Basic PN**	**KTP600 Basic DP** **KTP700 Basic DP** **KTP1000 Basic DP** **KTP1200 Basic DP**
SIMATIC S7 - PPI [1)]	—	MPI/DP (X2)
SIMATIC S7 - MPI	—	MPI/DP (X2)
SIMATIC S7 - PROFIBUS	—	MPI/DP (X2)
SIMATIC S7 - PROFINET	PROFINET (X1)	—
SIMATIC HMI HTTP protocol	—	—
OPC	—	—
Allen-Bradley EtherNet/IP	PROFINET (X1)	—
Allen-Bradley DF1	—	MPI/DP (X2) [2)]
Mitsubishi TCP/IP	PROFINET (X1)	—
Mitsubishi FX	—	MPI/DP (X2) (RS422)
Modicon Modbus TCP	PROFINET (X1)	—
Modicon Modbus RTU	—	MPI/DP (X2) [3)]
Omron Host Link	—	MPI/DP (X2) (RS422)

1) For SIMATIC S7-200 only

2) Direct communication with PLC5 or KF2 module, otherwise only approved with RS422-RS232 converter (option).
Order number: 6AV6 671-8XE00-0AX0

3) only approved with RS 422-RS232 converter
Order number: 6AV6 671-8XE00-0AX0

1.5.1.3 Area pointers for Basic Panels (Basic Panels)

Introduction

Area pointers are parameter fields from which the HMI device obtains information about the location and size of data areas in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations. Based on the evaluation of data stored in the data areas, the PLC and HMI device initiate mutually defined actions.

WinCC uses the following area pointers:

- Job mailbox
- Project ID
- Screen number
- Data record
- Date/time
- Date/time PLC
- Coordination

Availability of the area pointers

The following table shows the availability of the area pointers on the HMI devices. Note that the area pointers can be used only for available communication drivers.

Area pointer

	KP300 Basic	KTP400 Basic PN	KTP600 Basic PN	KTP600 Basic DP	KTP700 Basic PN / DP	KTP900 Basic PN	KTP1000 Basic PN / DP	KTP1200 Basic PN / DP	TP1500 Basic PN
Screen number	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Data record	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Date/time	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Date/time PLC	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Coordination	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Project ID	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Job mailbox	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

1.6 Communicating with SIMATIC S7 1500 (Basic Panels)

1.6.1 Communication with SIMATIC S7 1500 (Basic Panels)

Introduction

This section describes the communication between an HMI device and the SIMATIC S7 1500 PLC.

You can configure the following communication channels for the SIMATIC S7 1500 PLC:

- PROFINET
- PROFIBUS

HMI connection for communication

Configure connections between the HMI device and a SIMATIC S7 1500 in the "Devices & Networks" Editor.

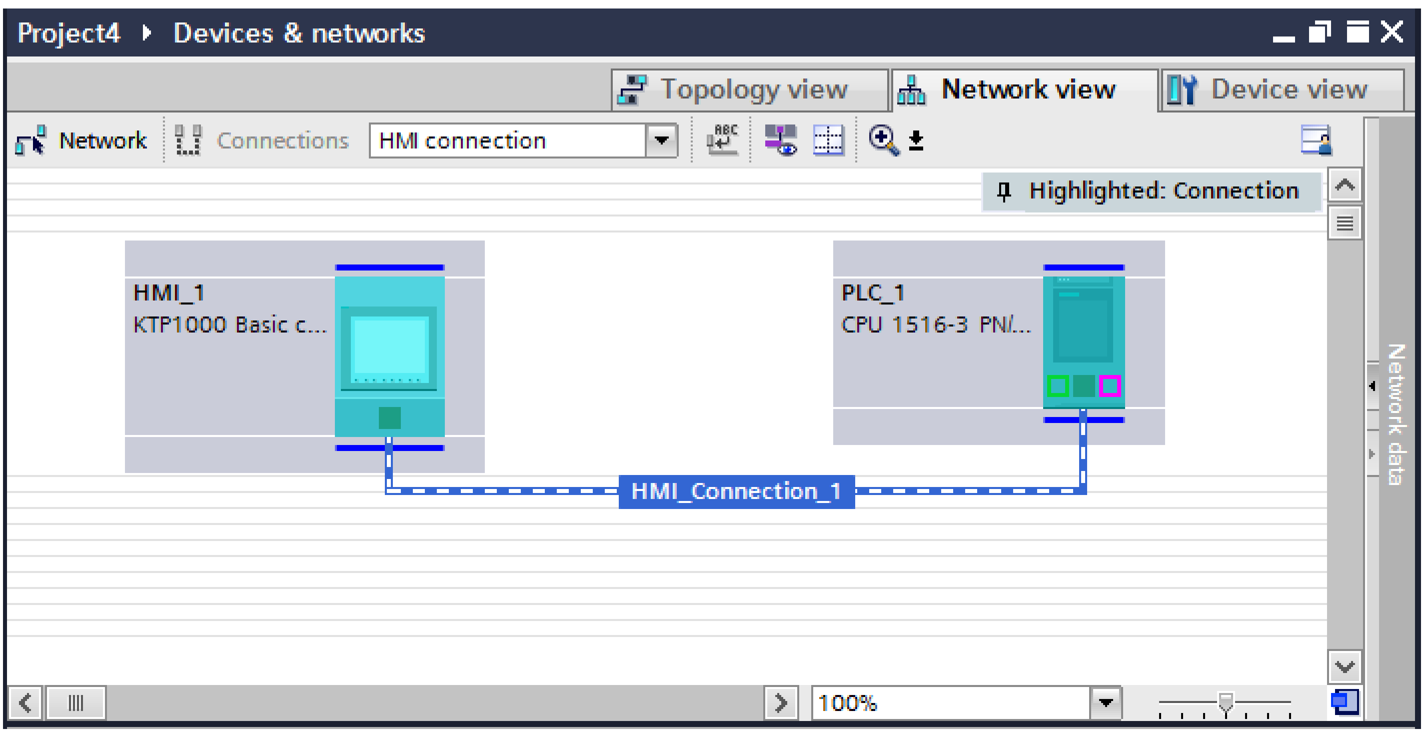
1.6.2 Communication via PROFINET (Basic Panels)

1.6.2.1 Configuring an HMI connection (Basic Panels)

Communication via PROFINET (Basic Panels)

HMI connections via PROFINET

If you have inserted an HMI device and a SIMATIC S7 1500 into the project, interconnect the two PROFINET interfaces in the "Devices & Networks" editor.



You can also connect multiple HMI devices to a single SIMATIC S7 1500 and multiple SIMATIC S7 1500 to a single HMI device.

The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used.

Additional information is available in the documentation for the respective HMI device.

HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFINET in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device via PROFINET in the "Connections" editor of the HMI device.

Configuring an HMI connection via PROFINET (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC S7 1500 via PROFINET or Ethernet in the "Devices & Networks" editor.

⚠ CAUTION
Communication via Ethernet
In Ethernet-based communication, the end user is responsible for the security of his data network.
Targeted attacks can overload the device and interfere with proper functioning.

Requirements

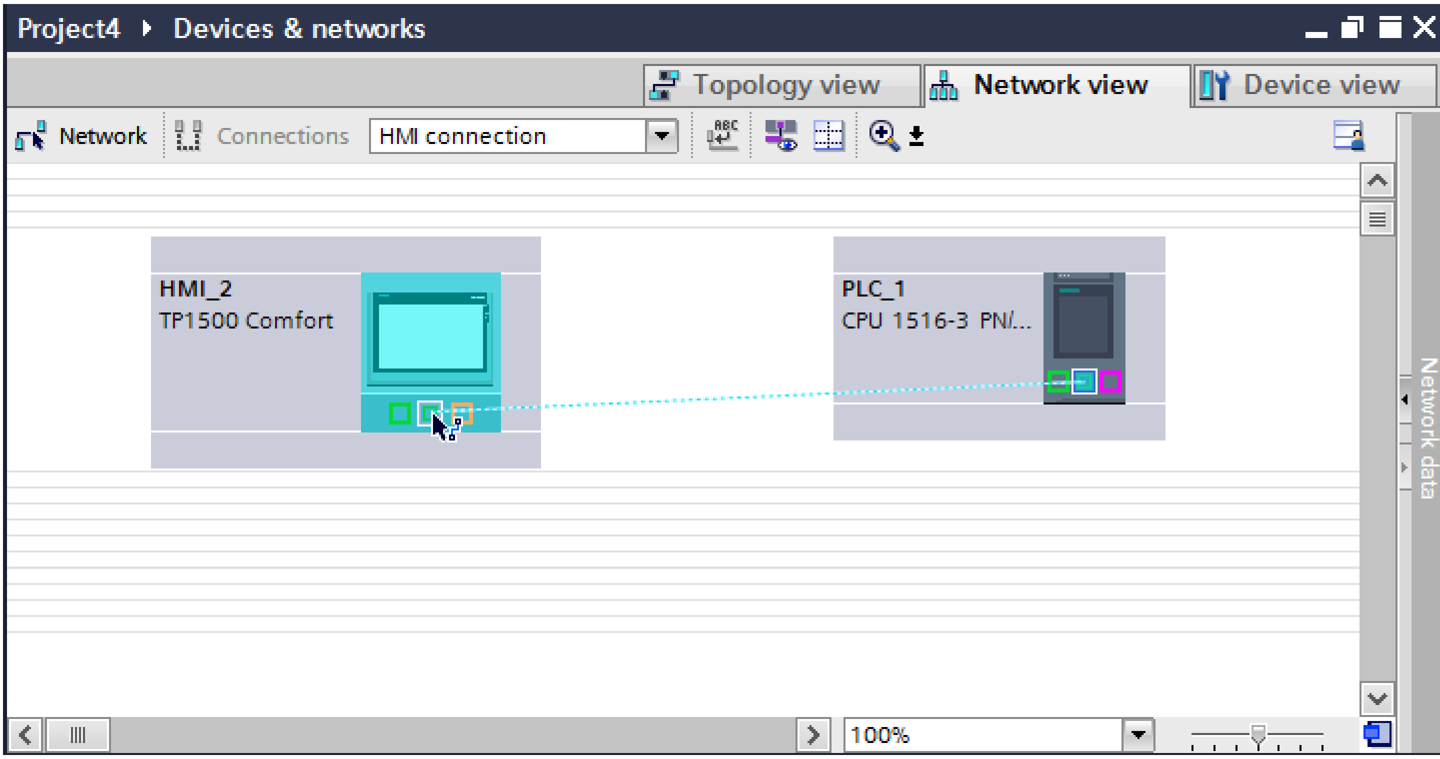
The following communication partners are created in the "Devices & Networks" editor:

- HMI device with PROFINET or Ethernet interface
- SIMATIC S7 1500 with PROFINET interface

Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button and select "HMI connection" for the connection type.
   The devices available for connection are highlighted in color.

3. Click the PROFINET interface of the PLC and use a drag-and-drop operation to draw a connection to the PROFINET or Ethernet interface of the HMI device.



4. Click the connecting line.
5. Click "Highlight HMI connection" and select the HMI connection.
   The connection is displayed graphically in the Inspector window.
6. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project. See the chapter "AUTOHOTSPOT" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created a connection between an HMI device and a SIMATIC S7 1500. The IP address and subnet mask connection parameters are configured.

1.6.2.2 PROFINET parameters (Basic Panels)

PROFINET parameters for the HMI connection (Basic Panels)

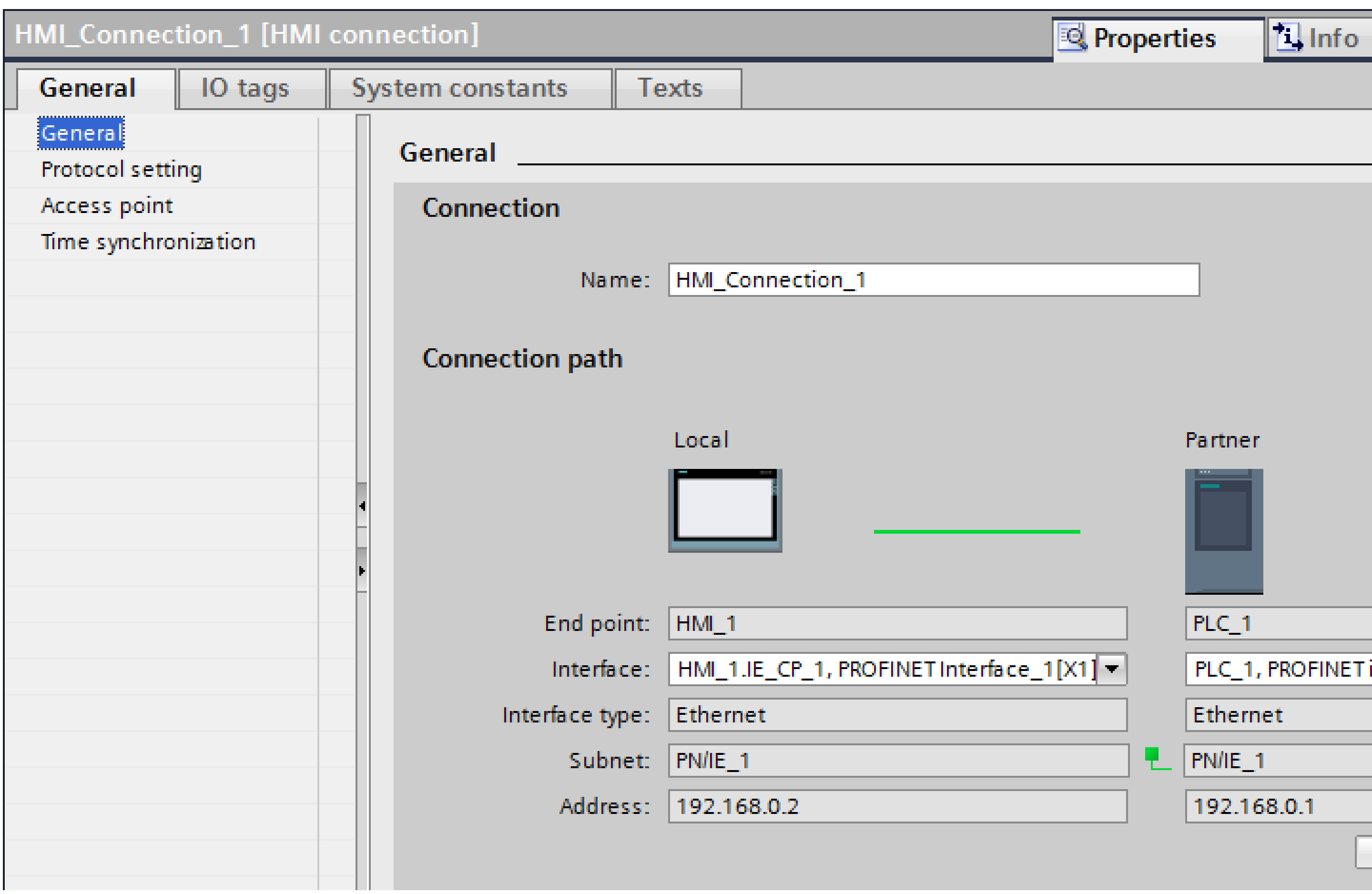
PROFINET parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and changing the HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

Shows the name of the HMI connection.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFINET parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area cannot be edited.
- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the selected IP address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

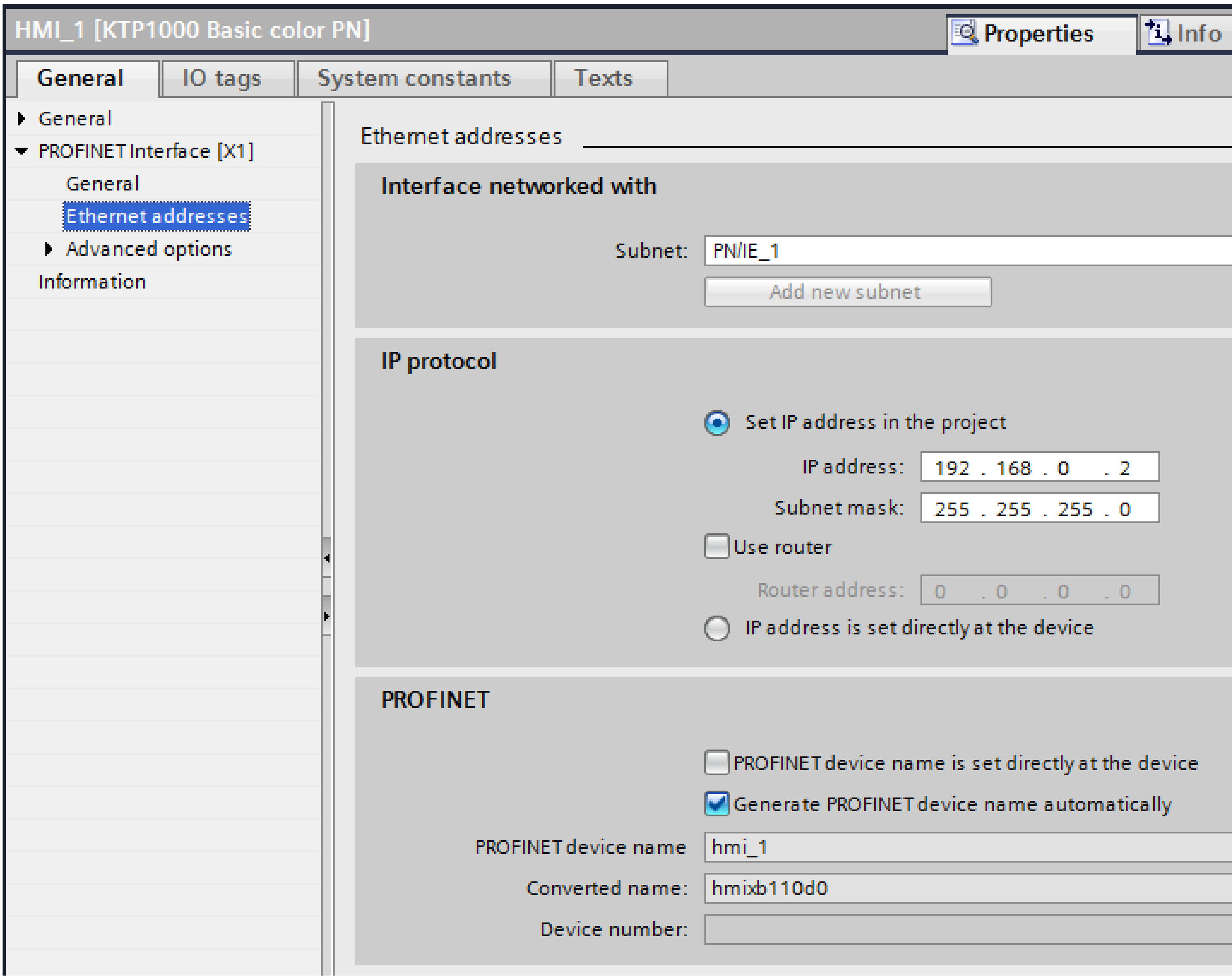
PROFINET parameters for the HMI device (Basic Panels)

PROFINET parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Set IP address in the project"
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

  Note

  The device is automatically restarted in the case of HMI devices with the Windows CE 3.0 operating system.

  HMI devices with Windows CE 3.0:

  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
- "Use IP router"
  If you are using an IP router, select "Use IP router" and enter the router address in the "Router address" field.
- "Set IP address using a different method"
  If the function "Set IP address using a different method" is activated, the IP address is not taken from the project. You have to enter the IP address directly in the Control Panel of the HMI device.

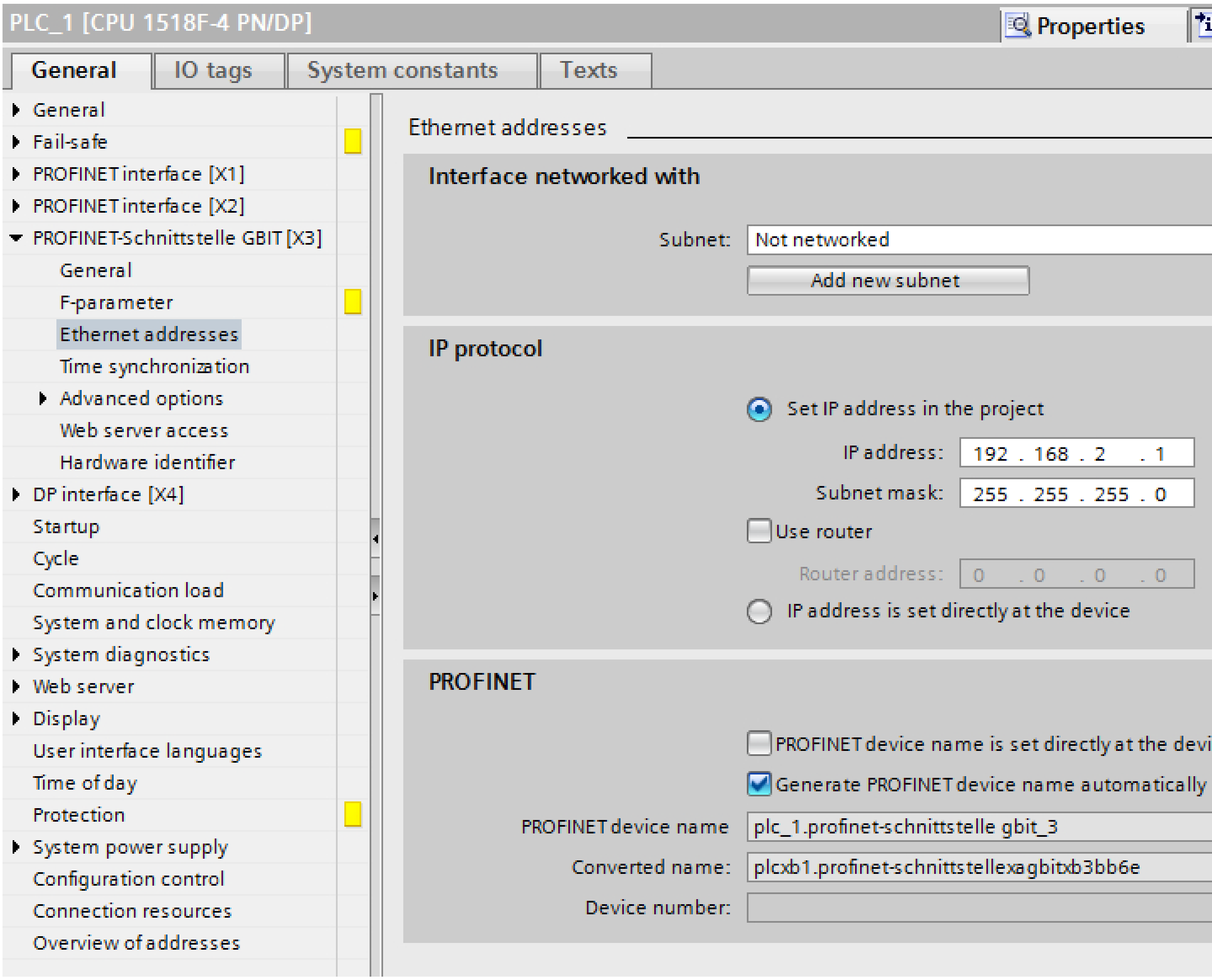
PROFINET parameters for the PLC (Basic Panels)

PROFINET parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFINET parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "IP address"
  You assign the IP address of the HMI device in the "IP address" area.
- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
  If you are using an IP router, select "Use IP router" and enter the router address in the field.

Protection of communication (Basic Panels)

Security levels (Basic Panels)

If you want to protect the controller and HMI device communication, you can assign protection levels for the communication.

For a SIMATIC S7-1500 CPU, you can enter multiple passwords and thereby set up different access rights for various user groups.

The passwords are entered in a table, so that exactly one protection level is assigned to each password.

The effect of the password is given in the "Protection" column.

For the SIMATIC S7-1500 controller, several aspects need to be considered when setting protection levels.

For additional information on this, see:Setting options for the protection (Page 77)

Example

When configuring the controller, you select the "Complete protection" protection level for a standard CPU (i.e., not an F-CPU).

Afterwards, you enter a separate password for every protection level above it in the table.

For users who do not know any of the passwords, the CPU is completely protected. Not even HMI access is possible.

For users who know one of the assigned passwords, the effect depends on the table row in which the password occurs:

- The password in row 1 (no protection) allows access as if the CPU were completely unprotected. Users who know this password have unrestricted access to the CPU.
- The password in row 2 (write protection) allows access as if the CPU were write-protected. Despite knowing the password, users who know this password only have read access to the CPU.
- The password in row 3 (read and write protection) allows access as if the CPU were read-protected and write-protected, so that only HMI access is possible for users who know this password.

Setting options for the protection (Basic Panels)

Access levels

The following section describes how to use the various access levels of the S7-1500 CPUs.

S7-1500 CPUs provide various access levels to limit the access to specific functions.

The individual access levels as well as their associated passwords are specified in the object properties of the CPU. You assign parameters for the access level in a table.

Access level	Access			Access permission
	HMI	Read	Write	Password
Full access (no protection)	✓	✓	✓	
Read access	✓	✓		
HMI access	✓			
No access (complete protection)				

The green checkmarks in the columns to the right of the respective access level specify which operations are possible without knowing the password of this access level.

If you want to use the functions of fields that are not selected in the "Access" column, a password has to be entered:

Example:

You set the access level "Read access". You can see from the table that write access is not possible during operation without entering a password.

The table also shows that full access is necessary for the "Write" function.

To use a function requiring write access during operation, the password for full access must therefore be entered.

> **NOTICE**
>
> **Configuring an access level does not replace know-how protection**
>
> Configuring access levels offers a high degree of protection against unauthorized changes to the CPU by restricting download privileges to the CPU. However, blocks on the memory card are not write- or read-protected. Use know-how protection to protect the code of blocks on the memory card.

Default characteristics

The default access level is "Full access (no protection)". Every user can read and change the hardware configuration and the blocks. A password is not set and is also not required for online access.

The access levels in detail

Below you can find an explanation of the existing access levels and which functions are possible at the respective access level.

- Full access (no protection):
  The hardware configuration and the blocks can be read and changed by all users.
- Read access for F-blocks (F-CPUs only):
  F-blocks of the safety program cannot be changed without authorization by the password of this access level or a higher level.
  Further information is available in the programming and operating manual *SIMATIC Safety - Configuring and Programming.*
- Read access:
  With this access level, read-only access to the hardware configuration and the blocks is possible without entering a password, which means you can download hardware configuration and blocks to the programming device. In addition, HMI access and access to diagnostics data is possible, as is changing the operating state (RUN/STOP).
  You cannot load blocks or a hardware configuration into the CPU. Moreover, no writing test functions and firmware updates are possible.
- HMI access:
  Only HMI access and access to diagnostics data is possible. Tags can be read and written via a HMI device.
  At this access level, you can neither load blocks and the hardware configuration into the CPU nor load blocks and the hardware configuration from the CPU into the programming device.
  In addition, the following is **not** possible: Writing test functions, changing the operating state (RUN/STOP) and firmware updates.
- No access (complete protection):
  Only identification data can be read, via "Accessible devices", for example.
  Neither read nor write access to the hardware configuration and the blocks is possible. HMI access is also not possible. The server function for PUT/GET communication is disabled in this access level (cannot be changed).
  Legitimation with a configured password provides you with access in accordance with the associated protection level.

Behavior of functions at different access levels

The table below describes which online functions are possible in the various protection levels.

Function	Full access	Read access	HMI access	No access
Identification of the device, via "Accessible devices", for example	Yes	Yes	Yes	Yes
HMI diagnostics view	Yes	Yes	Yes	No
Monitoring tags (M, I, Q, DB content) via HMI device	Yes	Yes	Yes	No
Modifying tags (M, I, Q, DB content) via HMI device	Yes	Yes	Yes	No

Diagnostics display (for example, device information, connection display, alarm display, diagnostic buffer)	Yes	Yes	Yes	No
Reading cycle time statistics (Online & Diagnostics)	Yes	Yes	Yes	No
Reading information from the hardware configuration (Online & Diagnostics)	Yes	Yes	Yes	No
Reading time-of-day	Yes	Yes	Yes	No
Executing online functions within the hardware configuration (Online & Diagnostics)	Yes	Yes	Yes	No
Acknowledging alarms	Yes	Yes	Yes	No
Receiving alarms	Yes	Yes	Yes	No
Enabling/disabling alarms	Yes	Yes	No	No
Reading tags via test function (STEP 7, tag table or watch table)	Yes	Yes	No	No
Requesting operating state change online (RUN/STOP/ warm restart)	Yes	Yes	No	No
Downloading data blocks, code blocks, hardware configuration to PG/PC	Yes	Yes	No	No
Set time-of-day	Yes	Yes	No	No
Deleting data blocks, code blocks, hardware configuration in the CPU	Yes	No	No	No
Downloading individual data blocks, code blocks, hardware configurations to the CPU	Yes	No	No	No
Loading PLC program to the device and resetting	Yes	No	No	No
Firmware update of CPUs or I/O modules	Yes	No	No	No
Modifying tags via test function (STEP 7, watch table)	Yes	No	No	No
Reading tags in the program status	Yes	No	No	No
Online editing of blocks	Yes	No	No	No
Modifying outputs in STOP mode	Yes	No	No	No

Behavior of a password-protected module during operation

The CPU protection takes effect after the settings are downloaded to the CPU.

Before an online function is executed, the necessary permission is checked and, if necessary, the user is prompted to enter a password.

Example: The module was configured with read access and you want to execute the "Modify tags" function. This requires write access to a test function; therefore, the assigned password must be entered to execute the function.

The functions protected by a password can only be executed by one programming device/PC at any one time. Another programming device/PC cannot log on.

Access authorization to the protected data is in effect for the duration of the online connection or until the access authorization is manually rescinded with "Online > Delete access rights".

Each access level allows unrestricted access to certain functions without entering a password, for example, identification using the "Accessible devices" function.

Access to a password-protected S7-1500 CPU can be restricted locally on the display. The restriction is only in effect when the operating mode switch is set to RUN.

Access password for the HMI connection (Basic Panels)

Introduction

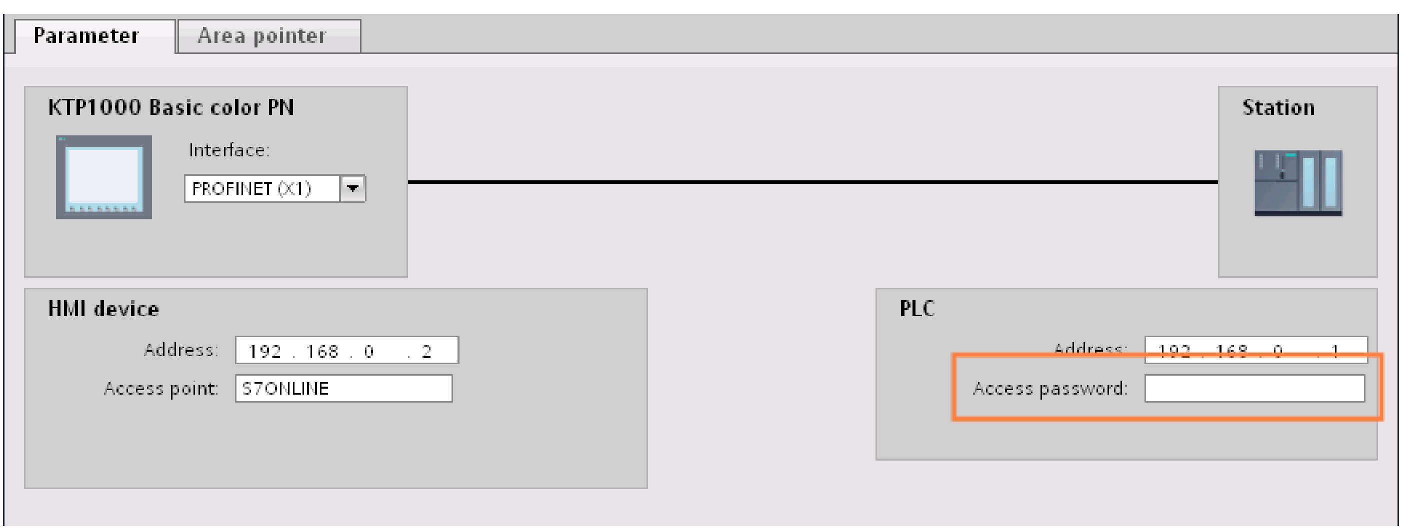
You secure access to a PLC by assigning a password.

You assign the password when you configure the password.

A password is mandatory as of protection level "Complete protection". Communication to the PLC is denied if an incorrect password or no password is entered.

Entering access password

You enter the password for the PLC in the "Connections" editor.



1.6.2.3 Setting port options (Basic Panels)

Setting the port options (Basic Panels)

Changing connection settings for the PROFINET IO port

You can change the network settings for the PROFINET IO port as required. By default, the settings are made automatically. In normal situations, this guarantees problem-free communication.

Possible settings for transmission rate / duplex

Depending on the selected device, you can make the following settings for "Transmission rate / duplex":

- Automatic setting
  Recommended default setting of the port. The transmission settings are automatically "negotiated" with the peer port. The "Enable autonegotiation" option is also enabled as a default, in other words, you can use cross cables or patch cables for the connection.
- TP/ITP at x Mbps full duplex (half duplex)
  Setting of the transmission rate and the full duplex/half duplex mode. The effectiveness depends on the "Enable autonegotiation" setting:
  - Autonegotiation enabled
    You can use both cross cable and patch cable.
  - Autonegotiation disabled
    Make sure that you use the correct cable (patch cable or cross cable)! The port is also monitored with this setting.
- Deactivated
  Depending on the module type, the drop down list box can contain the "- Disabled -" option. This option, for example, allows you to prevent access to an unused port for security reasons. With this setting, diagnostic events are not generated.

"Monitor" option

This option enables or disables port diagnostics. Examples of port diagnostics: The link status is monitored, in other words, the diagnostics are generated during link-down and the system reserve is monitored in the case of fiber optic ports.

Option "Enable autonegotiation "

The autonegotiation setting can only be changed if a concrete medium (for example, TP 100 Mbps full duplex) is selected. Whether or not a concrete medium can be set depends on the properties of the module.

If autonegotiation is disabled, this causes the port to be permanently specified, as for example, is necessary for a prioritized startup of the IO device.

You must make sure the partner port has the same settings because with this option the operating parameters of the connected network are not detected and the data transmission rate and transmission mode can accordingly not be optimally set.

Note

When a local port is connected, STEP 7 makes the setting for the partner port if the partner port supports the setting. If the partner port does not accept the setting, an error message is generated.

Wiring rules for disabled autonegotiation (Basic Panels)

Requirements

You have made the following settings for the port in question, for example, to accelerate the startup time of the IO device:

- Fixed transmission rate
- Autonegotiation incl. autocrossing disabled

The time for negotiating the transmission rate during startup has been saved.

If you have disabled autonegotiation, you must observe the wiring rules.

Wiring rules for disabled autonegotiation

PROFINET devices have the following two types of ports:

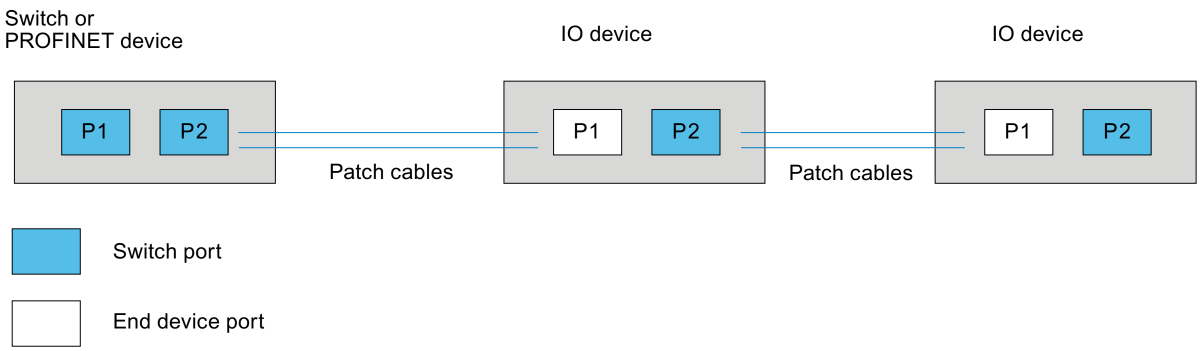
Type of port	PROFINET devices	Note
Switch port with crossed pin assignment	For IO devices: Port 2 For S7 CPUs with 2 ports: Ports 1 and 2	Crossed pin assignment means that the pin assignment for the ports for sending and receiving between the respective PROFINET devices is exchanged internally.
End device port with uncrossed pin assignment	For IO devices: Port 1 For S7 CPUs with one port: Port 1	-

Validity of the wiring rules

The cabling rules described in the following paragraph apply exclusively for the situation in which you have specified a fixed port setting.

Rules for cabling

You can connect several IO devices in line using a single cable type (patch cable). To do this, you connect port 2 of the IO device (distributed I/O) with port 1 of the next IO device. The following graphic gives an example with two IO devices.



Boundaries at the port (Basic Panels)

Requirements

To use boundaries, the respective device must have more than one port. If the PROFINET does not support boundary settings, they are not shown.

Enable boundaries

"Boundaries" are limits for transmission of certain Ethernet frames. The following boundaries can be set at a port:

- "End of discovery of accessible devices"
  No forwarding of DCP frames to identify accessible devices. Devices downstream from this port cannot be reached by the project tree under "Accessible devices". Devices downstream from this port cannot be reached by the CPU.
- "End of topology discovery"
  LLDP frames (Link Layer Discovery Protocol) are not forwarded for topology detection.
- "End of sync domain"
  No forwarding of sync frames transmitted to synchronize nodes within a sync domain.
  If you operate, for example, a PROFINET device with more than two ports in a ring, you should prevent the sync frame from being fed into the ring by setting a sync boundary (at the ports not inside the ring).
  Additional example: If you want to use several sync domains, configure a sync domain boundary for the port connected to a PROFINET device from the other sync domain.

Restrictions

The following restrictions must be observed:

- The individual check boxes can only be used if the port supports the function in question.
- If a partner port has been determined for the port, the following check boxes cannot be used:
  - "End of discovery of accessible devices"
  - "End of topology discovery"
- If autonegotiation is disabled, none of the check boxes can be used.

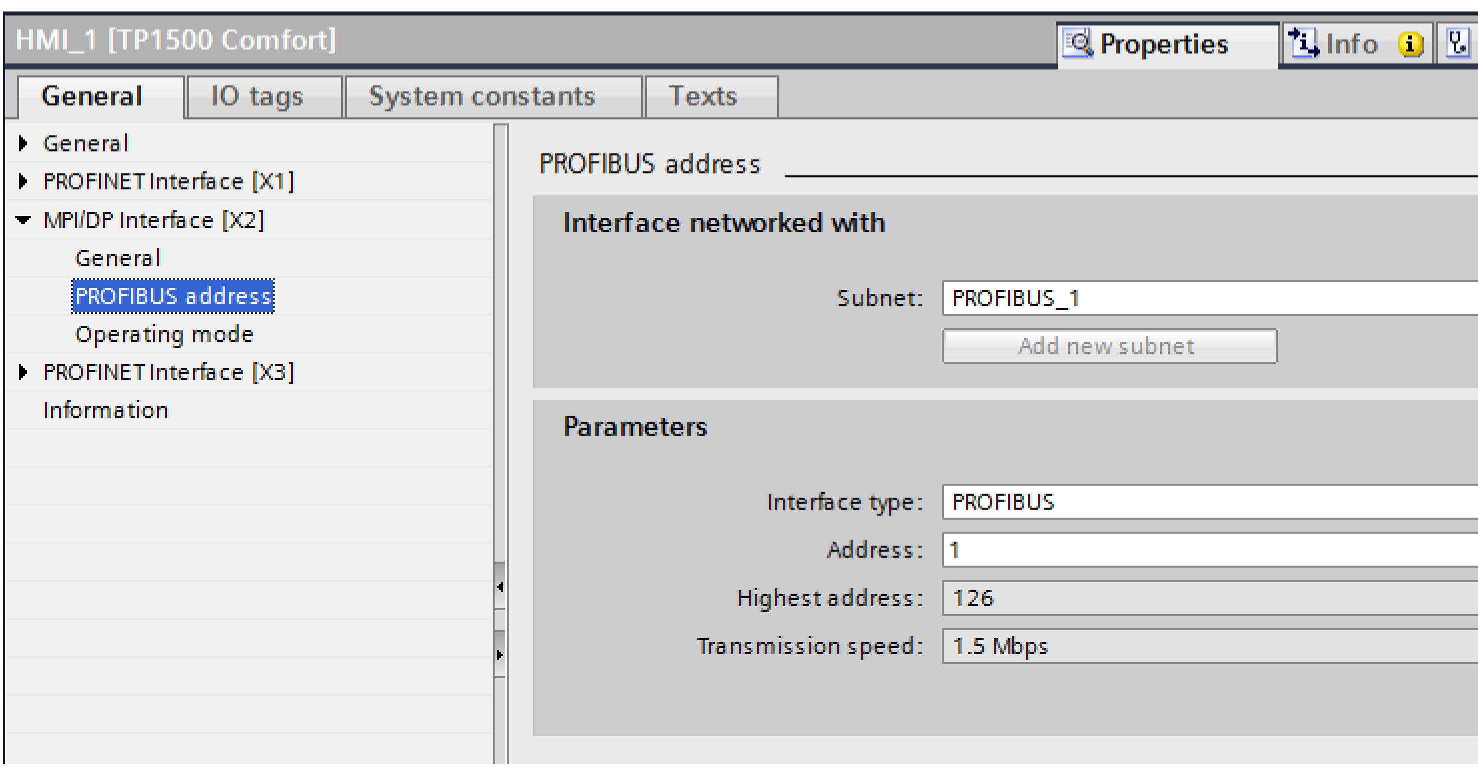
1.6.3 Communication via PROFIBUS (Basic Panels)

1.6.3.1 Configuring an HMI connection (Basic Panels)

Communication via PROFIBUS (Basic Panels)

HMI connections via PROFIBUS

If you have inserted an HMI device and a SIMATIC S7 1500 into the project, interconnect the two PROFIBUS interfaces in the "Devices & Networks" editor.



HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFIBUS in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device in the "Connections" editor of the HMI device.

Configuring an HMI connection via PROFIBUS (Basic Panels)

Introduction

You configure an HMI connection over PROFIBUS between HMI devices and a SIMATIC S7 1500 in the "Devices & Networks" editor.

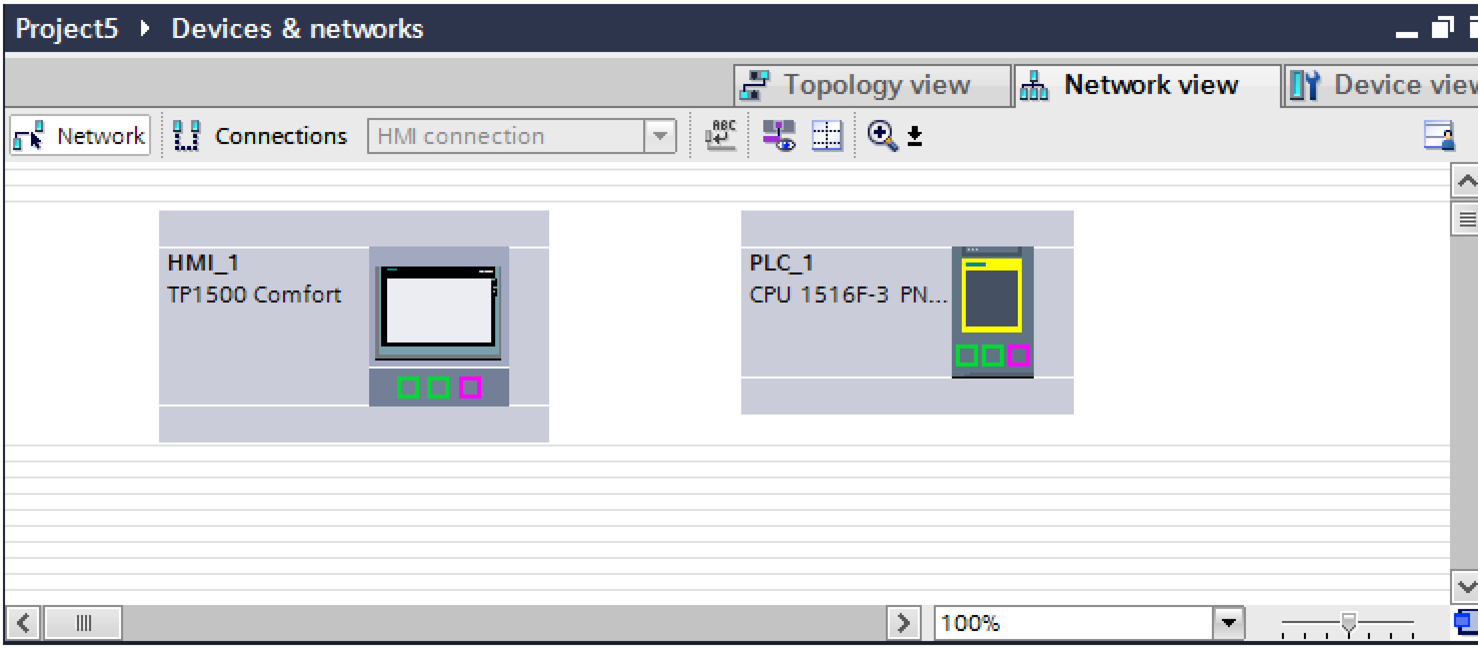
Requirements

The following communication partners are created in the "Devices & Networks" editor:

- HMI device with MPI/DP interface
- SIMATIC S7 1500 with PROFIBUS interface

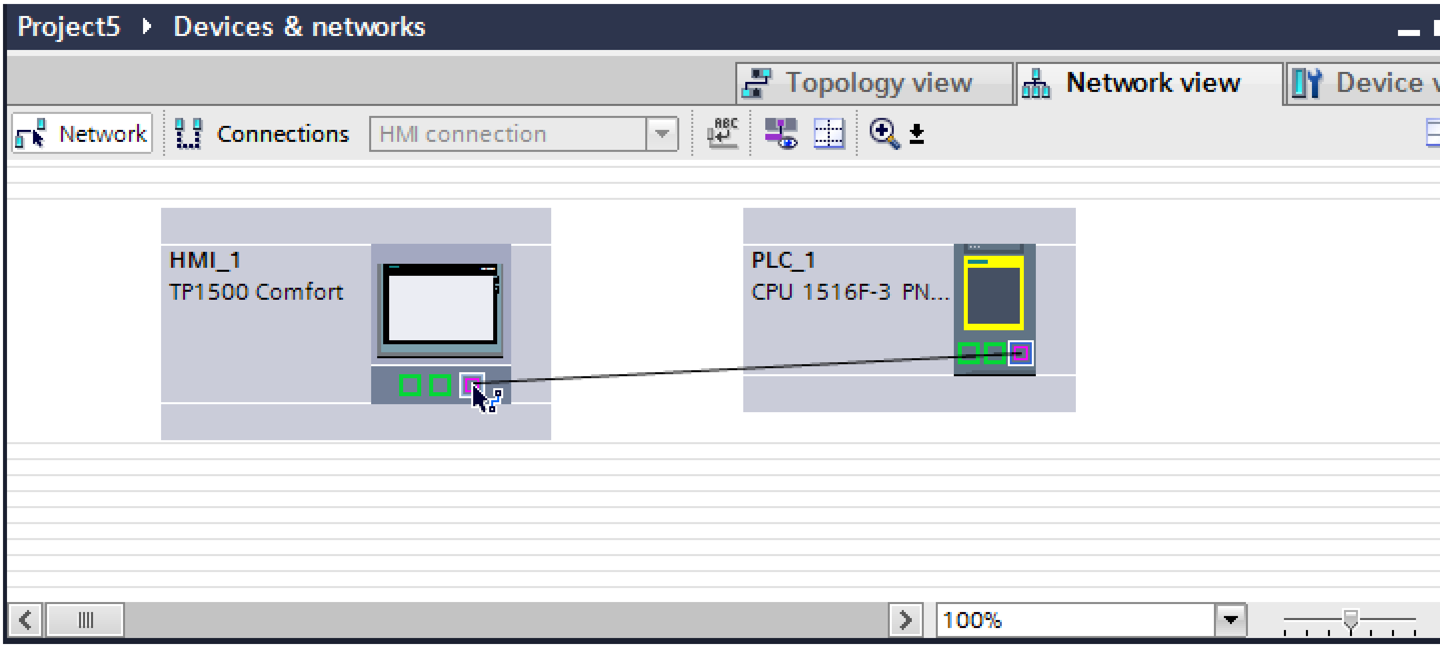
Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.



2. Click the "Connections" button.
   The devices available for connection are highlighted in color.
3. Click the HMI device interface.
4. Select the "PROFIBUS" interface type in the Inspector window under "Properties > General > HMI MPIDP > Parameters".

5. Click the interface of the PLC and use a drag-and-drop operation to draw a connection to the HMI device.



6. Click the connecting line.
7. Click "Highlight HMI connection" and select the HMI connection.
   The connection is displayed graphically in the Inspector window.
8. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project.
   See the chapter "AUTOHOTSPOT" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created an HMI connection over PROFIBUS between an HMI device and a SIMATIC S7 1500.

1.6.3.2 PROFIBUS parameters (Basic Panels)

PROFIBUS parameters for the HMI connection (Basic Panels)

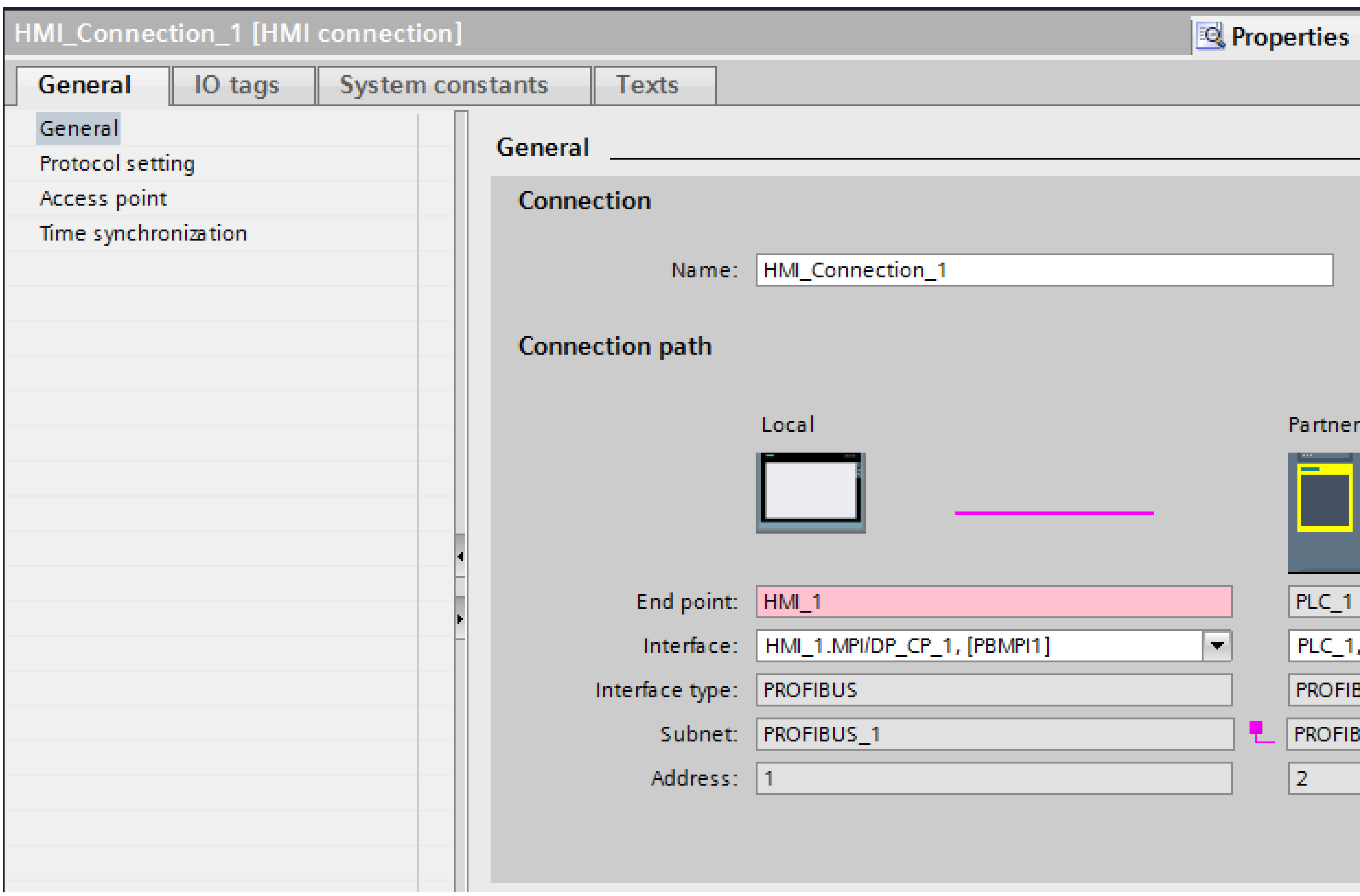
PROFIBUS parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and changing the HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

Displays whether the devices are networked together.

 - displayed if the devices are networked together.

 - displayed if the devices are not networked together.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFIBUS parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area cannot be edited.
- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the PROFIBUS address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

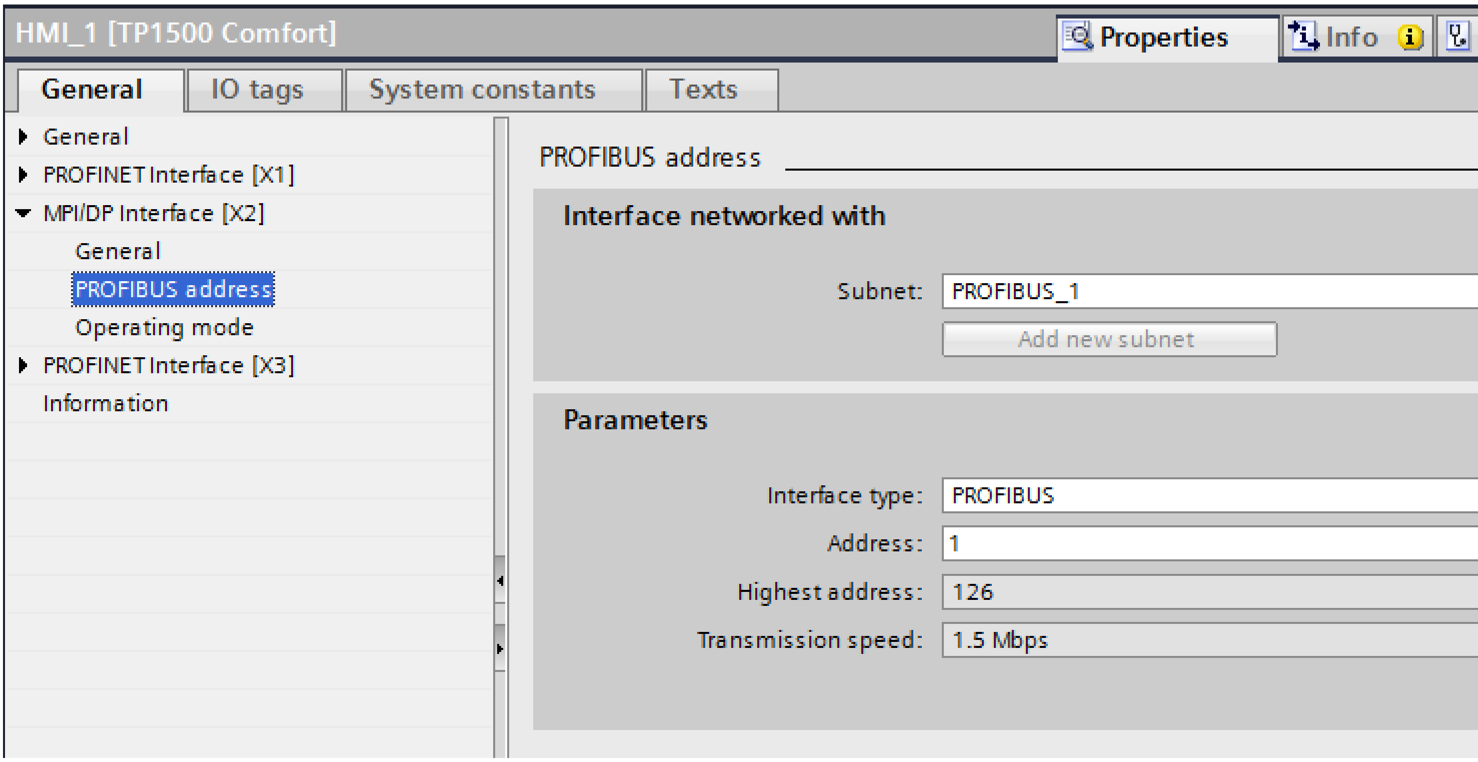
PROFIBUS parameters for the HMI device (Basic Panels)

PROFIBUS parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  You assign the interface type in the "Interface type" area. Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.
- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

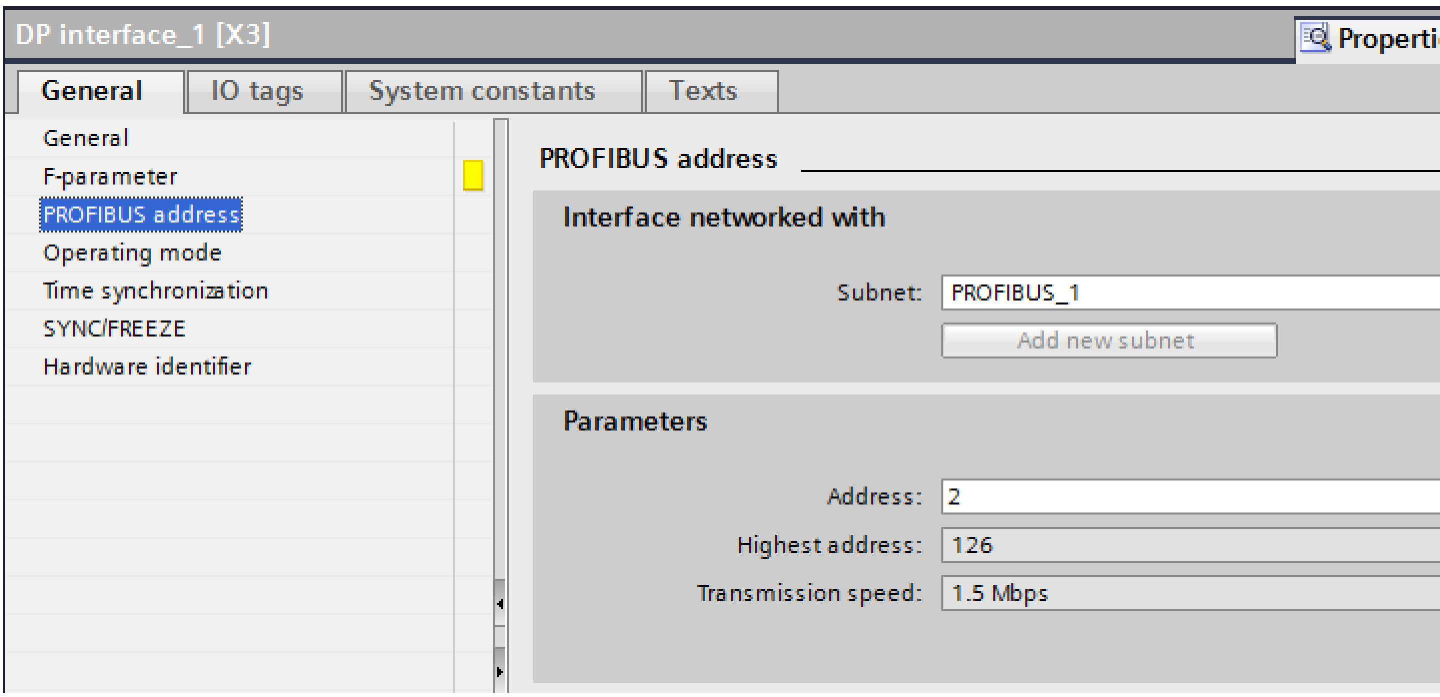
PROFIBUS parameters for the PLC (Basic Panels)

PROFIBUS parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFIBUS parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.

- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

Protection of communication (Basic Panels)

Access password for the HMI connection (Basic Panels)

Introduction

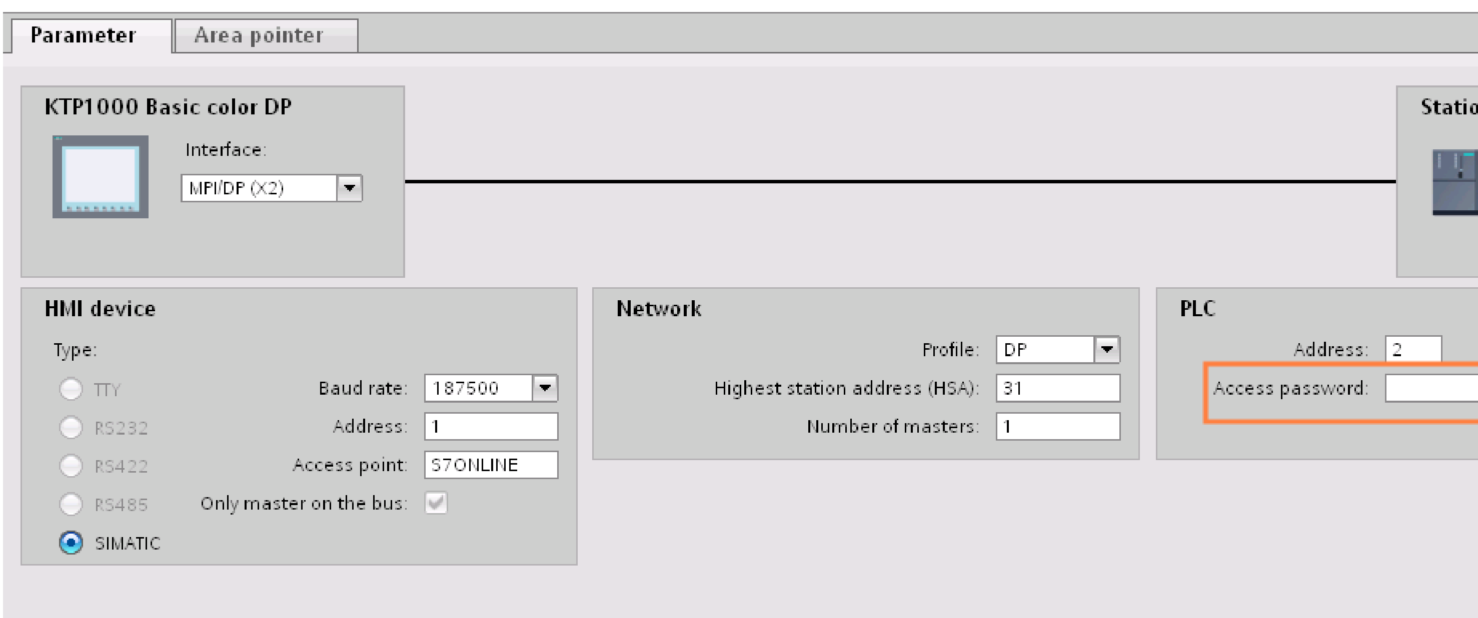
You secure access to a PLC by assigning a password.

You assign the password when you configure the password.

A password is mandatory as of protection level "Complete protection". Communication cannot be established with the PLC if you enter no password or an incorrect password.

Assigning password

You enter the password for the PLC in the "Connections" editor.



Security levels (Basic Panels)

If you want to protect the PLC and HMI device communication, you can assign protection levels for the communication.

For a SIMATIC S7-1500 CPU, you can enter multiple passwords and thereby set up different access rights for various user groups.

The passwords are entered in a table, so that exactly one protection level is assigned to each password.

The effect of the password is given in the "Protection" column.

For the SIMATIC S7-1200 controller, several aspects need to be considered when setting protection levels. For additional information on this, see: AUTOHOTSPOT

Example

You select the "Complete protection" protection level for a standard CPU (i.e., not an F-CPU) when configuring it.

Afterwards, you enter a separate password for every protection level above it in the table.

For users who do not know any of the passwords, the CPU is completely protected. Not even HMI access is possible.

For users who know one of the assigned passwords, the effect depends on the table row in which the password occurs:

- The password in row 1 (no protection) allows access as if the CPU were completely unprotected. Users who know this password have unrestricted access to the CPU.
- The password in row 2 (write protection) allows access as if the CPU were write-protected. Despite knowing the password, users who know this password only have read access to the CPU.
- The password in row 3 (read and write protection) allows access as if the CPU were read-protected and write-protected, so that only HMI access is possible for users who know this password.

1.6.4 Data exchange (Basic Panels)

1.6.4.1 Area pointers (Basic Panels)

General information on area pointers (Basic Panels)

Introduction

You use an area pointer to access a data area in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations.

The PLC and the HMI device trigger defined interactions based on the evaluation of stored data.

Configuration of area pointers

Before you use an area pointer, you enable it under "Connections > Area pointer". You then assign the area pointer parameters.

You can find more detailed information on configuring area pointers in:

Data exchange using area pointers (Page 54)

Restrictions

You can only configure the following data types for communication with SIMATIC S7 1500 for data exchange using area pointers:

- UInt and array of UInt
- Word and array of Word
- Int and array of Int
- "Array[0..15] of Bool" for area pointer "Coordination"
- Date_And_Time
- DTL and LDT

"Screen number" area pointer (Basic Panels)

Function

The HMI device saves information about the screen called on the HMI device to the "Screen number" area pointer.

This allows the transfer of the current screen contents from the HMI device to the PLC. The PLC can trigger specific reactions such as the call of a different screen.

Use

Configure and enable the area pointer in "Communication > Connections" before you put it into use. You can create only **one** instance of the "Screen number" area pointer and only on **one** PLC.

The screen number is always transferred to the PLC when a new screen is activated or when the focus within a screen changes from one screen object to another.

Structure

The area pointer is a data area in the memory of the PLC with a fixed length of 5 words.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1st word	Current screen type															
2nd word	Current screen number															

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
3rd word	Reserved															
4th word	Current field number															
5th word	Reserved															

- Current screen type
  "1" for root screen or
  "4" for permanent window
- Current screen number
  1 to 32767
- Current field number
  1 to 32767

"Date/time" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the HMI device to the PLC.

The PLC writes control job "41" or "40" to the job mailbox.

When it evaluates the control job, the HMI device writes its current date and the time in the data area configured in the "Date/time" area pointer. All definitions are coded in BCD format.

Note

You cannot use the "Date/Time PLC" area pointer if you have configured the "Date/Time" area pointer.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

The date/time data area has the following structure:

Data word	Most significant byte	Least significant byte	
	7 ... 0	7 ... 0	
n+0	Reserved	Hour (0 to 23)	Time
n+1	Minute (0 to 59)	Second (0 to 59)	
n+2	Reserved	Reserved	
n+3	Reserved	Weekday (1 to 7, 1=Sunday)	Date
n+4	Day (1 to 31)	Month (1 to 12)	
n+5	Year (80 to 99/0 to 29)	Reserved	

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

Using data types

The data types "Date_And_Time, DTL" and "LDT" can only be used with the "Date/time" and "Date/time PLC" area pointers.

The data format of the "Date/time" area pointer depends on job mailbox 40/41.

If there are no control tags linked to the area pointer, or a control tag is linked with the data type "Array[0..5] of UInt/Word/Int", the following applies:

The configuration of the "Date/time" area pointer is only used for job mailbox 41.

If job mailbox 40 is used, the data format "DATE_AND_TIME (BCD-encoded)" is used (shown in the next section).

If the "Date/time" and "Date/time PLC" area pointers are linked to a control tag with the data type "DATE_AND_TIME", "DTL" or "LDT", the associated data format is used in the corresponding area pointer.

"Date/time PLC" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the PLC to the HMI device. Use this area pointer if the PLC is the time master.

The PLC loads the data area of the area pointer. All definitions are coded in BCD format.

The HMI device reads the data cyclically within the configured acquisition cycle and synchronizes itself.

Note

Set an acquisition cycle of sufficient length for the Date/time area pointer in order to avoid any negative impact on HMI device performance.
Recommended: Acquisition cycle of 1 minute, if the process allows this.

"Date/Time PLC" is a global area pointer and may be configured only once per project.

Note

You cannot use the "Date/Time" area pointer if you have configured the "Date/Time PLC" area pointer.

The date/time data area has the following structure:

DATE_AND_TIME format (in BCD code)

<table>
<tr><th rowspan="2">Data word</th><th colspan="3">Most significant byte</th><th colspan="3">Least significant byte</th></tr>
<tr><th>7</th><th>. . . . . .</th><th>0</th><th>7</th><th>. . . . . .</th><th>0</th></tr>
<tr><td>n+0</td><td colspan="3">Year (80 to 99/0 to 29)</td><td colspan="3">Month (1 to 12)</td></tr>
<tr><td>n+1</td><td colspan="3">Day (1 to 31)</td><td colspan="3">Hour (0 to 23)</td></tr>
<tr><td>n+2</td><td colspan="3">Minute (0 to 59)</td><td colspan="3">Second (0 to 59)</td></tr>
<tr><td>n+3</td><td colspan="3">Reserved</td><td colspan="2">Reserved</td><td>Weekday (1 to 7, 1=Sunday)</td></tr>
<tr><td>n+4 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
<tr><td>n+5 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
</table>

1) The two data words must exist in the data area to ensure that the data format matches WinCC flexible and to avoid reading false information.

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

Using data types

The data types "Date_And_Time, DTL" and "LDT" can only be used with the "Date/time" and "Date/time PLC" area pointers.

The data format of the "Date/time" area pointer depends on job mailbox 40/41.

If there are no control tags linked to the area pointer, or a control tag is linked with the data type "Array[0..5] of UInt/Word/Int", the following applies:

The configuration of the "Date/time" area pointer is only used for job mailbox 41.

If job mailbox 40 is used, the data format "DATE_AND_TIME (BCD-encoded)" is used (shown in the next section).

If the "Date/time" and "Date/time PLC" area pointers are linked to a control tag with the data type "DATE_AND_TIME", "DTL" or "LDT", the associated data format is used in the corresponding area pointer.

"Coordination" area pointer (Basic Panels)

Function

The "Coordination" area pointer is used to implement the following functions:

- Detecting the startup of the HMI device in the control program
- Detecting the current operating mode of the HMI device in the control program
- Detecting whether the HMI device is ready to communicate in the control program

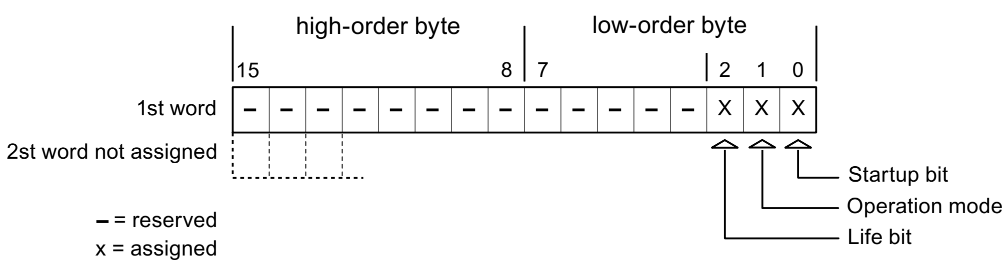
By default, the "Coordination" area pointer has the length of one word and cannot be changed.

Usage

Note

The HMI device always writes the entire coordination area when updating the area pointer. The control program can for this reason not make changes to the coordination area.

Assignment of the bits in the "Coordination" area pointer



Startup bit

The startup bit is set briefly to "0" by the HMI device during startup. It sets the bit permanently to "1" when startup is completed.

Operating mode

The operating mode bit is set to 1 as soon as the user switches the HMI device offline. The status of the operating mode bit is "0" during normal operation of the HMI device. You can determine the current operating mode of the HMI device by reading this bit in the control program.

Life bit

The HMI device inverts the life bit at intervals of approximately one second. You can check whether or not there is still a connection to the HMI device by querying this bit in the control program.

"Project ID" area pointer (Basic Panels)

Function

When Runtime starts, a check can be carried out as to whether the HMI device is connected to the correct PLC. This check is important when operating with several HMI devices.

For this, the HMI device compares a value stored on the PLC with the value specified in configuration. This ensures compatibility of the configuration data with the PLC program. If there is no concordance, a system event is given on the HMI device and Runtime is stopped.

Use

Note

HMI connections cannot be switched "online".

The HMI connection in which the "Project ID" area pointer is used must be switched "online".

To use this area pointer, set up the following during the configuration:

- Define the version of the configuration. Values between 1 and 255 are possible.
  You enter the version in the editor "Runtime settings > General" in the "Identification" area.
- Data address of the value for the version that is stored in the PLC:
  You enter the data address in the editor "Communication > Connections" under "Address".

Connection failure

A connection failure to a device on which the "project ID" area pointer is configured results in all the other connections of the device being switched to "offline".

This behavior has the following requirements:

- You have configured several connections in a project.
- You are using the "project ID" area pointer in at least one connection.

Causes which may set connections "offline":

- The PLC is not available.
- The connection has been switched offline in the engineering system.

"Job mailbox" area pointer (Basic Panels)

Function

The PLC can use the job mailbox to transfer jobs to the HMI device to trigger corresponding actions on the HMI device. These functions include, for example:

- Display screen
- Set date and time

Data structure

The first word of the job mailbox contains the job number. Depending on the job mailbox, up to three parameters can be transferred.

Word	Most significant byte	Least significant byte
n+0	0	Job number
n+1	Parameter 1	
n+2	Parameter 2	
n+3	Parameter 3	

The HMI device evaluates the job mailbox if the first word of this job is not equal to zero. This means that the parameters must be entered in the job mailbox first, followed by the job number.

When the HMI device accepts the job mailbox, the first word is set to 0 again. The execution of the job mailbox is generally not completed at this point in time.

Job mailboxes

All job mailboxes and their parameters are listed below. The "No." column contains the job number of the job mailbox. Job mailboxes can only be triggered by the PLC when the HMI device is online.

No.	Function	
14	**Set time (BCD-coded)**	
	Parameter 1	Left byte: - Right byte: hours (0-23)
	Parameter 2	Left byte: minutes (0-59) Right byte: seconds (0-59)
	Parameter 3	-
15	**Setting the date (BCD coded)** 3) 4)	
	Parameter 1	Left byte: - Right byte: weekday (1-7: Sunday-Saturday)
	Parameter 2	Left byte: day (1-31) Right byte: month (1-12)
	Parameter 3	Left byte: year
23	**User logon**	

No.	Function	
14	**Set time (BCD-coded)**	
	Logs the user on with the name "PLC user" at the HMI device with the group number transferred in Parameter 1. The logon is possible only when the transferred group number exists in the project.	
	Parameter 1	Group number 1 to 255
	Parameter 2, 3	-
24	**User logoff**	
	Logs off the current user. (The function corresponds to the "logoff" system function)	
	Parameter 1, 2, 3	-
40	**Transfer date/time to PLC**	
	(in the S7 format DATE_AND_TIME) An interval of at least 5 seconds must be maintained between two successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
41	**Transfer date/time to PLC**	
	(In OP/MP format) An interval of at least 5 seconds must be maintained between successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
46	**Update tag**	
	Causes the HMI device to read the current value of the tags from the PLC whose update ID matches the value transferred in Parameter 1. (Function corresponds to the "UpdateTag" system function.)	
	Parameter 1	1 - 100
49	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Warnings" class from the alarm buffer.	
	Parameter 1, 2, 3	-
50	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Errors" class from the alarm buffer.	
	Parameter 1, 2, 3	-
51	**Screen selection** 2)	
	Parameter 1	Screen number
	Parameter 2	-
	Parameter 3	Field number
69	**Reading data record from PLC** 1)	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	0: Do not overwrite existing data record 1: Overwrite existing data record
70	**Writing data record to PLC** 1)	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	-

1)	Only for devices supporting recipes.
2)	OP 73, OP 77A and TP 177A HMI devices also execute the "Screen selection" job mailbox if the on-screen keyboard is active.
3)	The weekday is ignored on HMI device KTP 600 BASIC PN.
4)	The weekday is ignored when you configure the "Date/Time PLC" area pointer.

"Data record" area pointer (Basic Panels)

"Data record" area pointer (Basic Panels)

Function

When data records are transferred between the HMI device and PLC, both partners access common communications areas on the PLC.

Data transfer types

There are two ways of transferring data records between the HMI device and PLC:

- Transfer without synchronization
- Transfer with synchronization via the data mailbox

Data records are always transferred directly, which means that the tag values are read straight from an address or written straight to an address configured for this tag without being redirected via an interim memory.

Initiating the transfer of data records

There are three ways of triggering the transfer:

- Operator input in the recipe view
- Job mailboxes
  The transfer of data records can also be triggered by the PLC.
- Triggering by configured functions

If the transfer of data records is triggered by a configured function or by a job mailbox, the recipe view on the HMI device remains operable. The data records are transferred in the background.

Simultaneous processing of several transfer requests is, however, not possible. In this case, the HMI device rejects the other transfer requests with a system alarm.

Transfer without synchronization (Basic Panels)

If you select asynchronous transfer of data records between the HMI device and PLC, there is no coordination over the common data areas. It is therefore unnecessary to set up a data area during configuration.

Asynchronous data record transfer can be a useful alternative, for example when:

- The system is capable of excluding the risk of uncontrolled overwriting of data by the communication peer.
- The PLC does not require information about the recipe number and data record number.
- The transfer of data records is triggered by the operator of the HMI device.

Reading values

When a read job is triggered, the values are read from the PLC addresses and transferred to the HMI device.

- Triggering by the operator in the recipe view:
  The values are downloaded to the HMI device. You can then, for example, process, edit, or save these values in the HMI device.
- Triggering by a function or job mailbox:
  The values are saved immediately to the data volume.

Writing values

When a write job is triggered, the values are written to the PLC addresses.

- Triggering by the operator in the recipe view:
  The current values are written to the PLC.
- Triggering by a function or job mailbox:
  The current values are written to the PLC from the data medium.

Transfer with synchronization (Basic Panels)

If you select synchronous transfer, both communication partners set status bits in the common data area. You can use this mechanism to prevent uncontrolled overwriting of data in either direction in your control program.

Application

Synchronous data record transfer can be a useful solution, for example, when:

- The PLC is the "active partner" in the transfer of data records.
- The PLC evaluates the information about the recipe number and data record number.
- The transfer of data records is triggered by means of a Job mailbox.

Requirements

In order to synchronize transfer of data records between the HMI device and the PLC, the following requirements must be met during configuration:

- An area pointer has been set up: "Communication > Connections" editor in "Area pointer".
- The PLC with which the HMI device synchronizes transfer of data records is specified in the recipe:
  "Recipes" editor in the inspector window the option "Coordinated transfer of data records" under "General > Synchronization > Settings"

Structure of the data area

The data area has a fixed length of 5 words. Structure of the data area:

	15		0
1. Word	Current recipe number (1 - 999)		
2. Word	Current data record number (0 - 65535)		
3. Word	Reserved		
4. Word	Status (0, 2, 4, 12)		
5. Word	Reserved		

- Status
  The status word (word 4) can adopt the following values:

Value		Meaning
Decimal	Binary	
0	0000 0000	Transfer permitted, data mailbox free
2	0000 0010	Transferring
4	0000 0100	Transfer completed without error
12	0000 1100	Transfer completed with error

Sequence of a transfer triggered by a job mailbox (Basic Panels)

The transfer of data records between the HMI device and the PLC can be initiated by either one of these stations.

The two job mailboxs No. 69 and No. 70 are available for this type of transfer.

No. 69: Read data record from PLC ("PLC → DAT")

Job mailbox no. 69 transfers data records from the PLC to the HMI device. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	69
Word 2	Recipe number (1-999)	

	Left byte (LB)	Right byte (RB)
Word 3	Data record number (1 to 65535)	
Word 4	Do not overwrite existing data record: 0 Overwrite existing data record: 1	

No. 70: Write data record to PLC ("DAT → PLC")

Job mailbox No. 70 transfers data records from the HMI device to the PLC. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	70
Word 2	Recipe number (1-999)	
Word 3	Data record number (1-65,535)	
Word 4	—	

Sequence when reading from the PLC with job mailbox "PLC → DAT" (no. 69)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data record.	Abort without return message.
3	The HMI device reads the values from the PLC and saves these to the data record defined in the job mailbox.	
4	• If "Overwrite" was selected in the job, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "Do not overwrite" was selected in the job, and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Sequence of writing to the PLC with job mailbox "DAT → PLC" (no. 70)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data mailbox.	Abort without return message.
3	The HMI device fetches the values of the data record specified in the function from the data medium and writes the values to the PLC.	

Step	Action	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Sequence of the transfer when triggered by a configured function (Basic Panels)

Reading from the PLC using a configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system event.
3	The HMI device reads the values from the PLC and stores them in the data record specified in the function.	
4	• If "Yes" was selected for the "Overwrite" function, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "No" was selected for the "Overwrite" function and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC by means of configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system event.
3	The HMI device fetches the values of the data record specified in the function from the data medium and transfers the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Possible causes of error when transferring data records (Basic Panels)

Possible causes of error

The section below shows possible error causes which lead to the cancellation of data record transfer:

- Tag address not set up on the PLC
- Overwriting data records not possible
- Recipe number does not exist
- Data record number does not exist

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Reaction to an aborted transfer due to errors

If the transfer of data records is aborted due to errors, the HMI device reacts as follows:

- Triggering by the operator in the recipe view
  Information in the status bar of the recipe view and output of system alarms
- Triggered by function
  Output of system alarms
- Triggering by job mailbox
  No return message on the HMI device

You can nonetheless evaluate the status of the transfer by querying the status word in the data record.

1.6.4.2 Trends (Basic Panels)

General information on trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

Trend request and trend transfer (Basic Panels)

Trend request area

The HMI device sets corresponding bits in the trend request area when you open a screen which contains one or more trends on the HMI device. After closing the screen, the HMI device resets the relevant bits in the trend request area.

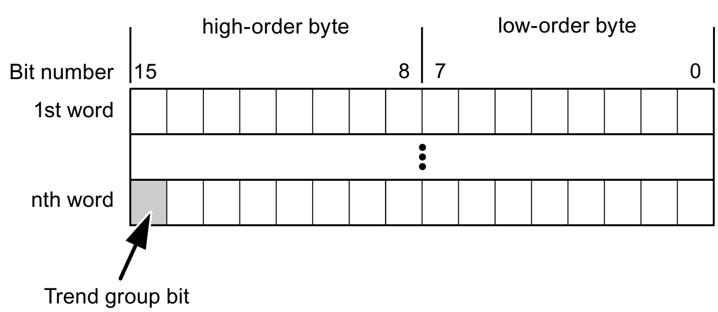
The trend request area can be used for evaluation purposes in the PLC to determine which trend is currently being displayed on the HMI device. Trends can also be triggered without evaluation of the trend request area.

Trend transfer area 1

This area is used to trigger trends. You must set the bit assigned to the trend in the trend transfer area and set the trend group bit in your control program. The trend group bit is the last bit in the trend transfer area.

The HMI device detects the trigger. The HMI device reads either a value or the entire buffer from the PLC. It then resets the trend bit and the trend group bit.

The following picture shows the structure of a trend transfer area.



The trend transfer area must not be modified by the PLC program until the trend group bit has been reset.

Trend transfer area 2

Trend transfer area 2 is required for trends configured with a switch buffer. The trend transfer areas 1 and 2 have a similar structure.

Switch buffer

The switch buffer is a second buffer for the same trend that can be set up during configuration.

The PLC writes to Buffer 2 while the HMI device reads values from Buffer 1, and writes to Buffer 1 when the HMI device is reading Buffer 2. This prevents the PLC from overwriting trend values while the trend is being read by the HMI device.

Permitted data types for trends (Basic Panels)

For SIMATIC S7

You assign one bit to each trend during configuration. Tags and array tags of the "Word" or "Int" data type are permitted.

1.6.4.3 Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configure alarms

Several steps are needed to configure alarms, such as operational messages, error alarms, and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a SIMATIC communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
SIMATIC S7 PLCs	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

How the bit positions are counted

For connections with a SIMATIC communication driver, the following counting method applies:

How the bit positions are counted	Byte 0								Byte 1							
	Most significant byte								Least significant byte							
In SIMATIC S7 PLCs	7							0	7							0
In WinCC you configure:	15							8	7							0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

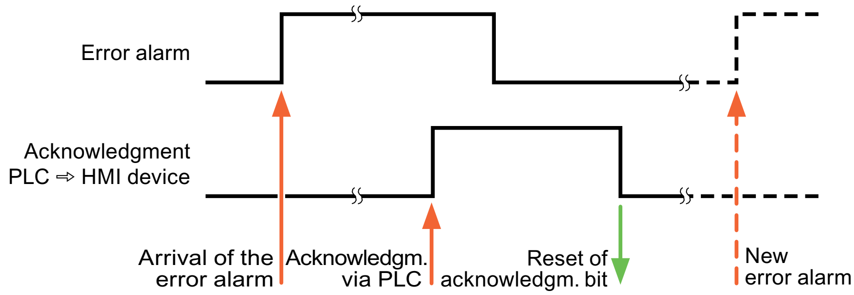
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

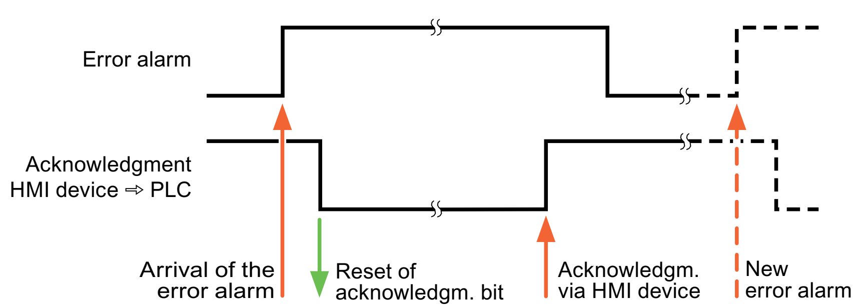
If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.6.5 Performance features of communication (Basic Panels)

1.6.5.1 SIMATIC S7-1500 device dependency (Basic Panels)

Device dependency

If you use devices from an earlier version of the TIA Portal with TIA Portal V13, it may not be possible to configure connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC S7-1500
KP300 Basic	No
KP400 Basic	No
KTP400 Basic PN	No
KTP600 Basic DP	No
KTP600 Basic PN	No
KTP1000 Basic DP	No
KTP1000 Basic PN	No
TP1500 Basic PN	No

Basic Panels V12.0

HMI devices	SIMATIC S7-1500
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V13.0

HMI devices	SIMATIC S7-1500
KTP400 Basic	Yes
KTP700 Basic	Yes
KTP900 Basic	Yes
KTP1200 Basic	Yes

Basic Panels V13.0.1

HMI devices	SIMATIC S7-1500
KTP400 Basic	Yes
KTP700 Basic	Yes
KTP900 Basic	Yes
KTP1200 Basic	Yes

1.6.5.2 Valid data types for SIMATIC S7 1500 (Basic Panels)

Valid data types for connections with SIMATIC S7 1500

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Length	
BOOL	1 bit	
BYTE	1 byte	
WORD	2 bytes	
DWORD	4 bytes	
CHAR	1 byte	
WCHAR	2 bytes	RT Professional
ARRAY of WCHAR	--	
INT	2 bytes	
DINT	4 bytes	
REAL	4 bytes	
TIME	4 bytes	
DATE	2 bytes	
TIME_OF_DAY	4 bytes	
S5TIME	2 bytes	
COUNTER	2 bytes	
TIMER	2 bytes	
DATE_AND_TIME	8 bytes	
STRING	(2+n) bytes, n = 0 to 254	
WSTRING	(4+2*n) bytes, n = 0 to 254	Basic Panels
	(4+2*n) bytes, n = 0 to 4094	Panels, RT Advanced
	(4+2*n) bytes, n = 0 to 65534	RT Professional
DTL	12 bytes	
LDT	8 bytes	
LINT	8 bytes	
LREAL	8 bytes	
LTIME	8 bytes	
LTIME_OF_DAY	8 bytes	
SINT	1 byte	

Data type	Length
UDINT	4 bytes
UINT	2 bytes
ULINT	8 bytes
USINT	1 byte

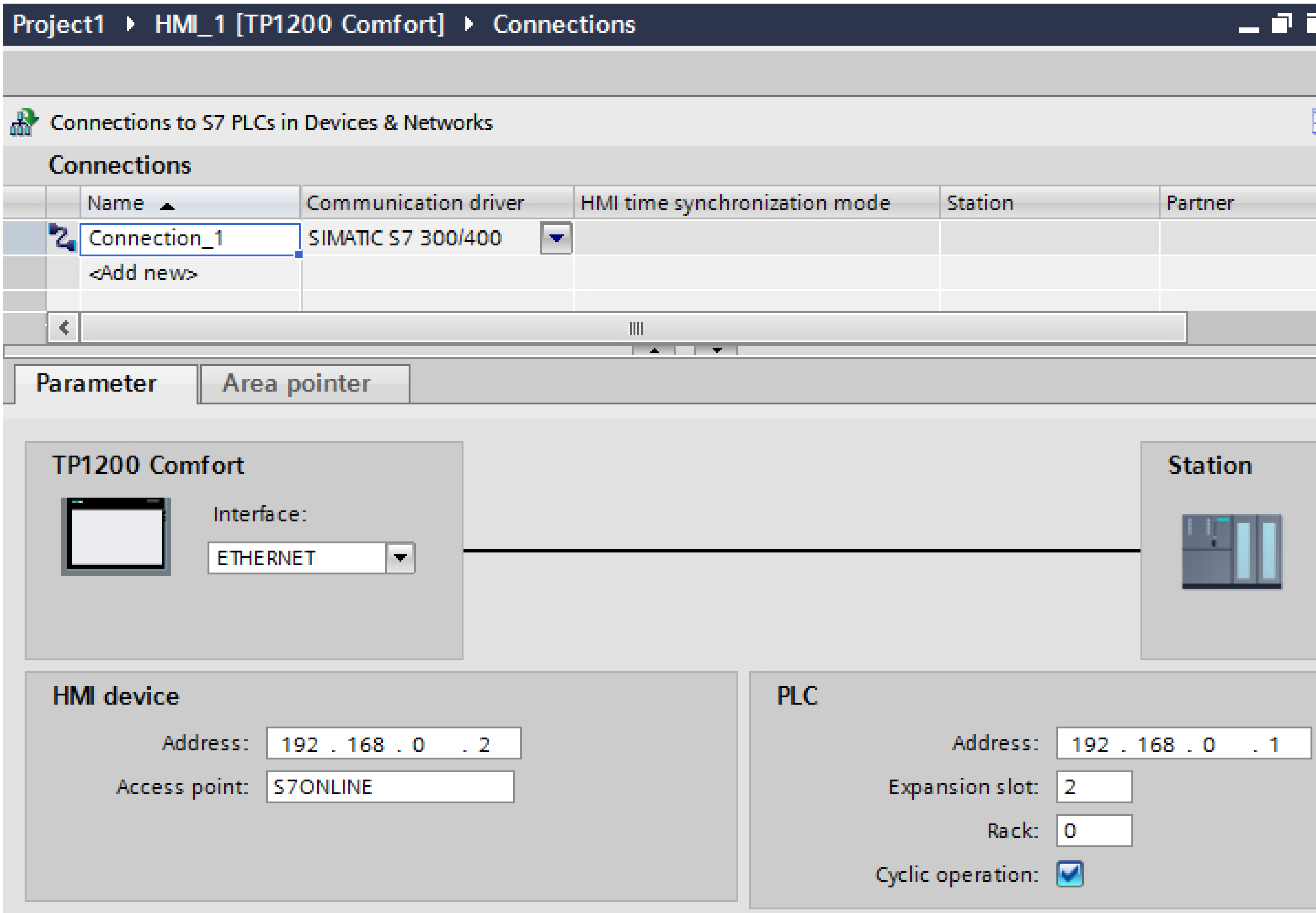
1.6.6 Configuring connections in the "Connections" editor (Basic Panels)

1.6.6.1 Connection parameters (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device", "Network", and "PLC" areas are available for assigning parameters according to the interface used.


Project1 ▸ HMI_1 [TP1200 Comfort] ▸ Connections
Connections to S7 PLCs in Devices & Networks
Connections
Name
Communication driver
HMI time synchronization mode
Station
Partner
Connection_1
SIMATIC S7 300/400
<Add new>
Parameter
Area pointer
TP1200 Comfort
Interface:
ETHERNET
Station
HMI device
Address:
192 . 168 . 0 . 2
Access point:
S7ONLINE
PLC
Address:
192 . 168 . 0 . 1
Expansion slot:
2
Rack:
0
Cyclic operation:

1.6.6.2 Ethernet parameters (Basic Panels)

Parameters for the HMI device

You set the parameters for the HMI device in the network under "HMI device".. The changes are not transferred automatically to the HMI device. You must change the settings in the Control Panel of the HMI device.

- "Interface"
  If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC.

  Note

  The IP address in the Control Panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

  The IP address already set up in the Control Panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

  The IP address is transferred to the HMI device during project transfer.
  To set up the IP address of the HMI device:

  – Click the HMI device.

  – Open the "Device configuration" editor.

  – Click the Ethernet interface.

  – Assign the IP address in the inspector window under:
  "General > PROFINET interface > Ethernet addresses"

- "Address"
  You assign the IP address of the HMI device in the "Address" area.
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

- "Access point"
  Specifies the access point for the PG/PC interface that can be used to reach the communication partner.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  Under "Address", set the IP address of the S7 module to which the HMI device is connected.
- "Access password"
  Enter a password in the "Access password" field. This password must match the one you saved to the PLC.

Note

You only need a password if you have set "Complete protection" at the PLC.

No connection is set up to the PLC if the "Complete protection" security level is stored on the PLC and you do not enter a password.

1.6.6.3 PROFIBUS parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the PROFIBUS network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

Note

If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  You set the PROFIBUS address of the HMI device under "Address". The PROFIBUS address must be unique in the PROFIBUS network.
- "Only master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master).
  In S7-200, you must set an HMI device as the master.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the network

Under "Network", you set the parameters for the PROFIBUS network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "DP", "Universal", or "Standard". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual PROFIBUS address. The setting must be identical throughout the network.

Note

If you set a baud rate of 1.5 Mbaud in the OP 73 or the OP 77A, the highest station address must be less than or equal 63.

- "Number of masters"
  For "Number of masters", set the number of masters in the PROFIBUS network. This information is necessary to correctly calculate the bus parameters.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the PROFIBUS address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Access password"
  Enter a password in the "Access password" field. This password must match the one you saved to the PLC.

Note

You only need a password if you have set "Complete protection" at the PLC.

No connection is set up to the PLC if the "Complete protection" security level is stored on the PLC and you do not enter a password.

1.7 Communicating with SIMATIC S7 1200 (Basic Panels)

1.7.1 Communication with SIMATIC S7 1200 (Basic Panels)

Introduction

This section describes the communication between an HMI device and the SIMATIC S7 1200 PLC.

You can configure the following communication channels for the SIMATIC S7 1200 PLC:

- PROFINET
- PROFIBUS

HMI connection for communication

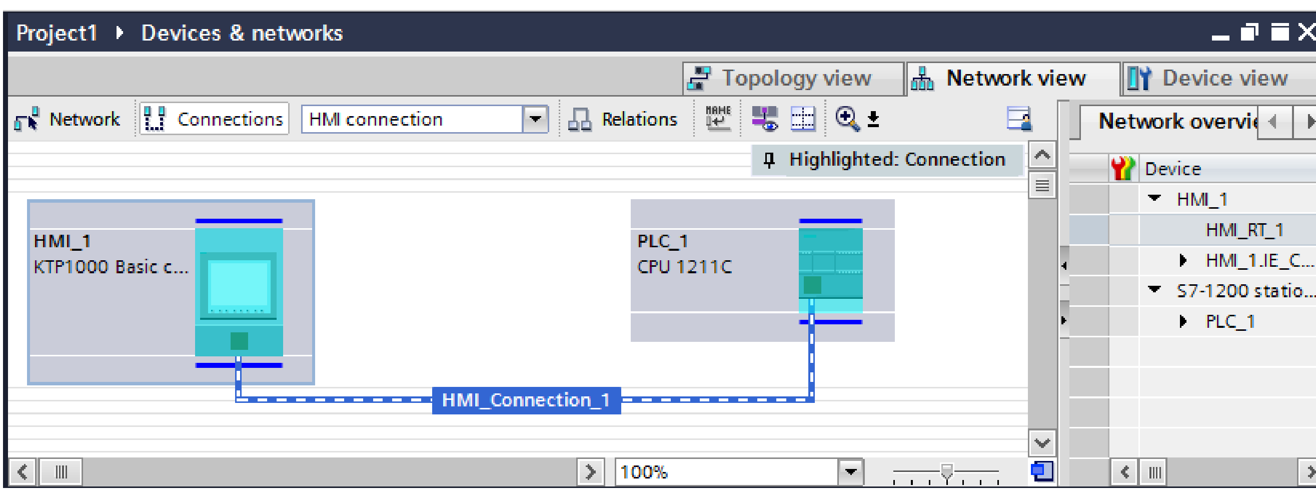
You configure connections between the HMI device and a SIMATIC S7 1200 in the "Devices & Networks" editor. If you have configured a HMI device with a serial port, you must configure a PROFIBUS-capable communication module to the SIMATIC S7 1200.

1.7.2 Communication via PROFINET (Basic Panels)

1.7.2.1 Communication via PROFINET (Basic Panels)

HMI connections via PROFINET

If you have inserted an HMI device and a SIMATIC S7 1200 into the project, you interconnect the two PROFINET interfaces in the "Devices & Networks" editor.



You can also connect multiple HMI devices to one SIMATIC S7 1200 and multiple SIMATIC S7 1200s to one HMI device. The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used.

Additional information is available in the documentation for the respective HMI device.

HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFINET in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device via PROFINET in the "Connections" editor of the HMI device.

1.7.2.2 Configuring an HMI connection via PROFINET (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC S7 1200 via PROFINET or Ethernet in the "Devices & Networks" editor.

> ⚠ CAUTION
>
> **Communication via Ethernet**
>
> In Ethernet-based communication, the end user is responsible for the security of his data network.
>
> Targeted attacks can overload the device and interfere with proper functioning.

Requirements

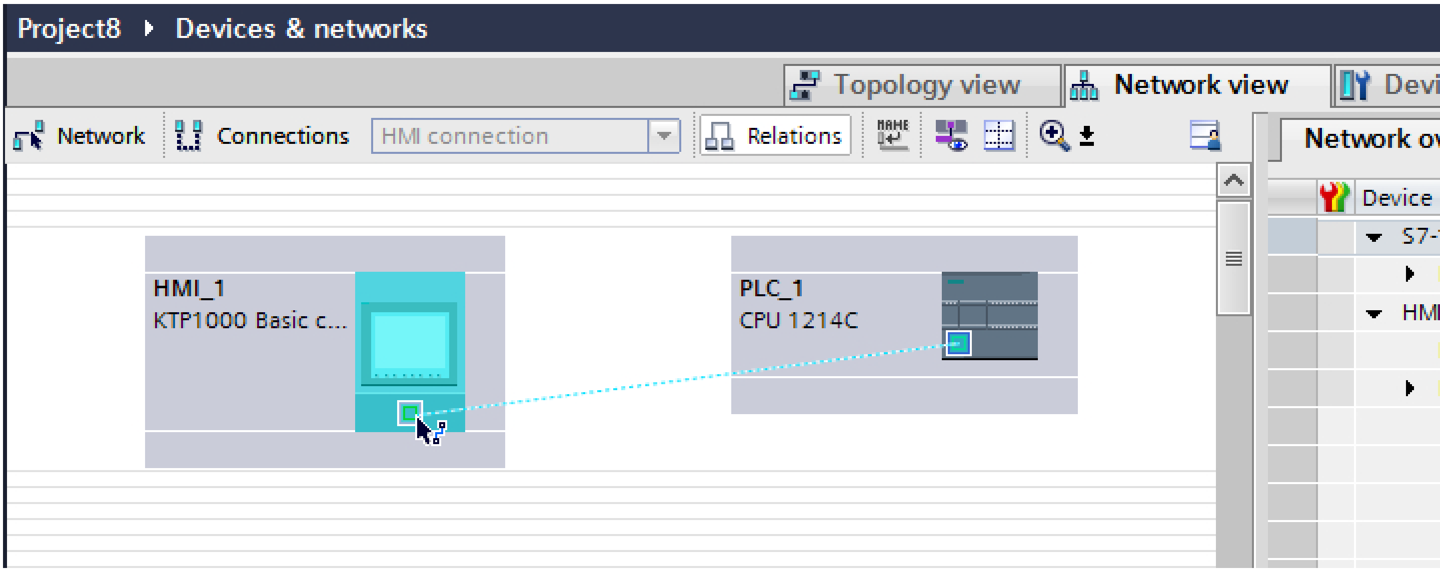
The following communication partners are created in the "Devices & Networks" editor:

- SIMATIC S7 1200
- HMI device with PROFINET or Ethernet interface

Procedure

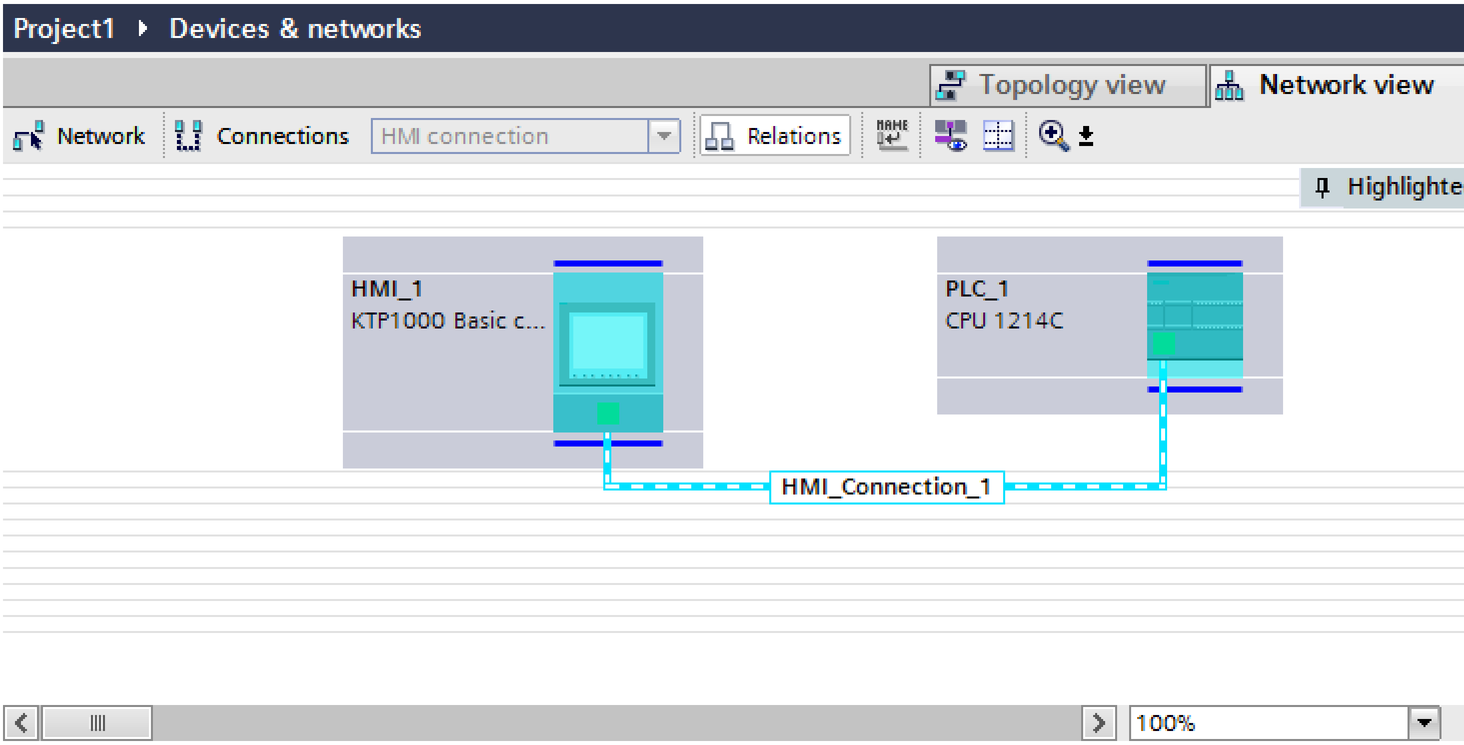
1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button and select "HMI connection" for the connection type.
   The devices available for connection are highlighted in color.

3. Click the PROFINET interface of the PLC and use a drag-and-drop operation to draw a connection to the PROFINET or Ethernet interface of the HMI device.



4. Click the connecting line.

5. Click "Highlight HMI connection" and select the HMI connection.



The connection is displayed graphically in the Inspector window.

6. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project. See the chapter "PROFINET parameters (Page 123)" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created a connection between an HMI device and a SIMATIC S7 1200. The IP address and subnet mask connection parameters are configured.

1.7.2.3 PROFINET parameters (Basic Panels)

PROFINET parameters for the HMI connection (Basic Panels)

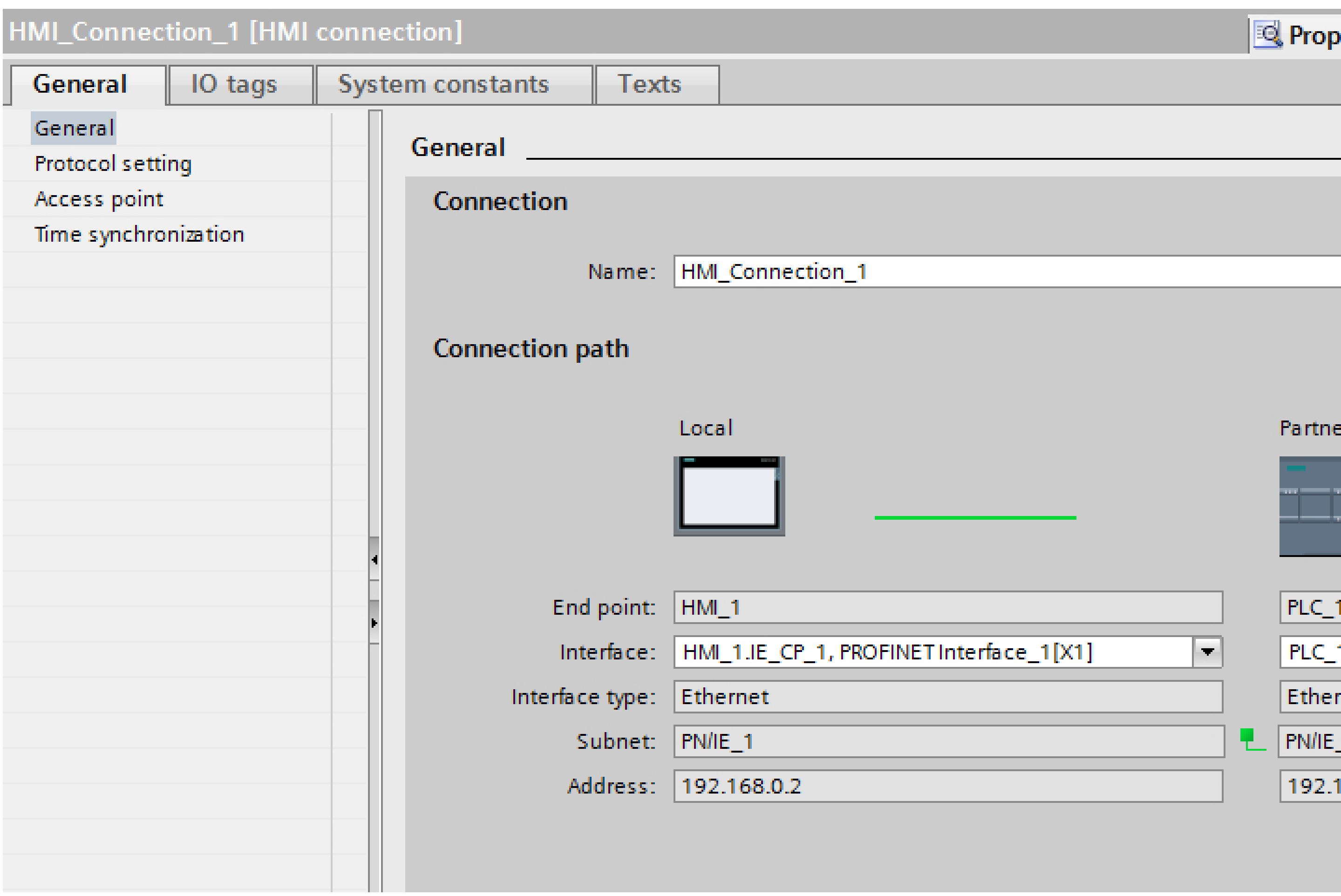
PROFINET parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and editing HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



Connection

The "Connection" area displays the HMI connection created for communication between the devices.

You can edit the name of the HMI connection in this area.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFINET parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area cannot be edited.
- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the selected IP address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

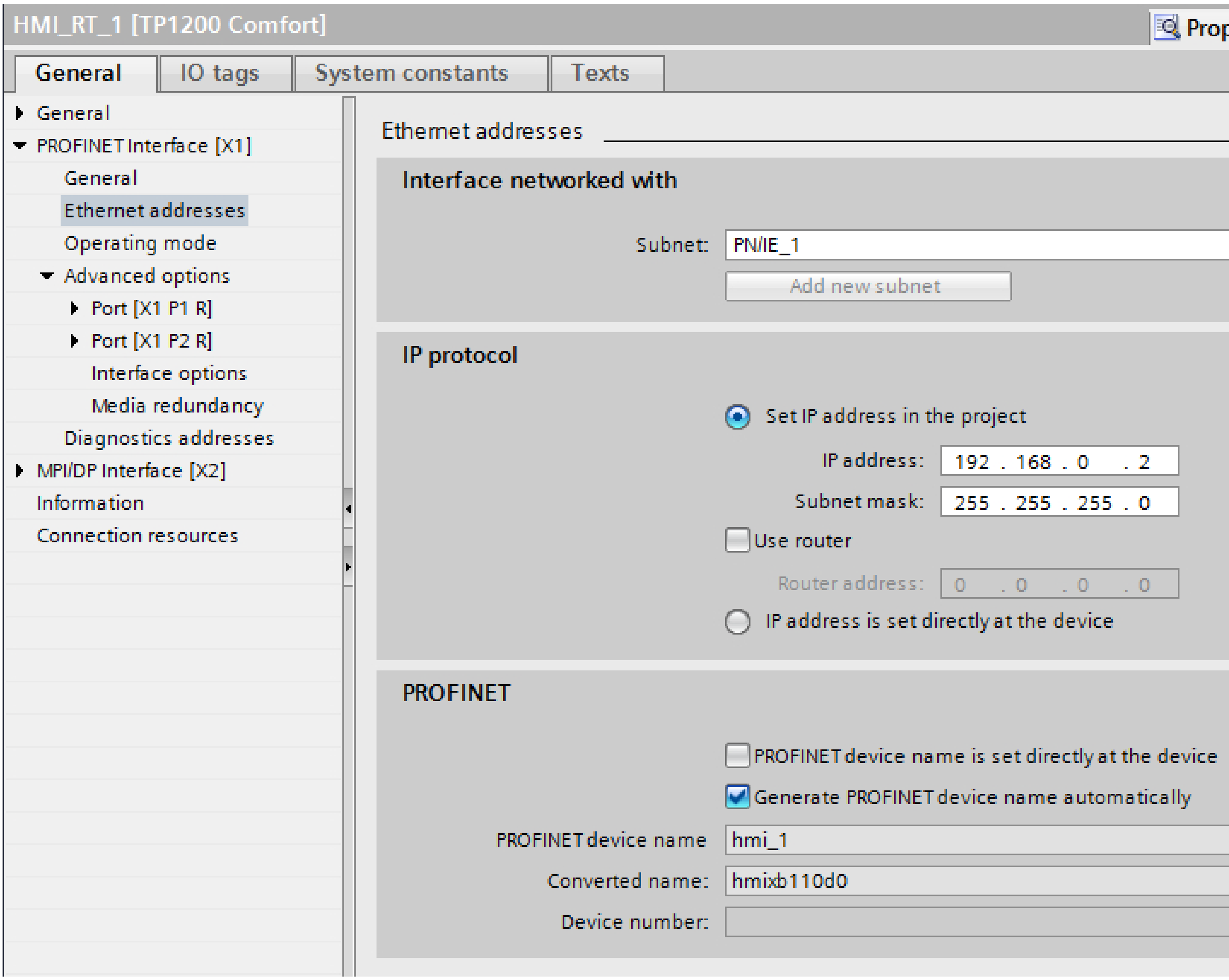
PROFINET parameters for the HMI device (Basic Panels)

PROFINET parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Set IP address in the project"
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

  Note

  The device is automatically restarted in the case of HMI devices with the Windows CE 3.0 operating system.

  HMI devices with Windows CE 3.0:
  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
- "Use IP router"
  If you are using an IP router, select "Use IP router" and enter the router address in the "Router address" field.
- "Set IP address using a different method"
  If the function "Set IP address using a different method" is activated, the IP address is not taken from the project. You have to enter the IP address directly in the Control Panel of the HMI device.

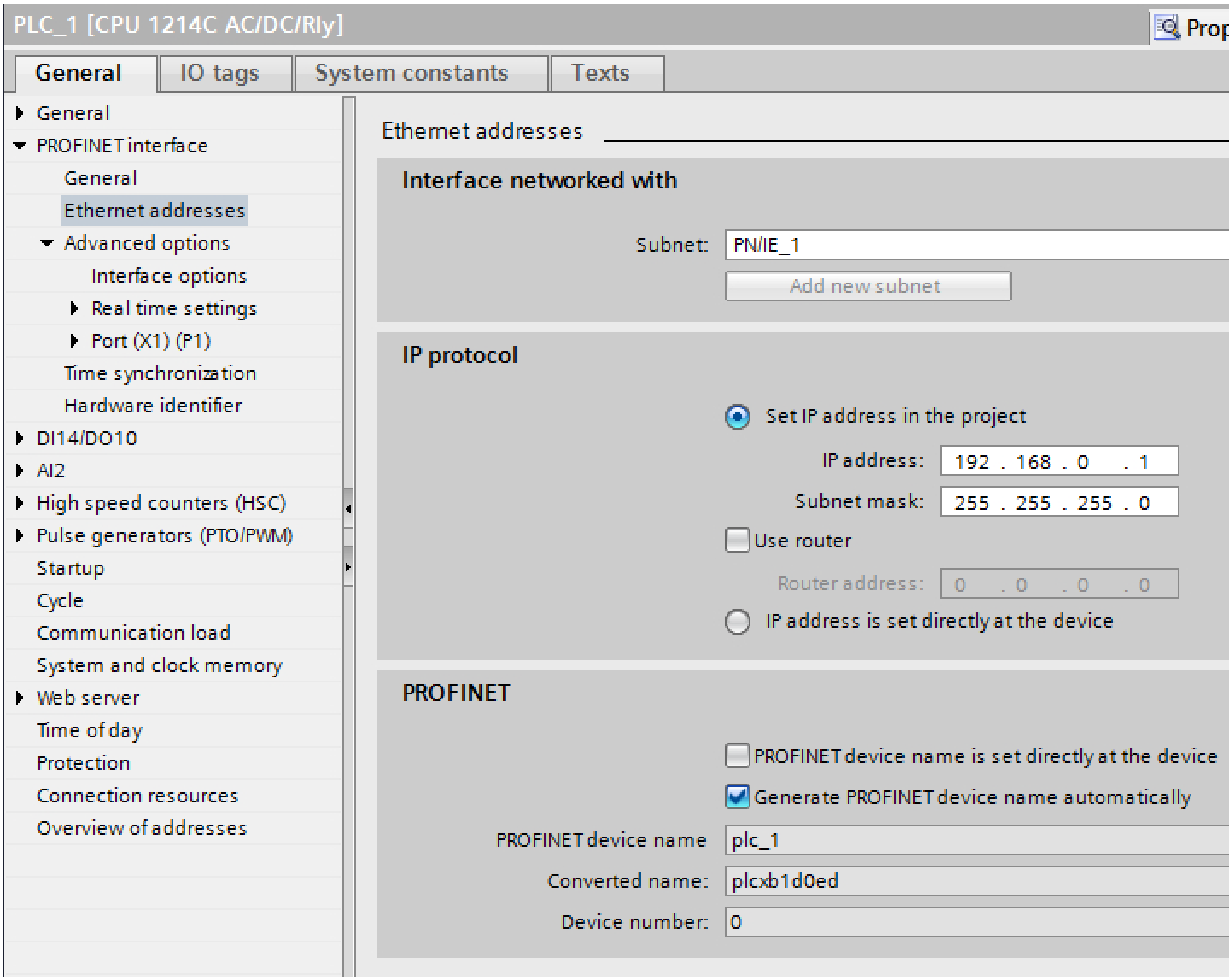
PROFINET parameters for the PLC (Basic Panels)

PROFINET parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFINET parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "IP address"
  You assign the IP address of the HMI device in the "IP address" area.
- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
  If you are using an IP router, select "Use IP router" and enter the router address in the field.

Configuring Industrial Ethernet (Basic Panels)

Rules for the network configuration

The Ethernet interfaces of the devices have a default IP address that you can change.

IP address

The IP parameters are visible if the communication-capable devices support the TCP/IP protocol.

The IP address consists of 4 decimal figures in the range of 0 to 255. The decimal figures are separated from one another by a dot.

Example: 140.80.0.2

The IP address consists of the following:

- The address of the (sub) net
- The address of the node (generally also called host or network node)

Subnet mask

The subnet mask splits these two addresses. It determines which part of the IP address addresses the network and which part of the IP address addresses the node.

The set bits of the subnet mask determine the network part of the IP address.

Example:

Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000

In the example given for the above IP address, the subnet mask shown here has the following meaning:

The first 2 bytes of the IP address identify the subnet - i.e. 140.80. The last two bytes address the node, thus 0.2.

It is generally true that:

- The network address results from AND linking the IP address and subnet mask.
- The node address results from AND NOT linking the IP address and subnet mask.

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. It determines 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 of between 224 and 255 is also possible for the first decimal number of the IP address (address class D etc). This is, however, 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 lower points of the subnet mask to "1". For each bit set to "1", the number of "private" networks doubles and the number of nodes they contain is halved. Externally, the network functions like an individual network as it did previously.

Example:

You have a subnet of address class B (e.g. 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 nodes with addresses between 129.80.001.xxx and 129.80.127.xxx are on one subnet, all nodes with addresses between 129.80.128.xxx and 129.80.255.xxx are on another subnet.

Router

The job of the routers is to connect the subnets. If an IP datagram is to be sent to another network, it first has to be conveyed to a router. To make this possible, in this case you have to enter the address of the router for each node in the subnet.

The IP address of a node in the subnet and the address of the router may only differ at the points at which there is a "0" in the subnet mask.

Protection of communication (Basic Panels)

Security levels (Basic Panels)

If you want to protect the controller and HMI device communication, you can assign protection levels for the communication.

For a SIMATIC S7-1200 CPU, you can enter multiple passwords and thereby set up different access rights for various user groups.

The passwords are entered in a table, so that exactly one protection level is assigned to each password.

The effect of the password is given in the "Protection" column.

For the SIMATIC S7-1200 controller, several aspects need to be considered when setting protection levels.

For additional information on this, see:

Setting options for the protection (FW as of V4) (Page 130)

Setting options for the protection level (FW V1 to V3) (Page 132)

Example

When configuring the controller, you select the "Complete protection" protection level for a standard CPU (i.e., not an F-CPU).

Afterwards, you enter a separate password for every protection level above it in the table.

For users who do not know any of the passwords, the CPU is completely protected. Not even HMI access is possible.

For users who know one of the assigned passwords, the effect depends on the table row in which the password occurs:

- The password in row 1 (no protection) allows access as if the CPU were completely unprotected. Users who know this password have unrestricted access to the CPU.
- The password in row 2 (write protection) allows access as if the CPU were write-protected. Despite knowing the password, users who know this password only have read access to the CPU.
- The password in row 3 (read and write protection) allows access as if the CPU were read-protected and write-protected, so that only HMI access is possible for users who know this password.

Setting options for the protection (FW as of V4) (Basic Panels)

Protection level

The following section describes how to use the various access levels of the S7-1200 CPUs as of V4.

S7-1200 CPUs provide various access levels to limit the access to specific functions.

The parameters for the access levels are assigned in a table. The green checkmarks in the columns to the right of the respective access level specify which operations are possible without knowing the password of this access level. If you want to use the functions of check boxes that are not selected, a password has to be entered.

NOTICE
Configuring an access level does not replace know-how protection
Configuring access levels prevents unauthorized changes to the CPU by restricting download privileges. However, blocks on the memory card are not write- or read-protected. Use know-how protection to protect the code of blocks on the memory card.

Default characteristics

The default access level is "Full access (no protection)". Every user can read and change the hardware configuration and the blocks. A password is not set and is also not required for online access.

The access levels in detail

With an S7-1200 CPU, you can configure the following access levels:

- Full access (no protection): The hardware configuration and the blocks can be read and changed by all users.
- Read access: With this access level, only read access to the hardware configuration and the blocks is possible without entering a password - meaning that you can load the hardware configuration and blocks into the programming device. In addition, HMI access and access to diagnostics data is possible.
  You cannot load blocks or a hardware configuration into the CPU without entering the password. Moreover, writing test functions and firmware updates are **not** possible without a password.
- HMI access: With this access level, only HMI access and access to diagnostics data is possible without entering the password.
  Without entering the password, you can neither load blocks and hardware configuration into the CPU, nor load blocks and hardware configuration from the CPU into the programming device. In addition, the following is **not** possible without a password: Writing test functions, changing the operating state (RUN/STOP) and firmware updates.
- No access (complete protection): When the CPU is completely protected, no read or write access to the hardware configuration and the blocks is possible. HMI access is also not possible. The server function for PUT/GET communication is disabled in this access level (cannot be changed).
  Authorization with the password again provides you full access to the CPU.

Behavior of a password-protected module during operation

The CPU protection takes effect after the settings are downloaded to the CPU.

Validity is checked before the online function is executed. If password protection is in place, you are prompted to enter a password.

Example: The module was configured with read access and you want to execute the "Modify tags" function. This requires write access; therefore, the assigned password must be entered to execute the function.

The functions protected by a password can only be executed by one programming device/PC at any one time. Another programming device/PC cannot log on.

Access authorization to the protected data is in effect for the duration of the online connection or until the access authorization is manually rescinded with "Online > Delete access rights".

Each access level allows unrestricted access to certain functions without entering a password, for example, identification using the "Accessible devices" function.

Setting options for the protection level (FW V1 to V3) (Basic Panels)

Protection level

The following section describes how to use the various protection levels of the S7-1200 CPUs V1 to V3.

Effects of the protection level setting

You can choose between the following protection levels:

- No protection: This corresponds to the default behavior. You cannot enter a password. Read and write access is always permitted.
- Write protection: Only read-only access is possible. You cannot change any data on the CPU and cannot load any blocks or a configuration. HMI access and communication between CPUs are excluded from the write protection. Assignment of a password is required to select this protection level.
- Write/read protection: No write or read access is possible in the "Accessible devices" area or in the project for devices that are switched online. Only the CPU type and the identification data can be displayed in the project tree under "Accessible devices". Display of online information or blocks under "Accessible devices", or in the project for devices interconnected online, is possible.
  HMI access and communication between CPUs are excluded from the write protection. Assignment of a password is required to select this protection level.

Behavior of a password-protected CPU during operation

The CPU protection takes effect after the settings are downloaded to the CPU.

Validity is checked before the online function is executed. If password protection is in place, you are prompted to enter a password.

Example: The module was assigned write protection and you want to execute the "Modify tags" function. This requires write access; therefore, the assigned password must be entered to execute the function.

The functions protected by a password can only be executed by one programming device/PC at any one time. Another programming device/PC cannot log on with a password.

Access authorization to the protected data is in effect for the duration of the online connection or until the access authorization is manually rescinded with "Online > Delete access rights". Access authorization will also expire when the project is closed.

Note

You can not restrict functions for process control, monitoring, and communications.

Some functions are still protected due to their use as online data. RUN/STOP in the "Online Tools" task card or "Set the time" in the diagnostics and online editor is therefore write-protected.

Access password for the HMI connection (Basic Panels)

Introduction

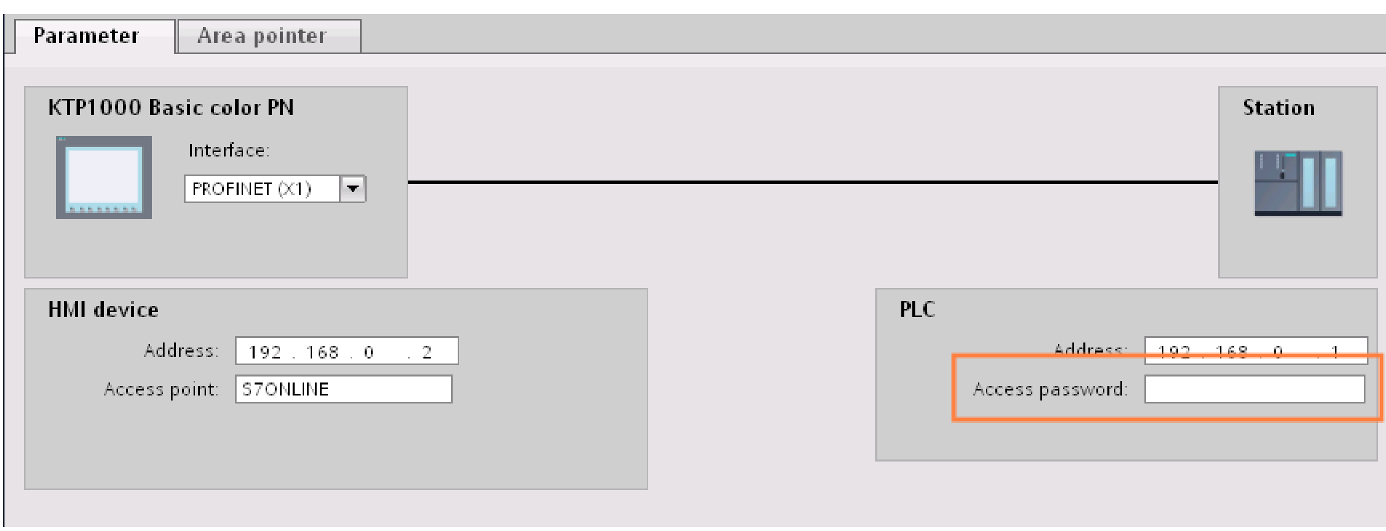
You secure access to a PLC by assigning a password.

You assign the password when you configure the password.

A password is mandatory as of protection level "Complete protection". Communication to the PLC is denied if an incorrect password or no password is entered.

Entering access password

You enter the password for the PLC in the "Connections" editor.



1.7.2.4 Setting port options (Basic Panels)

Setting the port options (Basic Panels)

Changing connection settings for the PROFINET IO port

You can change the network settings for the PROFINET IO port as required. By default, the settings are made automatically. In normal situations, this guarantees problem-free communication.

Possible settings for transmission rate / duplex

Depending on the selected device, you can make the following settings for "Transmission rate / duplex":

- Automatic setting
  Recommended default setting of the port. The transmission settings are automatically "negotiated" with the peer port. The "Enable autonegotiation" option is also enabled as a default, in other words, you can use cross cables or patch cables for the connection.
- TP/ITP at x Mbps full duplex (half duplex)
  Setting of the transmission rate and the full duplex/half duplex mode. The effectiveness depends on the "Enable autonegotiation" setting:
  - Autonegotiation enabled
    You can use both cross cable and patch cable.
  - Autonegotiation disabled
    Make sure that you use the correct cable (patch cable or cross cable)! The port is also monitored with this setting.
- Deactivated
  Depending on the module type, the drop down list box can contain the "- Disabled -" option. This option, for example, allows you to prevent access to an unused port for security reasons. With this setting, diagnostic events are not generated.

"Monitor" option

This option enables or disables port diagnostics. Examples of port diagnostics: The link status is monitored, in other words, the diagnostics are generated during link-down and the system reserve is monitored in the case of fiber optic ports.

Option "Enable autonegotiation "

The autonegotiation setting can only be changed if a concrete medium (for example, TP 100 Mbps full duplex) is selected. Whether or not a concrete medium can be set depends on the properties of the module.

If autonegotiation is disabled, this causes the port to be permanently specified, as for example, is necessary for a prioritized startup of the IO device.

You must make sure the partner port has the same settings because with this option the operating parameters of the connected network are not detected and the data transmission rate and transmission mode can accordingly not be optimally set.

Note

When a local port is connected, STEP 7 makes the setting for the partner port if the partner port supports the setting. If the partner port does not accept the setting, an error message is generated.

Wiring rules for disabled autonegotiation (Basic Panels)

Requirements

You have made the following settings for the port in question, for example, to accelerate the startup time of the IO device:

- Fixed transmission rate
- Autonegotiation incl. autocrossing disabled

The time for negotiating the transmission rate during startup has been saved.

If you have disabled autonegotiation, you must observe the wiring rules.

Wiring rules for disabled autonegotiation

PROFINET devices have the following two types of ports:

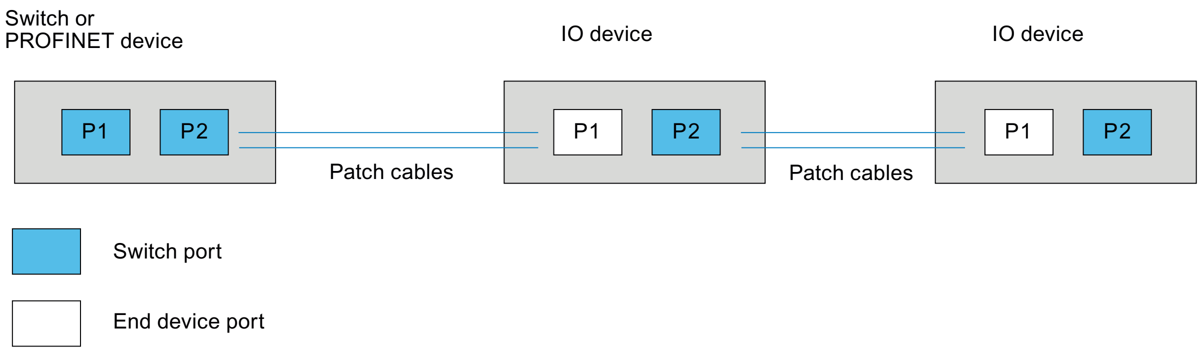
Type of port	PROFINET devices	Note
Switch port with crossed pin assignment	For IO devices: Port 2 For S7 CPUs with 2 ports: Ports 1 and 2	Crossed pin assignment means that the pin assignment for the ports for sending and receiving between the respective PROFINET devices is exchanged internally.
End device port with uncrossed pin assignment	For IO devices: Port 1 For S7 CPUs with one port: Port 1	-

Validity of the wiring rules

The cabling rules described in the following paragraph apply exclusively for the situation in which you have specified a fixed port setting.

Rules for cabling

You can connect several IO devices in line using a single cable type (patch cable). To do this, you connect port 2 of the IO device (distributed I/O) with port 1 of the next IO device. The following graphic gives an example with two IO devices.



Boundaries at the port (Basic Panels)

Requirements

To use boundaries, the respective device must have more than one port. If the PROFINET does not support boundary settings, they are not shown.

Enable boundaries

"Boundaries" are limits for transmission of certain Ethernet frames. The following boundaries can be set at a port:

- "End of discovery of accessible devices"
  No forwarding of DCP frames to identify accessible devices. Devices downstream from this port cannot be reached by the project tree under "Accessible devices". Devices downstream from this port cannot be reached by the CPU.
- "End of topology discovery"
  LLDP frames (Link Layer Discovery Protocol) are not forwarded for topology detection.
- "End of sync domain"
  No forwarding of sync frames transmitted to synchronize nodes within a sync domain.
  If you operate, for example, a PROFINET device with more than two ports in a ring, you should prevent the sync frame from being fed into the ring by setting a sync boundary (at the ports not inside the ring).
  Additional example: If you want to use several sync domains, configure a sync domain boundary for the port connected to a PROFINET device from the other sync domain.

Restrictions

The following restrictions must be observed:

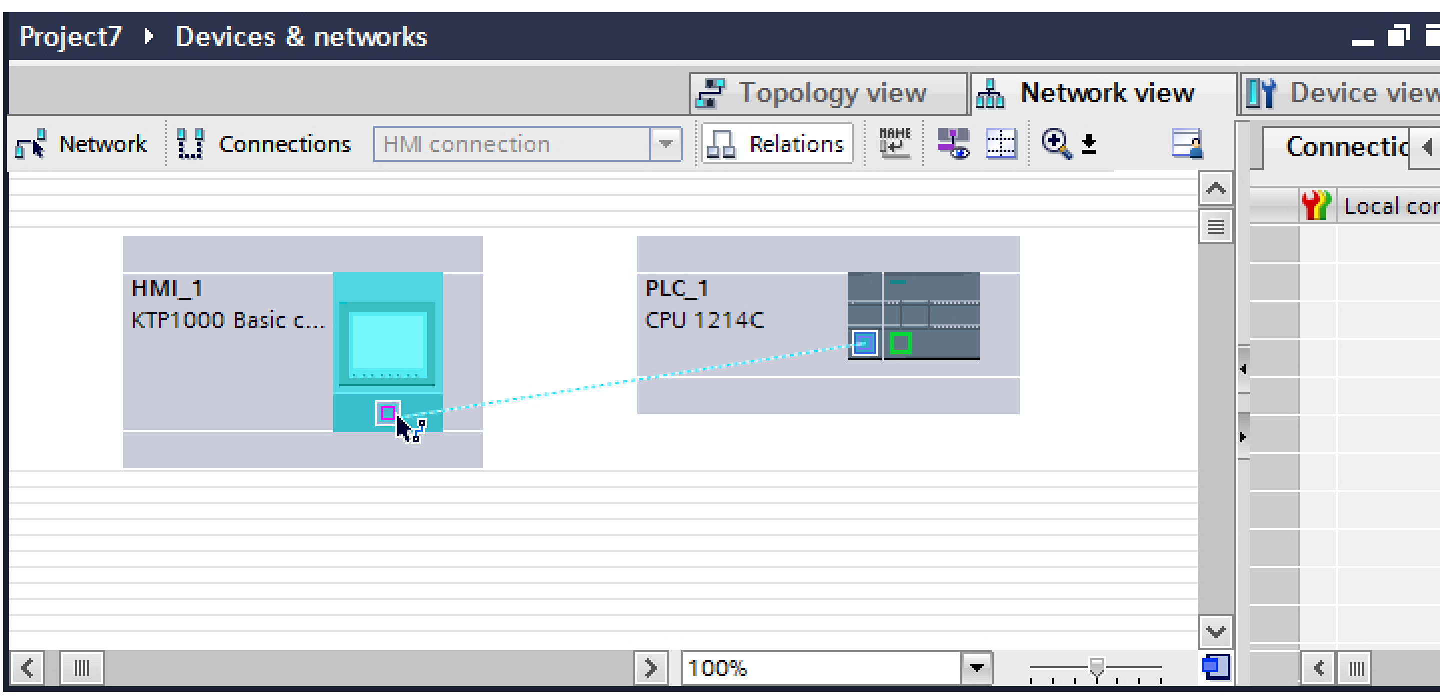
- The individual check boxes can only be used if the port supports the function in question.
- If a partner port has been determined for the port, the following check boxes cannot be used:
  - "End of discovery of accessible devices"
  - "End of topology discovery"
- If autonegotiation is disabled, none of the check boxes can be used.

1.7.3 Communication via PROFIBUS (Basic Panels)

1.7.3.1 Communication via PROFIBUS (Basic Panels)

HMI connections via PROFIBUS

If you want to connect a SIMATIC S7 1200 to a HMI device via PROFIBUS, you must configure a PROFIBUS-capable communication module to a slot of the controller first.



HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFIBUS in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device in the "Connections" editor of the HMI device.

1.7.3.2 Configuring an HMI connection via PROFIBUS (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC S7-1200 via PROFIBUS in the "Devices & Networks" editor.

Requirements

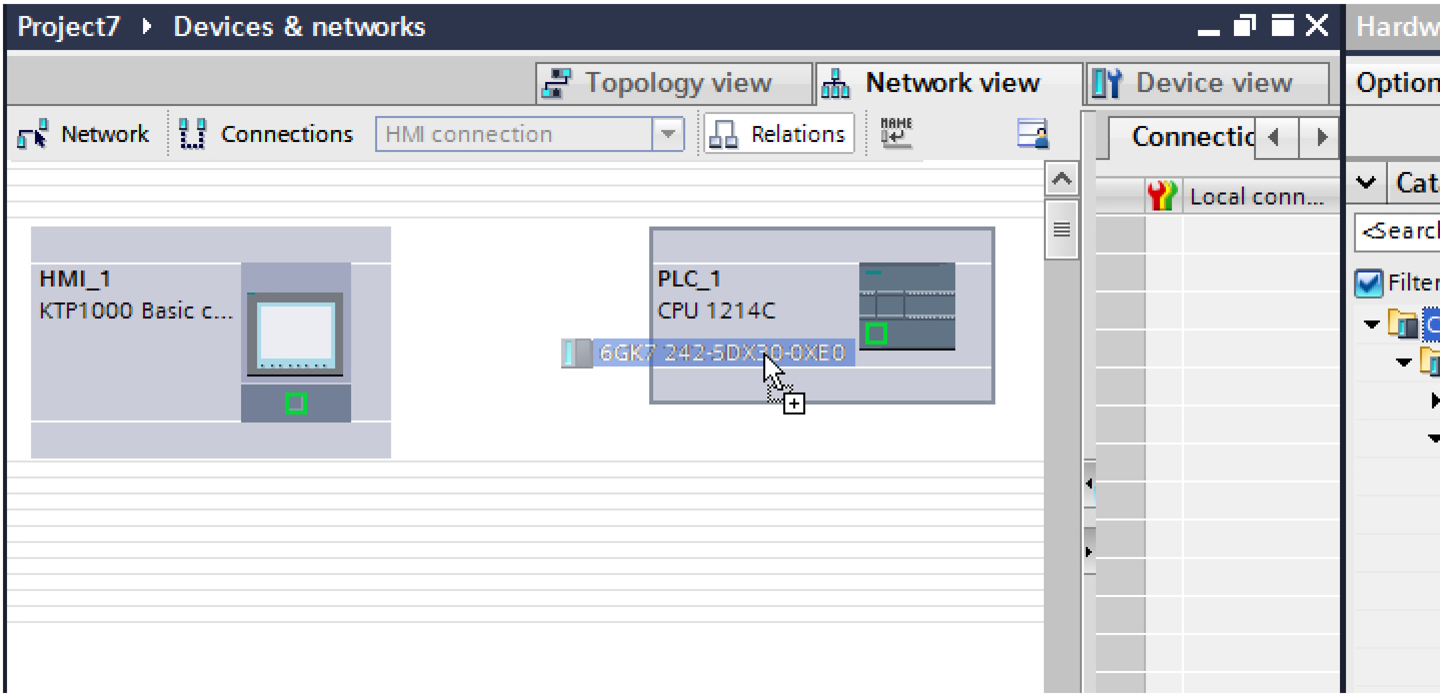
The following communication partners are created in the "Devices & Networks" editor:

- HMI device with MPI/DP interface
- SIMATIC S7-1200

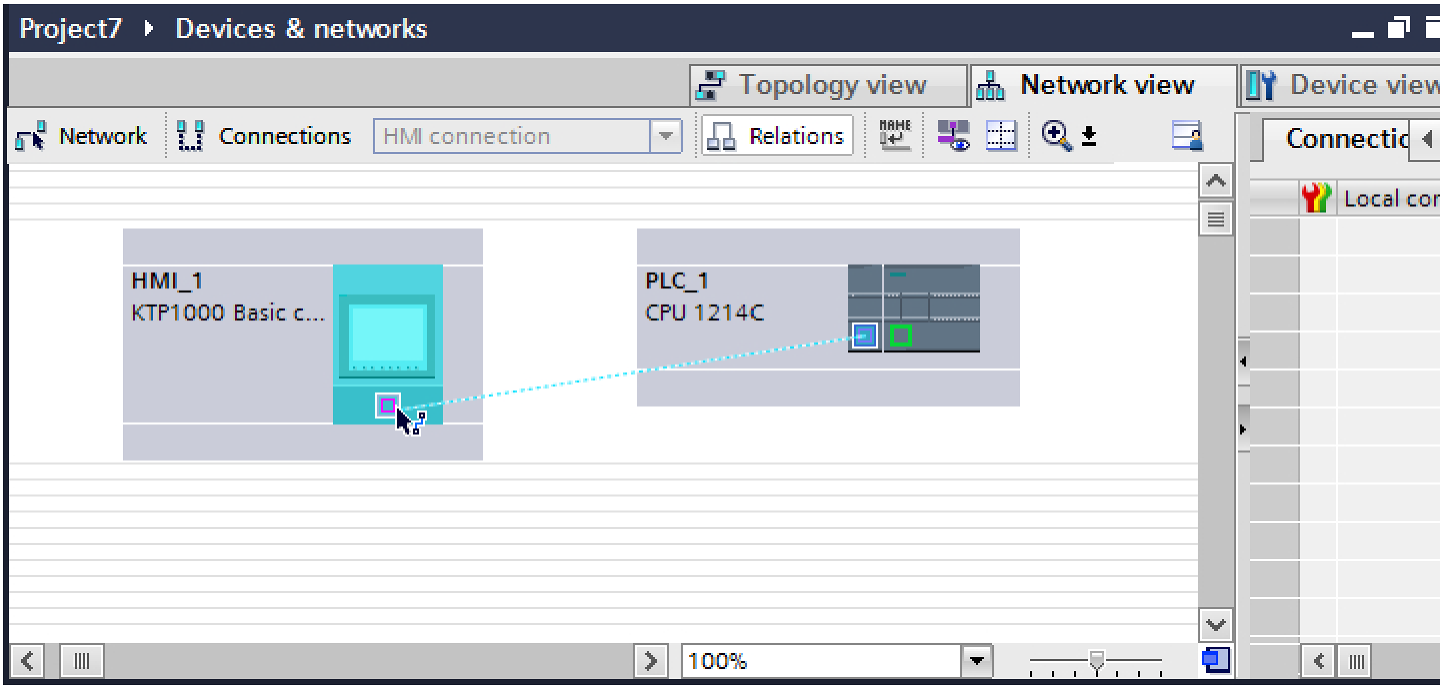
Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button.
   The devices available for connection are highlighted in color.

3. Use a drag-and-drop operation to move a PROFIBUS-capable communication module from the hardware catalog to the PLC.



4. Click the HMI device interface.
5. Select the "PROFIBUS" interface type in the Inspector window under "Properties > General > PROFIBUS address/ MPI address > Parameters".
6. Click the interface of the communication module and use a drag-and-drop operation to draw a connection to the HMI device.



7. Click the name of the connection.
   The connection is displayed graphically in the Inspector window.
8. Click "Highlight HMI connection" and select the HMI connection.
9. Click the communication partners in the "Network view" and change the PROFIBUS parameters in the Inspector window according to the requirements of your project.
   See the section "PROFIBUS parameters (Page 140)" for additional details.

   Note

   The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

   You can change the local name for the connection only in the table.

Result

You have created an HMI connection between an HMI device and a SIMATIC S7-1200 via PROFIBUS.

1.7.3.3 PROFIBUS parameters (Basic Panels)

PROFIBUS parameters for the HMI connection (Basic Panels)

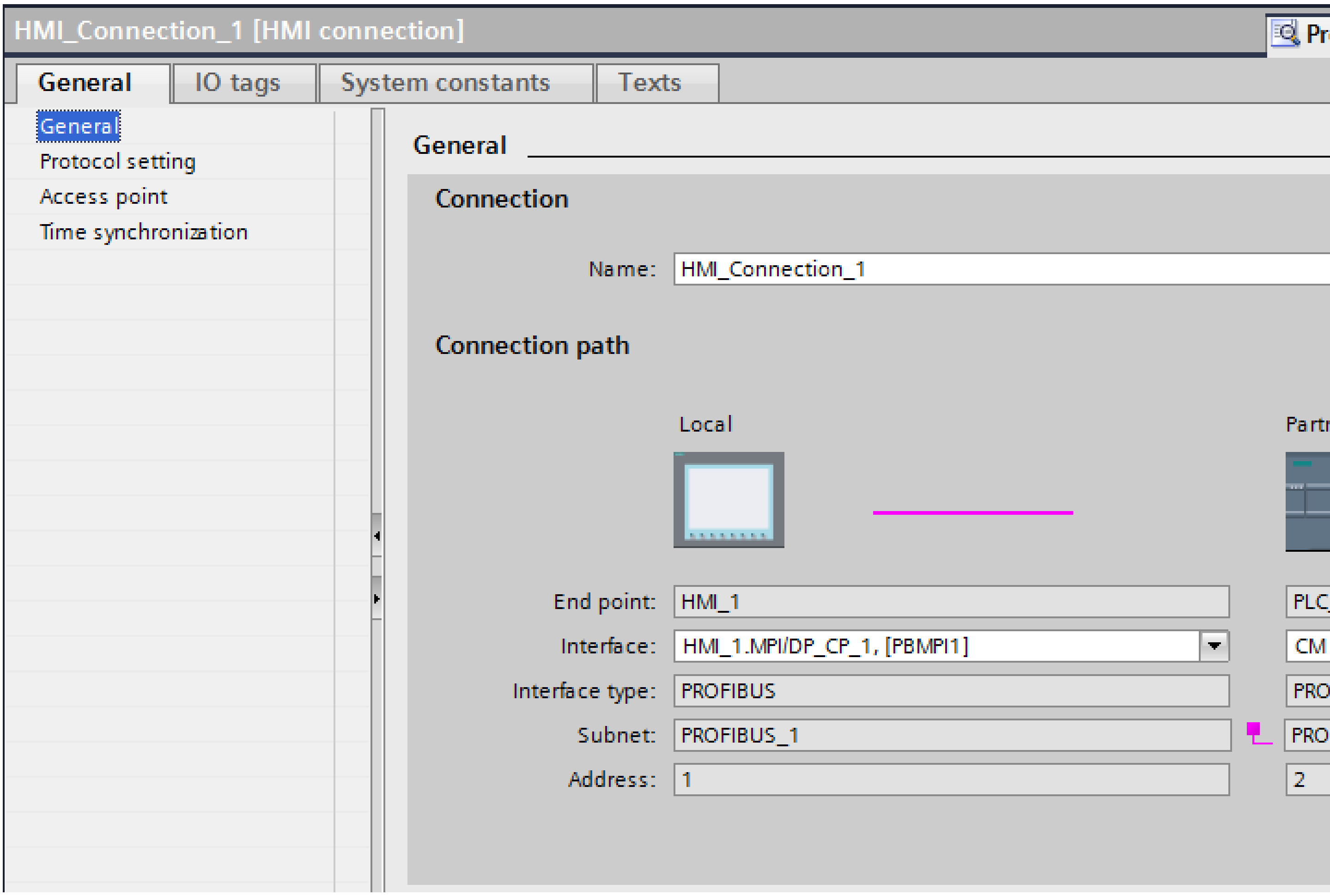
PROFIBUS parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and editing HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

The "Connection" area displays the HMI connection created for communication between the devices.

You can edit the name of the HMI connection in this area.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFIBUS parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"

Displays the selected interface type. This area cannot be edited.

- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the PROFIBUS address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

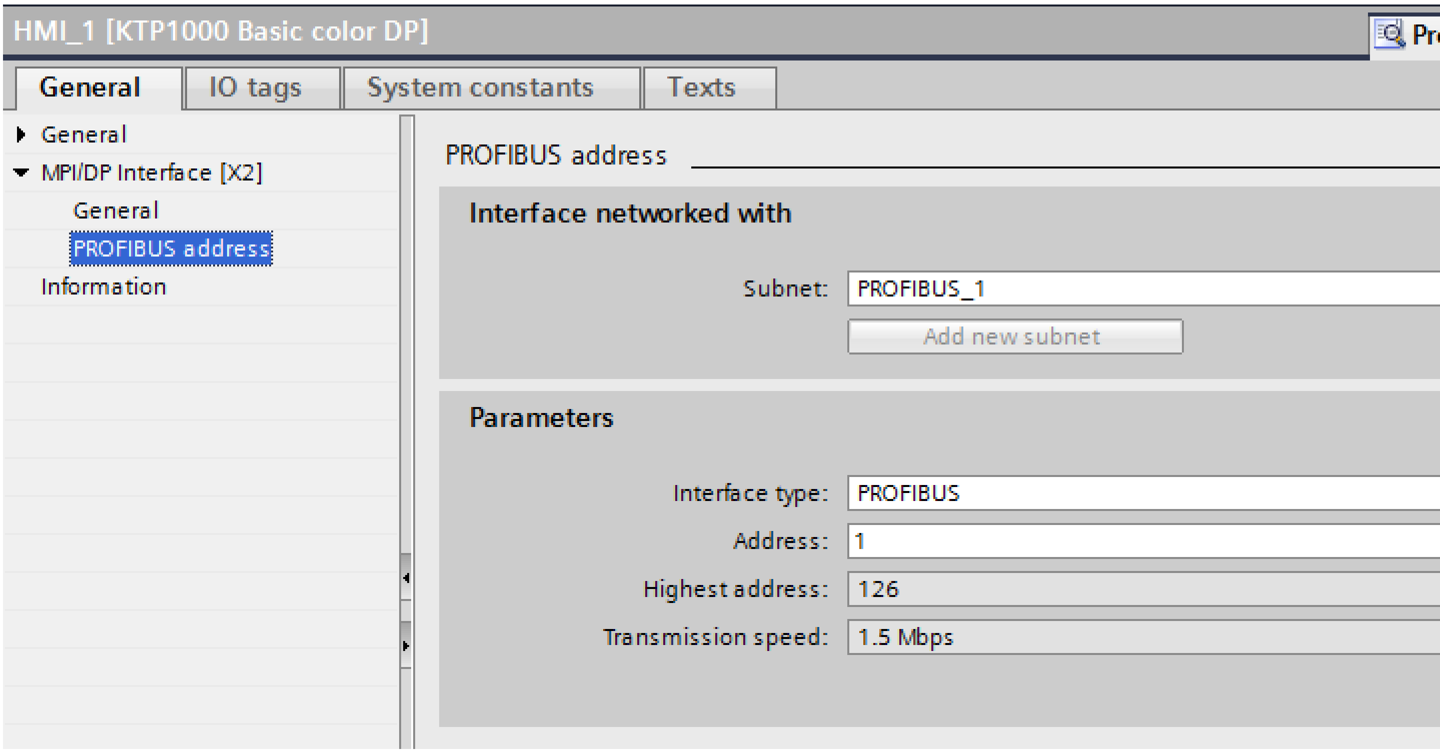
PROFIBUS parameters for the HMI device (Basic Panels)

PROFIBUS parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFIBUS parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.
- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

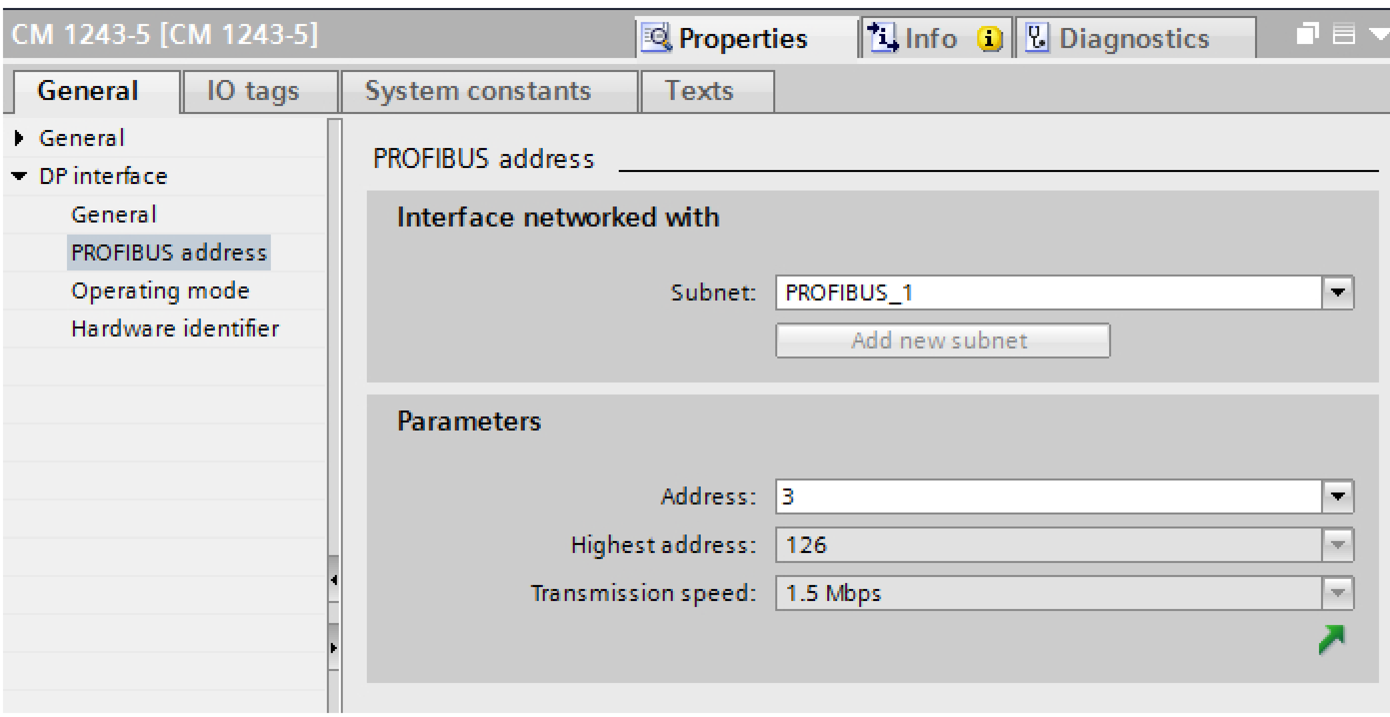
PROFIBUS parameters for the PLC (Basic Panels)

PROFIBUS parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFIBUS parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.

- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

Bus profiles with PROFIBUS (Basic Panels)

Introduction

Depending on the device types connected and protocols used on the PROFIBUS, different profiles are available. The profiles differ in terms of the setting options and calculation of bus parameters. The profiles are explained below.

Devices with different profiles on the same PROFIBUS subnet

The PROFIBUS subnet only functions without problem if the bus parameters of all devices have the same values.

Profiles and transmission rates

Profiles	Supported transmission speeds in Kbits/s
DP	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Standard	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Universal	9,6 19,2 93,75 187,5 500 1500

Meaning of profiles

Profile	Meaning
DP	Select the "DP" bus profile when the only devices connected to the PROFIBUS subnet are those which satisfy the requirements of standard EN 50170 Volume 2/3, Part 8-2 PROFIBUS. The bus parameter setting is optimized on these devices. This includes devices with DP master and DP slave interfaces of the SIMATIC S7 and distributed I/Os of other manufacturers.
Standard	Compared to the "DP" profile, the "Standard" profile also offers scope for devices of another project or devices which have not been configured here to be taken into account when calculating the bus parameters. The bus parameters are then calculated following a simple, non-optimized algorithm.
Universal	Select the "Universal" bus profile when individual devices on the PROFIBUS subnet use the PROFIBUS-FMS service. This includes the following devices for example: • CP 343-5 • PROFIBUS-FMS devices of other manufacturers As with the "Standard" profile, this profile allows you to take other devices into account when calculating the bus parameters.

Protection of communication (Basic Panels)

Security levels (Basic Panels)

If you want to protect the PLC and HMI device communication, you can assign protection levels for the communication.

For a SIMATIC S7-1500 CPU, you can enter multiple passwords and thereby set up different access rights for various user groups.

The passwords are entered in a table, so that exactly one protection level is assigned to each password.

The effect of the password is given in the "Protection" column.

For the SIMATIC S7-1200 controller, several aspects need to be considered when setting protection levels. For additional information on this, see: AUTOHOTSPOT

Example

You select the "Complete protection" protection level for a standard CPU (i.e., not an F-CPU) when configuring it.

Afterwards, you enter a separate password for every protection level above it in the table.

For users who do not know any of the passwords, the CPU is completely protected. Not even HMI access is possible.

For users who know one of the assigned passwords, the effect depends on the table row in which the password occurs:

- The password in row 1 (no protection) allows access as if the CPU were completely unprotected. Users who know this password have unrestricted access to the CPU.
- The password in row 2 (write protection) allows access as if the CPU were write-protected. Despite knowing the password, users who know this password only have read access to the CPU.
- The password in row 3 (read and write protection) allows access as if the CPU were read-protected and write-protected, so that only HMI access is possible for users who know this password.

Access password for the HMI connection (Basic Panels)

Introduction

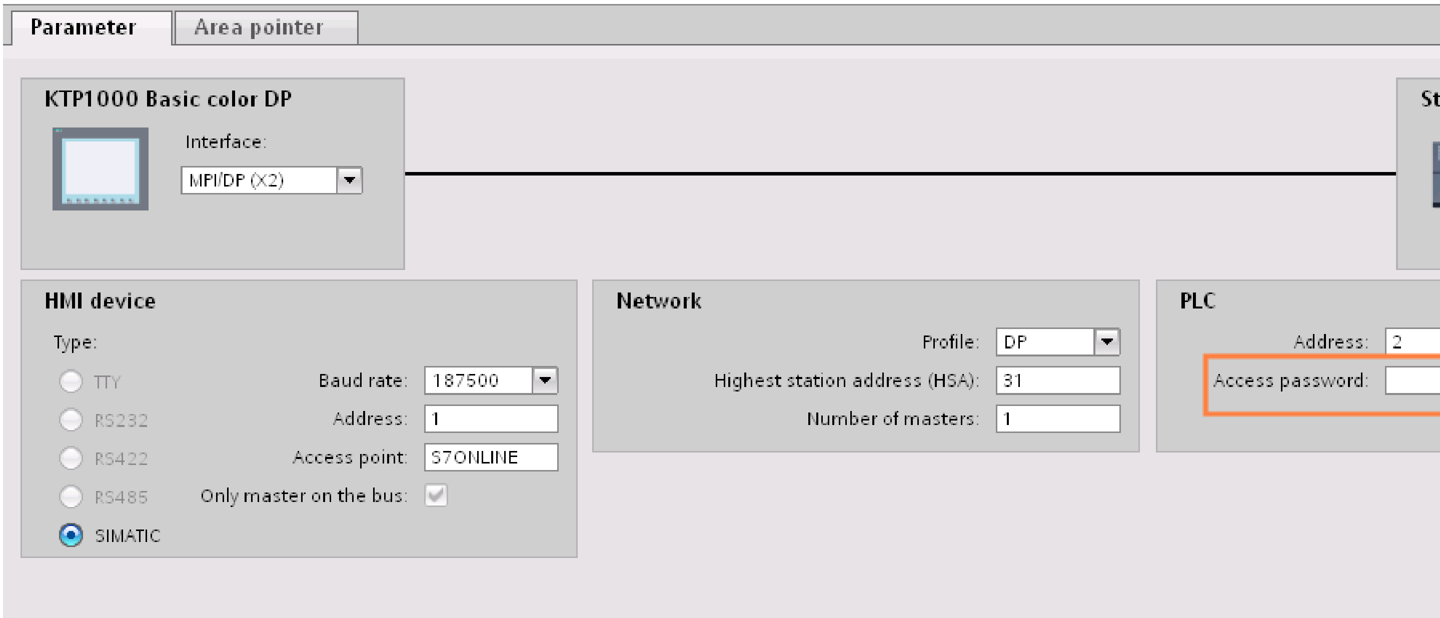
You secure access to a PLC by assigning a password.

You assign the password when you configure the password.

A password is mandatory as of protection level "Complete protection". Communication cannot be established with the PLC if you enter no password or an incorrect password.

Assigning password

You enter the password for the PLC in the "Connections" editor.



1.7.4 Data exchange (Basic Panels)

1.7.4.1 Data exchange using area pointers (Basic Panels)

General information on area pointers (Basic Panels)

Introduction

You use an area pointer to access a data area in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations.

The PLC and the HMI device trigger defined interactions based on the evaluation of stored data.

Configuration of area pointers

Before you use an area pointer, you enable it under "Connections > Area pointer". You then assign the area pointer parameters.

You can find more detailed information on configuring area pointers in:

AUTOHOTSPOT

Area pointer "Date/time" (Basic Panels)

Function

This area pointer is used to transfer the date and time from the HMI device to the PLC.

The PLC writes control job "41" to the job mailbox.

When it evaluates the control job, the HMI device writes its current date and the time in the data area configured in the "Date/time" area pointer.

Note

You cannot use the "Date/Time PLC" area pointer if you have configured the "Date/Time" area pointer.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

The date/time data area has the following structure:

Data word	Most significant byte	Least significant byte	
	7 … 0	7 … 0	
n+0	Reserved	Hour (0 to 23)	Time of day
n+1	Minute (0 to 59)	Second (0 to 59)	
n+2	Reserved	Reserved	
n+3	Reserved	Weekday (1 to 7, 1=Sunday)	Date
n+4	Day (1 to 31)	Month (1 to 12)	
n+5	Year (80 to 99/0 to 29)	Reserved	

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

Permitted data types

You can use the following data types when you configure the "Date/Time" area pointer.

- Int
- UInt
- Word
- DTL

Use of the "DTL" data type

You use data type "DTL" with communication driver S7-1200.

A tag of the "DTL" data type has a length of 12 bytes and saves information on date and time in a predefined structure.

The "DTL" data type has the following structure:

Byte	Component	Data type	Value range
0	Year	UINT	1970 to 2554
1			
2	Month	USINT	0 to 12
3	Day	USINT	1 to 31
4	Weekday	USINT	1(Sunday) to 7(Saturday) The weekday is not considered in the value entry.
5	Hour	USINT	0 to 23
6	Minute	USINT	0 to 59
7	Second	USINT	0 to 59
8	Nanoseconds	UDINT	0 to 999 999 999
9			
10			
11			

The "DTL" data type supports time information down to the nanosecond range. Because panels only support time information down to the millisecond, you may encounter the following restriction when using the area pointers:

For the transmission of time information from a panel to the controller, the smallest unit of time is 1 millisecond. The value range from microseconds to nanoseconds of the "DTL" data type is filled with zeros.

"Screen number" area pointer (Basic Panels)

Function

The HMI device saves information about the screen called on the HMI device to the "Screen number" area pointer.

This allows the transfer of the current screen contents from the HMI device to the PLC. The PLC can trigger specific reactions such as the call of a different screen.

Use

Configure and enable the area pointer in "Communication > Connections" before you put it into use. You can create only **one** instance of the "Screen number" area pointer and only on **one** PLC.

The screen number is always transferred to the PLC when a new screen is activated or when the focus within a screen changes from one screen object to another.

Structure

The area pointer is a data area in the memory of the PLC with a fixed length of 5 words.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1st word	Current screen type															
2nd word	Current screen number															
3rd word	Reserved															
4th word	Current field number															
5th word	Reserved															

- Current screen type
  "1" for root screen or
  "4" for permanent window
- Current screen number
  1 to 32767
- Current field number
  1 to 32767

"Date/time PLC" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the PLC to the HMI device. Use this area pointer if the PLC is the time master.

The PLC loads the data area of the area pointer.

The HMI device reads the data cyclically within the configured acquisition cycle and synchronizes itself.

Note

Set an acquisition cycle of sufficient length for the date/time area pointer PLC to avoid any negative impact on HMI device performance.
Recommended: Acquisition cycle of 1 minute, if the process allows this.

"Date/Time PLC" is a global area pointer and may be configured only once per project.

Note

You cannot use the "Date/Time" area pointer if you have configured the "Date/Time PLC" area pointer.

The date/time data area has the following structure:

DATE_AND_TIME format (in BCD code)

<table>
<tr><th rowspan="2">Data word</th><th colspan="3">Most significant byte</th><th colspan="3">Least significant byte</th></tr>
<tr><th>7</th><th>......</th><th>0</th><th>7</th><th>......</th><th>0</th></tr>
<tr><td>n+0</td><td colspan="3">Year (80 to 99/0 to 29)</td><td colspan="3">Month (1 to 12)</td></tr>
<tr><td>n+1</td><td colspan="3">Day (1 to 31)</td><td colspan="3">Hour (0 to 23)</td></tr>
<tr><td>n+2</td><td colspan="3">Minute (0 to 59)</td><td colspan="3">Second (0 to 59)</td></tr>
<tr><td>n+3</td><td colspan="3">Reserved</td><td colspan="2">Reserved</td><td>Weekday
(1 to 7, 1=Sun-day)</td></tr>
<tr><td>n+4 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
<tr><td>n+5 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
</table>

1) The two data words must exist in the data area to ensure that the data format matches WinCC flexible and to avoid reading false information.

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

Permitted data types

You can use the following data types when you configure the "Date/Time PLC" area pointer:

- DTL

Use of the "DTL" data type

Use of data type "DTL" with communication driver S7-1200. A tag of the "DTL" data type has a length of 12 bytes and saves information on date and time in a predefined structure.

The "DTL" data type has the following structure:

Byte	Component	Data type	Value range
0	Year	UINT	1970 to 2554
1			
2	Month	USINT	0 to 12
3	Day	USINT	1 to 31
4	Day of week	USINT	1(Sunday) to 7(Saturday) The weekday is not considered in the value entry.
5	Hour	USINT	0 to 23
6	Minute	USINT	0 to 59
7	Second	USINT	0 to 59

Byte	Component	Data type	Value range
8	Nanoseconds	UDINT	0 to 999 999 999
9			
10			
11			

The HMI devices do not support the use of nanoseconds. Values in the nanosecond range will be ignored during processing in Runtime.

The "DTL" data type supports time information down to the nanosecond range. Because panels only support time information down to the millisecond, you may encounter the following restriction when using the area pointers:

For the transmission of time information from a controller to a panel, the range from microseconds to nanoseconds is ignored. The time information is processed on the panel down to milliseconds.

Area pointer "Coordination" (Basic Panels)

Function

The "Coordination" area pointer is used to implement the following functions:

- Detecting the startup of the HMI device in the control program
- Detecting the current operating mode of the HMI device in the control program
- Detecting whether the HMI device is ready to communicate in the control program

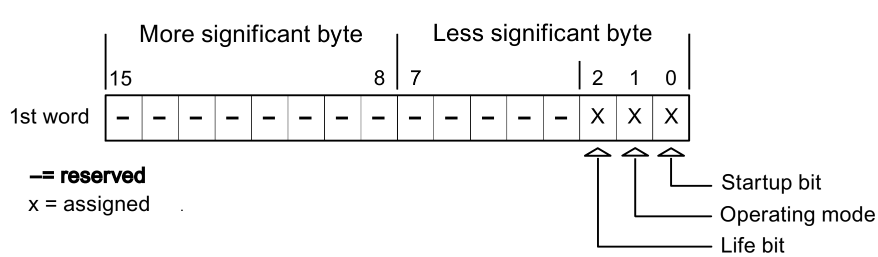
By default, the "Coordination" area pointer has the length of one word and cannot be changed.

Usage

Note

The HMI device always writes the entire coordination area when updating the area pointer. The control program can for this reason not make changes to the coordination area.

Assignment of the bits in the "Coordination" area pointer



Startup bit

The startup bit is set briefly to "0" by the HMI device during startup. It sets the bit permanently to "1" when startup is completed.

Operating mode

The operating mode bit is set to 1 as soon as the user switches the HMI device offline. The status of the operating mode bit is "0" during normal operation of the HMI device. You can determine the current operating mode of the HMI device by reading this bit in the control program.

Life bit

The HMI device inverts the life bit at intervals of approximately one second. You can check whether or not there is still a connection to the HMI device by querying this bit in the control program.

Processing in the PLC

For a simpler evaluation in the PLC program, use a Bool array for this area pointer when using the SIMATIC S7 1200 communication driver. You will have to map the complete 16-bit word of the area pointer. Configure a tag of the data type "Array [0 .. 15] of bool" for this purpose.

Permitted data types

You can use the following data types when you configure the "Coordination" area pointer.

- Word
- UInt
- Bool

Area pointer "Screen number" (Basic Panels)

Function

The HMI devices store information about the screen called up on the HMI device in the "Screen number" area pointer.

This allows the transfer of the current screen contents from the HMI device to the PLC. Certain reactions can be triggered in the PLC, such as the call of a different screen.

Use

Before the "Screen number" area pointer can be used, it must be set up and activated by selecting "Communication ▸ Area pointer". You can create only **one** instance of the "Screen number" area pointer and only on **one** PLC.

The screen number is always transferred to the PLC when a new screen is activated or when the focus within a screen changes from one screen object to another.

Structure

The area pointer is a data area in the memory of the PLC with a fixed length of 5 words.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1. Word	Current screen type															
2. Word	Current screen number															
3. Word	Reserved															
4th word	Current field number															
5. Word	Reserved															

- Current screen type
  "1" for root screen or
  "4" for permanent window
- Current screen number
  1 to 32767
- Current field number
  1 to 32767

Note

Device dependency

Permanent windows are not available on Basic Panels.

Permitted data types

You can use the following data types when you configure the "Screen number" area pointer.

- Word
- UInt

Area pointer "Project ID" (Basic Panels)

Function

When Runtime starts, a check can be carried out as to whether the HMI device is connected to the correct PLC. This check is important when operating with several HMI devices.

For this, the HMI device compares a value stored on the PLC with the value specified in configuration. This ensures compatibility of the configuration data with the PLC program.

A missing compatibility results in a corresponding alarm and Runtime will not be started.

Use

Note

HMI connections cannot be switched "online".

The HMI connection in which the "Project ID" area pointer is used must be switched "online".

To use this area pointer, set up the following during the configuration:

- Define the version of the configuration. Values between 1 and 255 are possible.
  You enter the version in the editor "Runtime settings > General" in the "Identification" area.
- This is where you select the PLC tag or the tag array that you have configured as the data area for the area pointer.

Connection failure

A connection failure to a device on which the "project ID" area pointer is configured results in all the other connections of the device being switched to "offline".

This behavior has the following requirements:

- You have configured several connections in a project.
- You are using the "project ID" area pointer in at least one connection.

Causes which may set connections "offline":

- The PLC is not available.
- The connection has been switched offline in the engineering system.

Permitted data types

You can use the following data types when you configure the "Project ID" area pointer.

- Word
- UInt

Area pointer "Job mailbox" (Basic Panels)

Function

The PLC can use the job mailbox to transfer jobs to the HMI device to trigger corresponding actions on the HMI device. These functions include, for example:

- Display screen
- Set date and time

Data structure

The first word of the job mailbox contains the job number. Depending on the job mailbox, up to three parameters can be transferred.

Word	Most significant byte	Least significant byte
n+0	0	Job number
n+1	Parameter 1	
n+2	Parameter 2	
n+3	Parameter 3	

The HMI device evaluates the job mailbox if the first word of this job is not equal to zero. This means that the parameters must be entered in the job mailbox first, followed by the job number.

When the HMI device accepts the job mailbox, the first word is set to 0 again. The execution of the job mailbox is generally not completed at this point in time.

Job mailboxes

All job mailboxes and their parameters are listed below. The "No." column contains the job number of the job mailbox. Job mailboxes can only be triggered by the PLC when the HMI device is online.

No.	Function	
14	**Set time (BCD-coded)**	
	Parameter 1	Left byte: - Right byte: hours (0-23)
	Parameter 2	Left byte: minutes (0-59) Right byte: seconds (0-59)
	Parameter 3	-
15	**Set date (BCD-coded)**	
	Parameter 1	Left byte: - Right byte: weekday (1-7: Sunday-Saturday)
	Parameter 2	Left byte: day (1-31) Right byte: month (1-12)
	Parameter 3	Left byte: year
23	**User logon**	
	Logs the user on with the name "PLC user" at the HMI device with the group number transferred in Parameter 1. The logon is possible only when the transferred group number exists in the project.	
	Parameter 1	Group number 1 to 255
	Parameter 2, 3	-
24	**User logoff**	
	Logs off the current user. (The function corresponds to the "logoff" system function)	
	Parameter 1, 2, 3	-
40	**Transfer date/time to PLC**	

No.	Function	
14	**Set time (BCD-coded)**	
	An interval of at least 5 seconds must be maintained between two successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
41	**Transfer date/time to PLC**	
	An interval of at least 5 seconds must be maintained between two successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
46	**Update tag**	
	Causes the HMI device to read the current value of the tags from the PLC whose update ID matches the value transferred in Parameter 1. (Function corresponds to the "UpdateTag" system function.)	
	Parameter 1	1 - 100
49	**Clear event buffer**	
	Deletes all analog alarms and discrete alarms of the "Warnings" class from the alarm buffer.	
	Parameter 1, 2, 3	-
50	**Clear error alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Errors" class from the alarm buffer.	
	Parameter 1, 2, 3	-
51	**Display selection**	
	Parameter 1	Screen number
	Parameter 2	-
	Parameter 3	Field number
69	**Reading data record from PLC** 1)	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	0: Do not overwrite existing data record 1: Overwrite existing data record
70	**Writing data record to PLC** 1)	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	-

1)	Only devices supporting recipes
2)	OP 73, OP 77A and TP 177A HMI devices also execute the "Screen selection" job mailbox if the on-screen keyboard is active.
3)	The weekday is ignored on HMI device KTP 600 BASIC PN.

Permitted data types

You can use the following data types when you configure the "Screen number" area pointer:

- Word
- UInt

"Data record" area pointer (Basic Panels)

"Data mailbox" area pointer (Basic Panels)

Function

When data records are transferred between the HMI device and PLC, both partners access common communications areas on the PLC.

Data transfer types

There are two ways of transferring data records between the HMI device and PLC:

- Transfer without synchronization
- Transfer with synchronization over the data mailbox

Data records are always transferred directly. That is, the tag values are read from an address or written to an address configured for this tag directly, without redirecting the values by means of interim memory.

Initiating the transfer of data records

There are three ways of triggering the transfer:

- Operator input in the recipe view
- Job mailboxes
  The transfer of data records can also be triggered by the PLC.
- Triggering by configured functions

If the transfer of data records is triggered by a job mailbox, the data in the recipe view will be updated as well. Avoid operating the recipe view while job mailboxes for transfer of data records are being triggered. If you have already started editing a data record and a job mailbox is triggered for transfer of data records, then this job mailbox will be rejected.

Permitted data types

You can use the following data types when you configure the "Data record" area pointer.

- Word
- UInt

Transfer without synchronization (Basic Panels)

If you select asynchronous transfer of data records between the HMI device and PLC, there is no coordination over the common data areas. It is therefore unnecessary to set up a data area during configuration.

Asynchronous data record transfer can be a useful alternative, for example, when:

- The system is capable of excluding the risk of uncontrolled overwriting of data by the communication peer.
- The PLC does not require information about the recipe number and data record number.
- The transfer of data records is triggered by the operator of the HMI device.

Reading values

When a read job is triggered, the values are read from the PLC addresses and transferred to the HMI device.

- Triggering by the operator in the recipe view:
  The values are downloaded to the HMI device. You can then process, edit, or save these values, for example.
- Triggering by a function or job mailbox:
  The values are saved immediately to the data volume.

Writing values

When a write job is triggered, the values are written to the PLC addresses.

- Triggering by the operator in the recipe view:
  The current values are written to the PLC.
- Triggering by a function or job mailbox:
  The current values are written to the PLC from the data medium.

Sequence of a transfer started by the operator in the recipe display (Basic Panels)

Reading from the PLC started by the operator in the recipe view

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe number to be read and the status "Transferring" in the data mailbox and sets the data record number to 0.	Abort with system alarm.
3	The HMI device reads the values from the PLC and displays them in the recipe view. If the recipes have synchronized tags, the values from the PLC are also written to the tags.	

Step	Action	
4	The HMI device sets the status "Transfer completed."	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC started by the operator in the recipe view

Step	Action	
	Check: Status word = 0?	
1	Yes	No
	The HMI device enters the recipe and data record number to be written and the status "Transferring" in the data mailbox.	Abort with system alarm.
2	The HMI device writes the current values to the PLC. If the recipes have synchronized tags, the changed values are synchronized between the recipe view and tags and then written to the PLC.	
3	The HMI device sets the status "Transfer completed."	
4	If required, the control program can now evaluate the transferred data.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Sequence of the transfer triggered by a job mailbox (Basic Panels)

The transfer of data records between the HMI device and the PLC can be initiated by either one of these stations.

The two job mailboxes No. 69 and No. 70 are available for this type of transfer.

No. 69: Read data record from PLC ("PLC → DAT")

Job mailbox no. 69 transfers data mailboxes from the PLC to the HMI device. The job mailbox is structured as follows:

	Most significant byte	Least significant byte
Word 1	0	69
Word 2	Recipe number (1-999)	
Word 3	Data record number (1-65,535)	
Word 4	Do not overwrite existing data record: 0 Overwrite existing data record: 1	

No. 70: Write data record to PLC ("DAT → PLC")

Job mailbox no. 70 transfers data mailboxes from the HMI device to the PLC. The job mailbox is structured as follows:

	Most significant byte	Least significant byte
Word 1	0	70
Word 2	Recipe number (1-999)	
Word 3	Data record number (1-65,535)	
Word 4	—	

Sequence when reading from the PLC with job mailbox "PLC → DAT" (no. 69)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data mailbox.	Abort without return message.
3	The HMI device reads the values and stores the values in the data record specified in the job mailbox.	
4	• If "Overwrite" was selected in the job, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed". • If "Do not overwrite" was selected in the job, and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Sequence writing to the PLC with job mailbox "DAT → PLC" (no. 70)

Step	Action	
1	Check: Status word = 0?	
	Yes	No

Step	Action	
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data mailbox.	Abort without return message.
3	The HMI device fetches the values of the data record specified in the job from the data medium and writes the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The PLC program can now evaluate the transferred data. To allow further transfers, the PLC program must set the status word to 0 again.	

Sequence of the transfer when triggered by a configured function (Basic Panels)

Reading from the PLC using a configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system alarm.
3	The HMI device reads the values from the PLC and stores them in the data record specified in the function.	
4	• If "Yes" was selected for the "Overwrite" function, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "No" was selected for the "Overwrite" function and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC by means of configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system alarm.
3	The HMI device fetches the values of the data record specified in the function from the data medium and transfers the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Possible causes of error when transferring data records (Basic Panels)

Possible causes of error

The section below shows possible error causes which lead to the cancellation of data record transfer:

- Tag address not set up on the PLC
- Overwriting data records not possible
- Recipe number does not exist
- Data record number does not exist

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Reaction to an aborted transfer due to errors

If the transfer of data records is aborted due to errors, the HMI device reacts as follows:

- Triggering by the operator in the recipe view
  Information in the status bar of the recipe view and output of system alarms
- Triggered by function
  Output of system alarms
- Triggering by job mailbox
  No return message on the HMI device

You can nonetheless evaluate the status of the transfer by querying the status word in the data mailbox.

Note

Availability for specific devices

Notes in the status bar of the recipe view are not available in Basic Panels.

1.7.4.2 Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

1.7.4.3 Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configure alarms

Several steps are needed to configure alarms, such as operational messages, error alarms, and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a SIMATIC communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
SIMATIC S7 PLCs	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

How the bit positions are counted

For connections with a SIMATIC communication driver, the following counting method applies:

How the bit positions are counted	Byte 0								Byte 1							
	Most significant byte								Least significant byte							
In SIMATIC S7 PLCs	7							0	7							0
In WinCC you configure:	15							8	7							0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

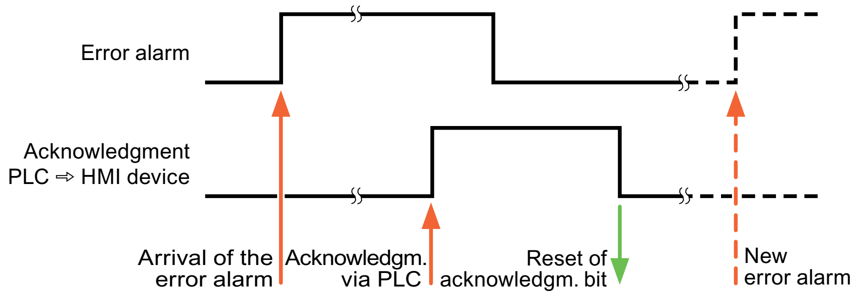
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

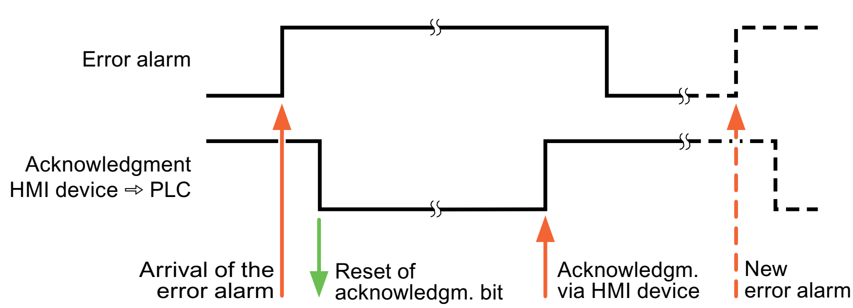
If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.7.5 Performance features of communication (Basic Panels)

1.7.5.1 S7-1200 device dependency (Basic Panels)

Communication with the SIMATIC S7-1200 controller

If you use devices from an earlier version of the TIA Portal with TIA Portal V13, it may not be possible to configure integrated connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KP300 Basic	Yes	Yes	Yes	Yes	Yes
KP400 Basic	Yes	Yes	Yes	Yes	Yes
KTP400 Basic PN	Yes	Yes	Yes	Yes	Yes
KTP600 Basic DP	Yes	Yes	Yes	Yes	Yes
KTP600 Basic PN	Yes	Yes	Yes	Yes	Yes
KTP1000 Basic DP	Yes	Yes	Yes	Yes	Yes
KTP1000 Basic PN	Yes	Yes	Yes	Yes	Yes
TP1500 Basic PN	Yes	Yes	Yes	Yes	Yes

Basic Panels V12.0

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KP300 Basic	No	Yes	Yes	Yes	Yes
KP400 Basic	No	Yes	Yes	Yes	Yes
KTP400 Basic PN	No	Yes	Yes	Yes	Yes
KTP600 Basic DP	No	Yes	Yes	Yes	Yes
KTP600 Basic PN	No	Yes	Yes	Yes	Yes
KTP1000 Basic DP	No	Yes	Yes	Yes	Yes
KTP1000 Basic PN	No	Yes	Yes	Yes	Yes
TP1500 Basic PN	No	Yes	Yes	Yes	Yes

Basic Panels V13.0

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KTP400 Basic PN	No	Yes	Yes	Yes	Yes
KTP700 Basic PN	No	Yes	Yes	Yes	Yes
KTP700 Basic DP	No	Yes	Yes	Yes	Yes
KTP900 Basic PN	No	Yes	Yes	Yes	Yes
KTP1200 Basic PN	No	Yes	Yes	Yes	Yes
KTP1200 Basic DP	No	Yes	Yes	Yes	Yes

Basic Panels V13.0.1

HMI devices	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KTP400 Basic PN	No	Yes	Yes	Yes	Yes
KTP700 Basic PN	No	Yes	Yes	Yes	Yes
KTP700 Basic DP	No	Yes	Yes	Yes	Yes
KTP900 Basic PN	No	Yes	Yes	Yes	Yes
KTP1200 Basic PN	No	Yes	Yes	Yes	Yes
KTP1200 Basic DP	No	Yes	Yes	Yes	Yes

1.7.5.2 Valid data types for SIMATIC S7-1200 (Basic Panels)

Valid data types for connections with SIMATIC S7-1200

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Length
BOOL	1 bit
SINT	1 byte
INT	2 bytes
DINT	4 bytes
USINT	1 byte
UINT	2 bytes
UDINT	4 bytes
REAL	4 bytes
LREAL	8 bytes
TIME	4 bytes
DATE	2 bytes

Data type	Length	
DTL	12 bytes	Basic Panels, Panels, RT Advanced
	8 bytes	RT Professional
TIME_OF_DAY, TOD	4 bytes	
STRING	(2+n) bytes, n = 0 to 254	
WSTRING	(4+2*n) bytes, n = 0 to 254	Basic Panels
	(4+2*n) bytes, n = 0 to 4094	Panels, RT Advanced
	(4+2*n) bytes, n = 0 to 65534	RT Professional
CHAR	1 byte	
Array of CHAR	--	
BYTE	1 byte	
WORD	2 bytes	
DWORD	4 bytes	
LDT	8 bytes	RT Professional
DATE_AND_TIME	8 bytes	RT Professional

1.7.6 Creating connections in the "Connections" editor (Basic Panels)

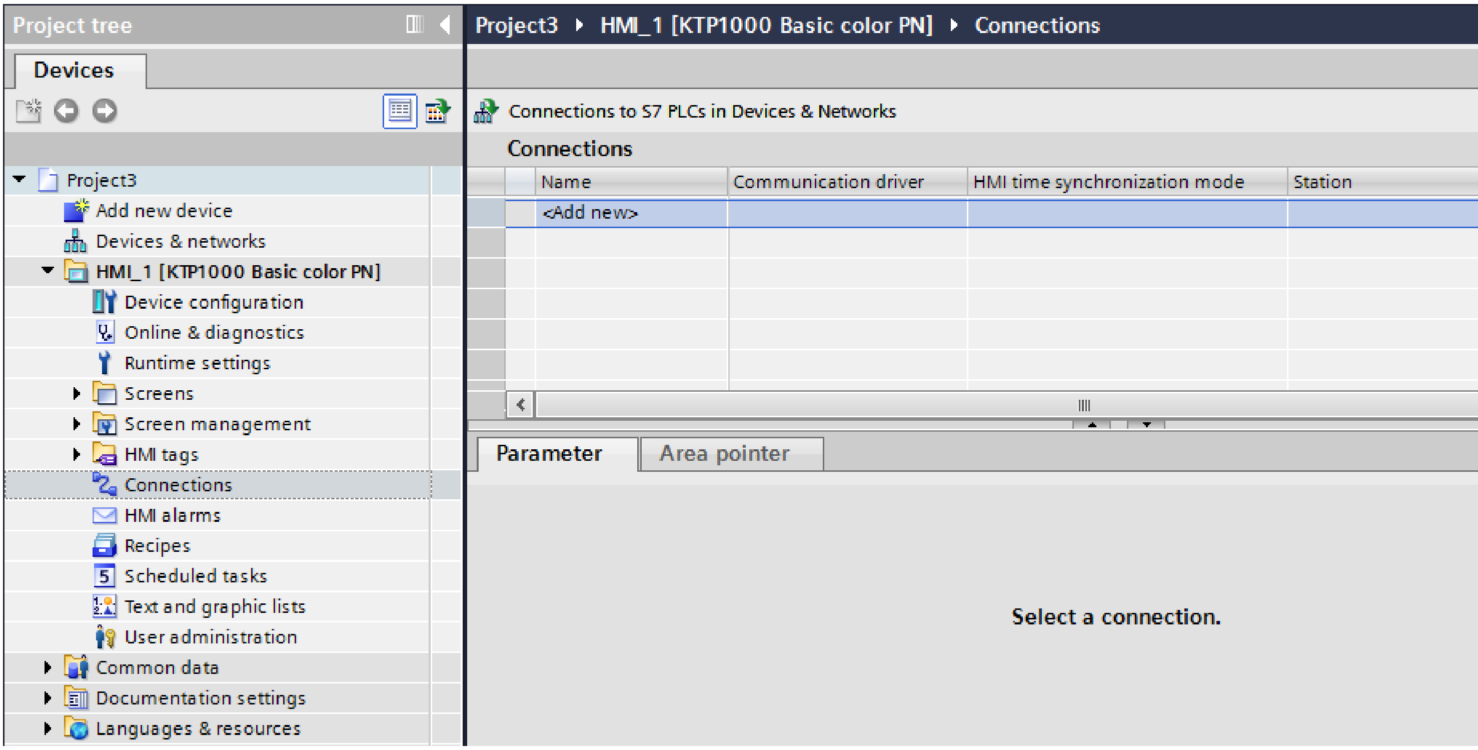
1.7.6.1 Creating a PROFINET connection (Basic Panels)

Requirements

- A project is open.
- An HMI device with a PROFINET interface has been created.

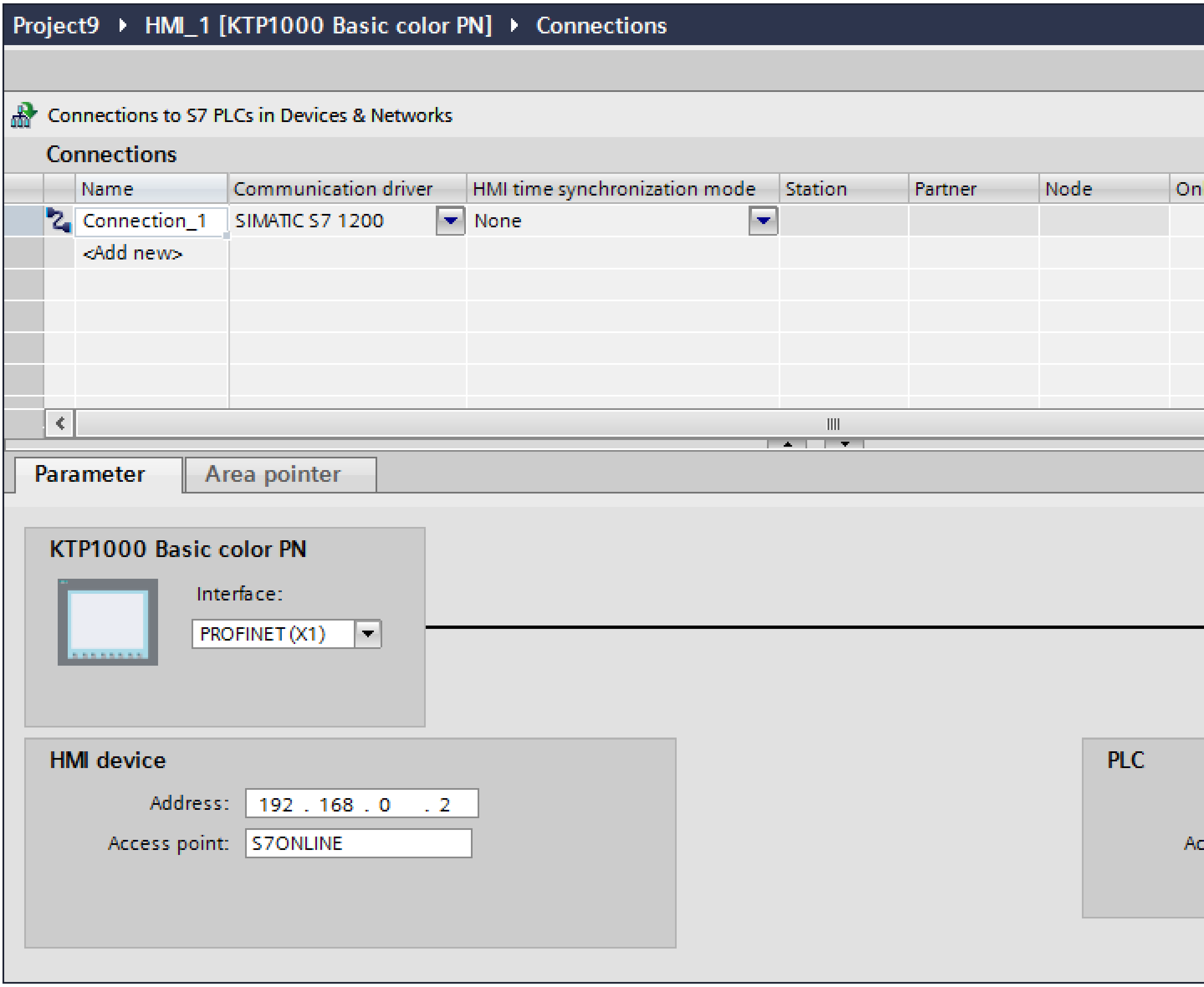
Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. In the "Communication drivers" column, select the "SIMATIC S7 1200" driver.
4. Click the name of the connection.

5. Select a PROFINET interface of the HMI device in the Inspector window under "Parameters > Interface".



6. Set the IP addresses of the communication partners in the Inspector window:
   – HMI device: "Parameters > HMI device > Address"
   – PLC: "Parameters > PLC > Address"

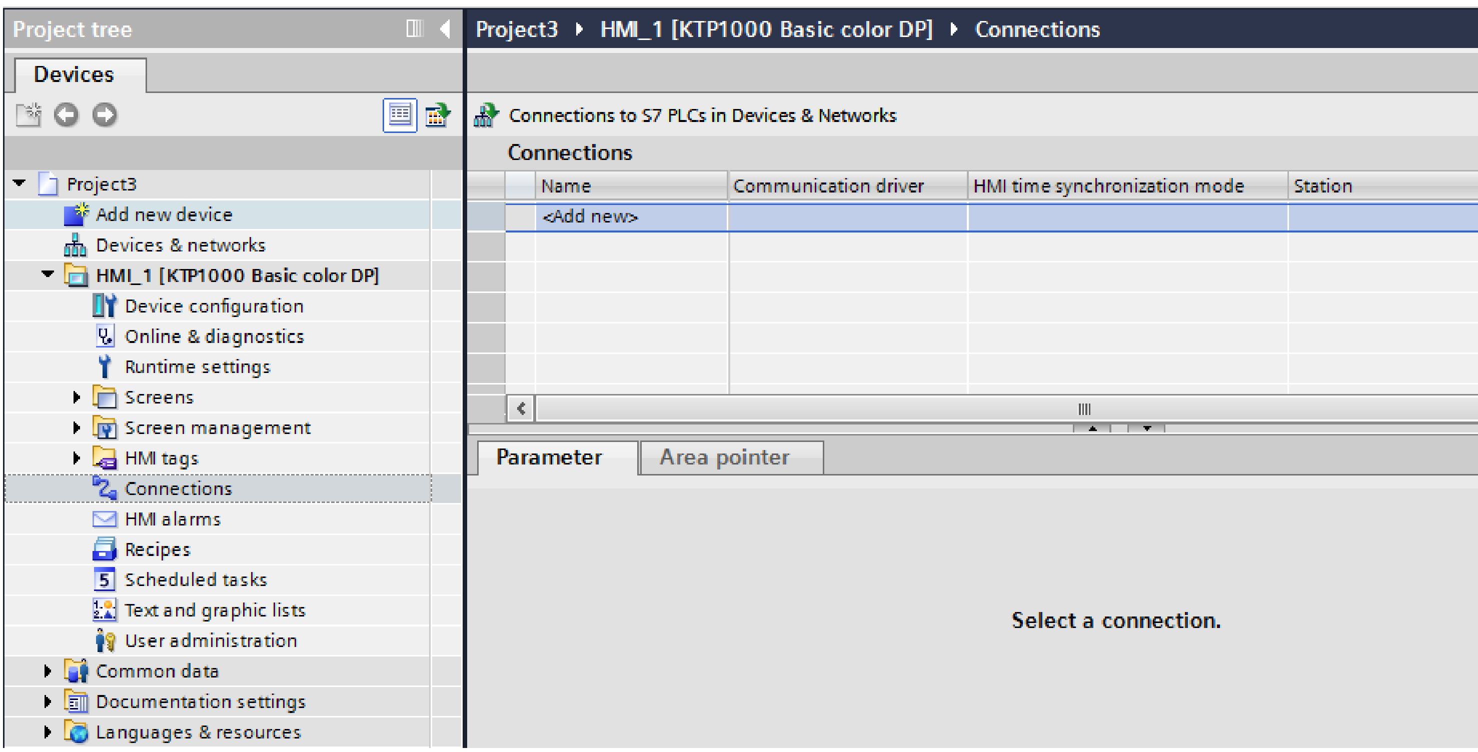
1.7.6.2 Creating a PROFIBUS DP connection (Basic Panels)

Requirements

- A project is open.
- An HMI device with a PROFIBUS interface has been created.

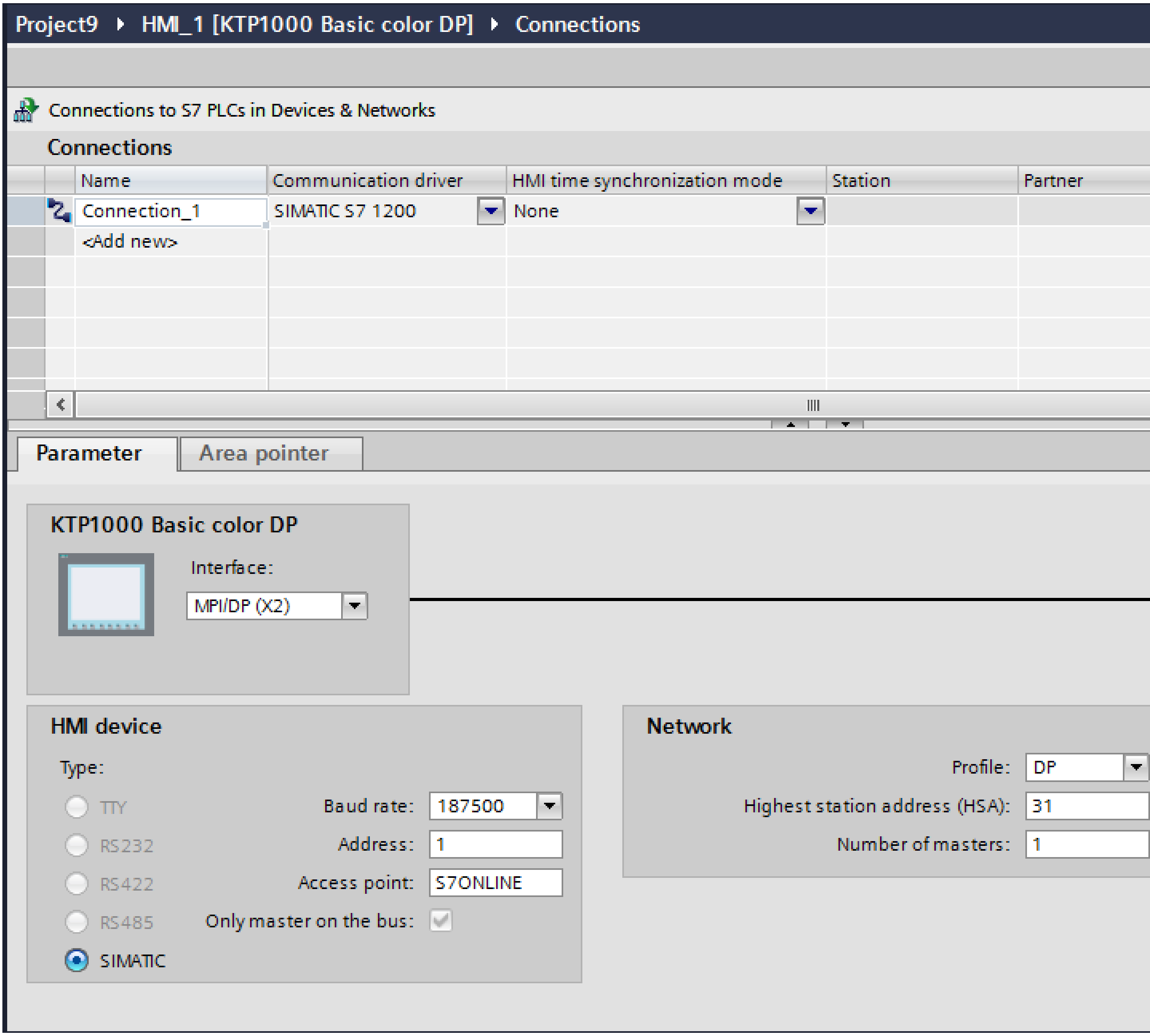
Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. In the "Communication drivers" column, select the "SIMATIC S7 1200" driver.
4. Click the name of the connection.
5. Select the "MPI/DP" interface in the Inspector window under "Parameters".

6. Select the "DP" profile in the Inspector window under "Parameters > Network".



7. Set the addresses of the communication partners in the Inspector window:
   - HMI device: "Parameters > HMI device > Address"
   - PLC: "Parameters > PLC > Address"

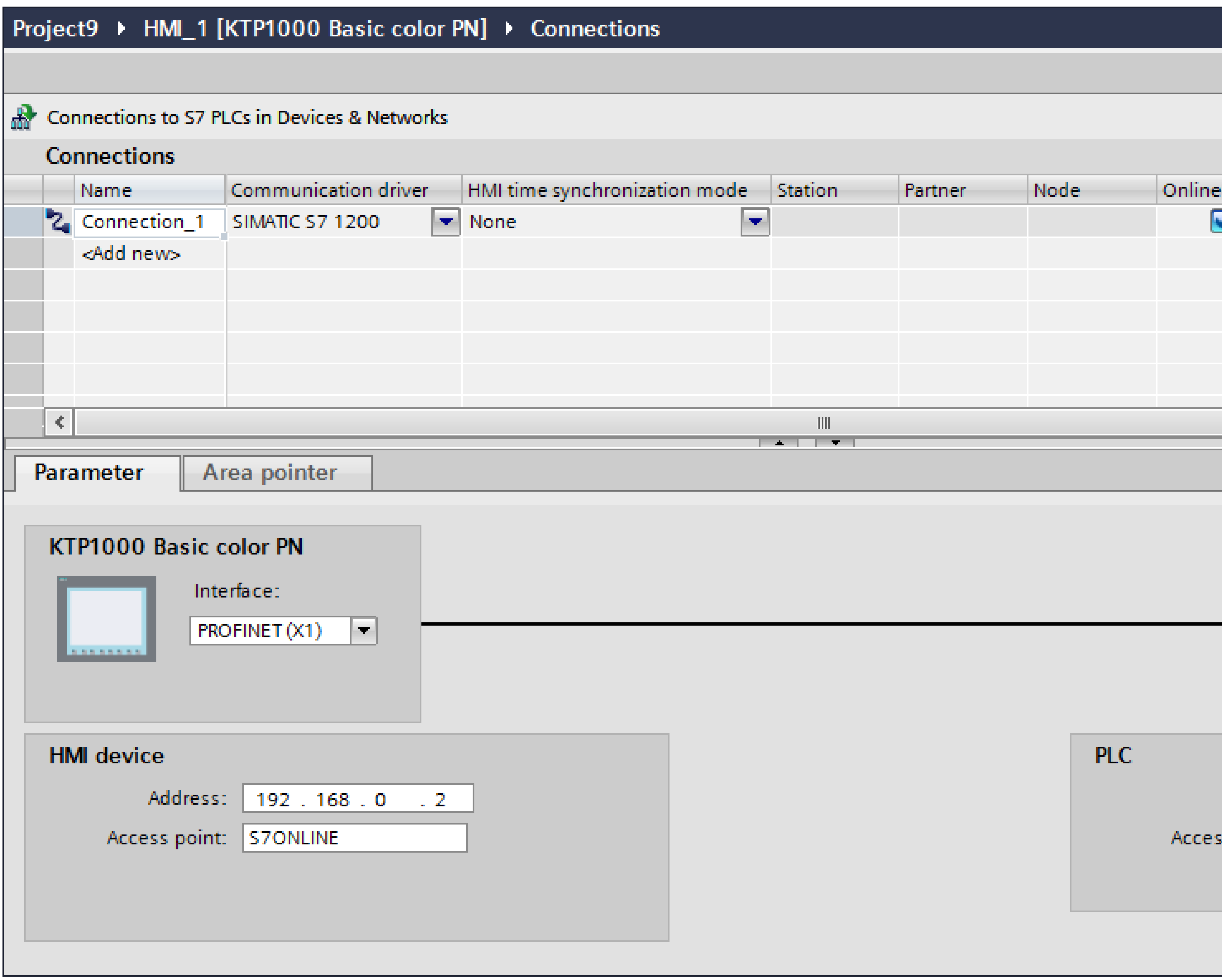
1.7.6.3 Parameters for the connection (Basic Panels)

Parameters for the connection (SIMATIC S7 1200) (Basic Panels)

Parameters to be set

To assign the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device", "Network", and "PLC" areas are available for assigning parameters according to the interface used.



Ethernet parameters (Basic Panels)

Parameters for the HMI device

You set the parameters for the HMI device in the network under "HMI device".. The changes are not transferred automatically to the HMI device. You must change the settings in the Control Panel of the HMI device.

- "Interface"
  If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC.

  Note

  The IP address in the Control Panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

  The IP address already set up in the Control Panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

  The IP address is transferred to the HMI device during project transfer.
  To set up the IP address of the HMI device:

  – Click the HMI device.

  – Open the "Device configuration" editor.

  – Click the Ethernet interface.

  – Assign the IP address in the inspector window under:
  "General > PROFINET interface > Ethernet addresses"

- "Address"
  You assign the IP address of the HMI device in the "Address" area.
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

- "Access point"
  Specifies the access point for the PG/PC interface that can be used to reach the communication partner.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  Under "Address", set the IP address of the S7 module to which the HMI device is connected.
- "Access password"
  Enter a password in the "Access password" field. This password must match the one you saved to the PLC.

Note

You only need a password if you have set "Complete protection" at the PLC.

No connection is set up to the PLC if the "Complete protection" security level is stored on the PLC and you do not enter a password.

PROFIBUS parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the PROFIBUS network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

Note

If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  You set the PROFIBUS address of the HMI device under "Address". The PROFIBUS address must be unique in the PROFIBUS network.
- "Only master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master).
  In S7-200, you must set an HMI device as the master.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the network

Under "Network", you set the parameters for the PROFIBUS network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "DP", "Universal", or "Standard". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual PROFIBUS address. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud in the OP 73 or the OP 77A, the highest station address must be less than or equal 63.
- "Number of masters"
  For "Number of masters", set the number of masters in the PROFIBUS network. This information is necessary to correctly calculate the bus parameters.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the PROFIBUS address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Access password"
  Enter a password in the "Access password" field. This password must match the one you saved to the PLC.

  Note

  You only need a password if you have set "Complete protection" at the PLC.

  No connection is set up to the PLC if the "Complete protection" security level is stored on the PLC and you do not enter a password.

1.8 Communicating with SIMATIC S7 300/400 (Basic Panels)

1.8.1 Communication with SIMATIC S7 300/400 (Basic Panels)

Introduction

This section describes the communication between an HMI device and the SIMATIC S7 300 and S7 400 PLCs. These two PLCs will be referred to jointly as SIMATIC S7 300/400.

You can configure the following communication channels for the SIMATIC S7 300/400 PLC:

- PROFINET
- PROFIBUS
- MPI

HMI connection for communication

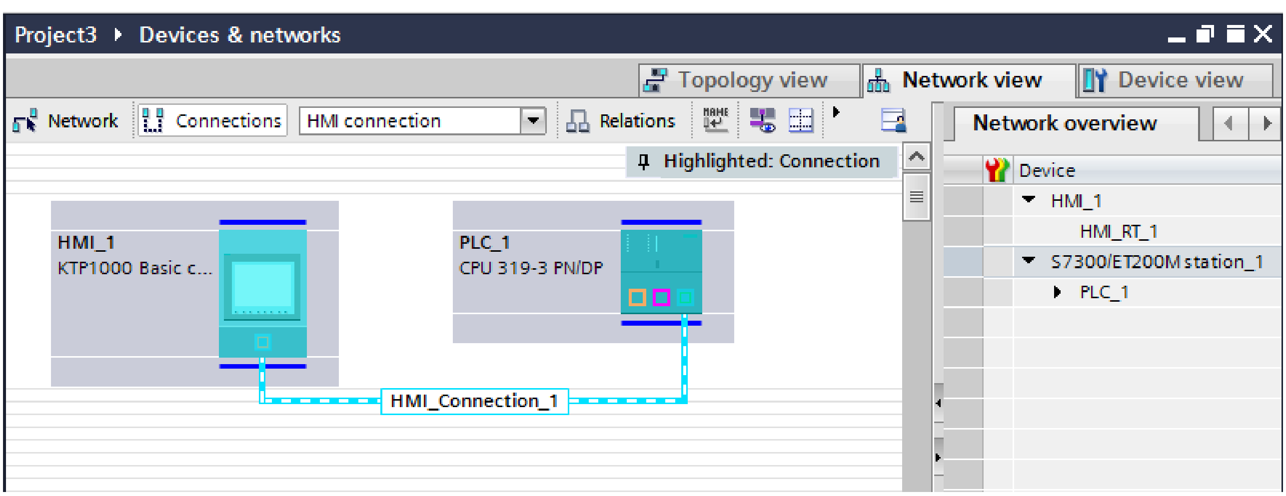
You configure connections between the HMI device and a SIMATIC S7 300/400 in the "Devices & Networks" editor.

1.8.2 Communication via PROFINET (Basic Panels)

1.8.2.1 Communication via PROFINET (Basic Panels)

HMI connections via PROFINET

If you have inserted an HMI device and a SIMATIC S7 300/400 into the project, you interconnect the two PROFINET interfaces in the "Devices & Networks" editor.



You can also connect multiple HMI devices to one SIMATIC S7 300/400 and multiple SIMATIC S7 300/400s to one HMI device. The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used.

Additional information is available in the documentation for the respective HMI device.

HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFINET in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device via PROFINET in the "Connections" editor of the HMI device.

1.8.2.2 Configuring an HMI connection via PROFINET (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC S7 300/400 via PROFINET or Ethernet in the "Devices & Networks" editor.

> **CAUTION**
>
> **Communication via Ethernet**
>
> In Ethernet-based communication, the end user is responsible for the security of his data network.
>
> Targeted attacks can overload the device and interfere with proper functioning.

Requirements

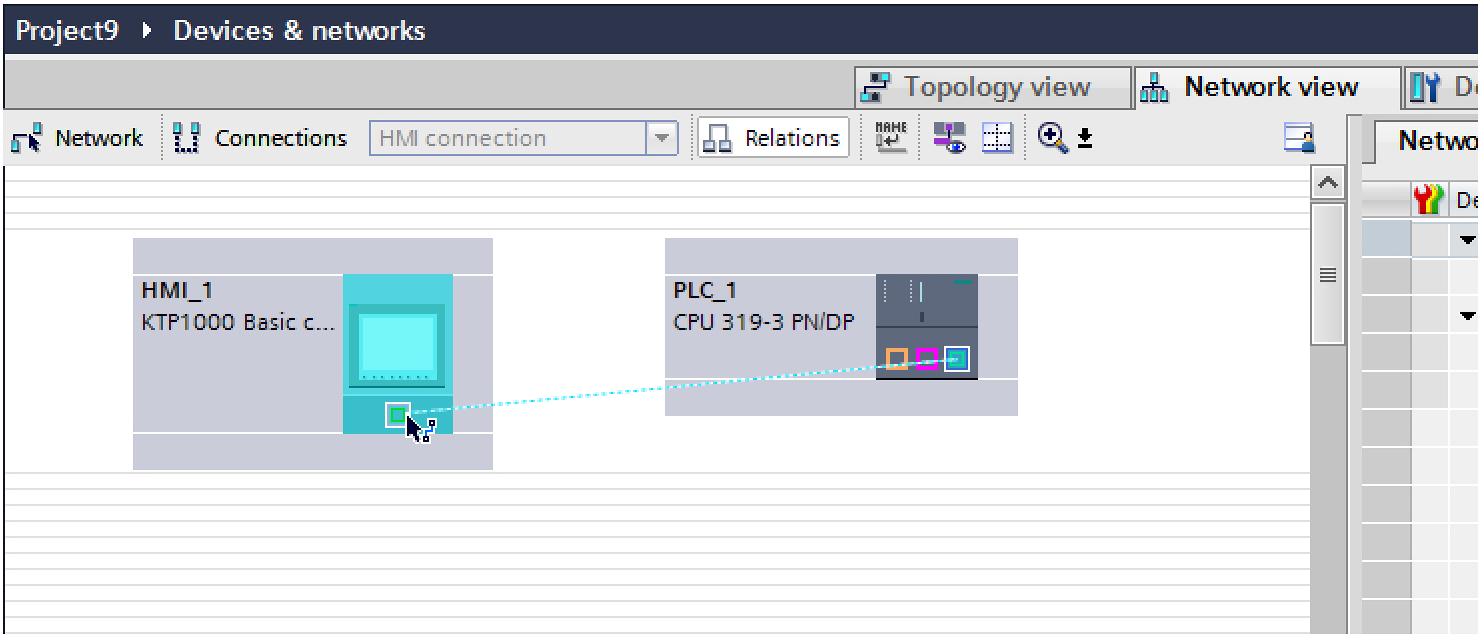
The following communication partners are created in the "Devices & Networks" editor:

- HMI device with PROFINET or Ethernet interface
- SIMATIC S7 300/400 with PROFINET interface.

Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button and select "HMI connection" for the connection type.
   The devices available for connection are highlighted in color.

3. Click the PROFINET interface of the PLC and use a drag-and-drop operation to draw a connection to the PROFINET or Ethernet interface of the HMI device.



4. Click the connecting line.
5. Click "Highlight HMI connection" and select the HMI connection.
   The connection is displayed graphically in the Inspector window.
6. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project.
   See the chapter "PROFINET parameters (Page 181)" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created a connection between an HMI device and a SIMATIC S7 300/400. The IP address and subnet mask connection parameters are configured.

1.8.2.3 PROFINET parameters (Basic Panels)

PROFINET parameters for the HMI connection (Basic Panels)

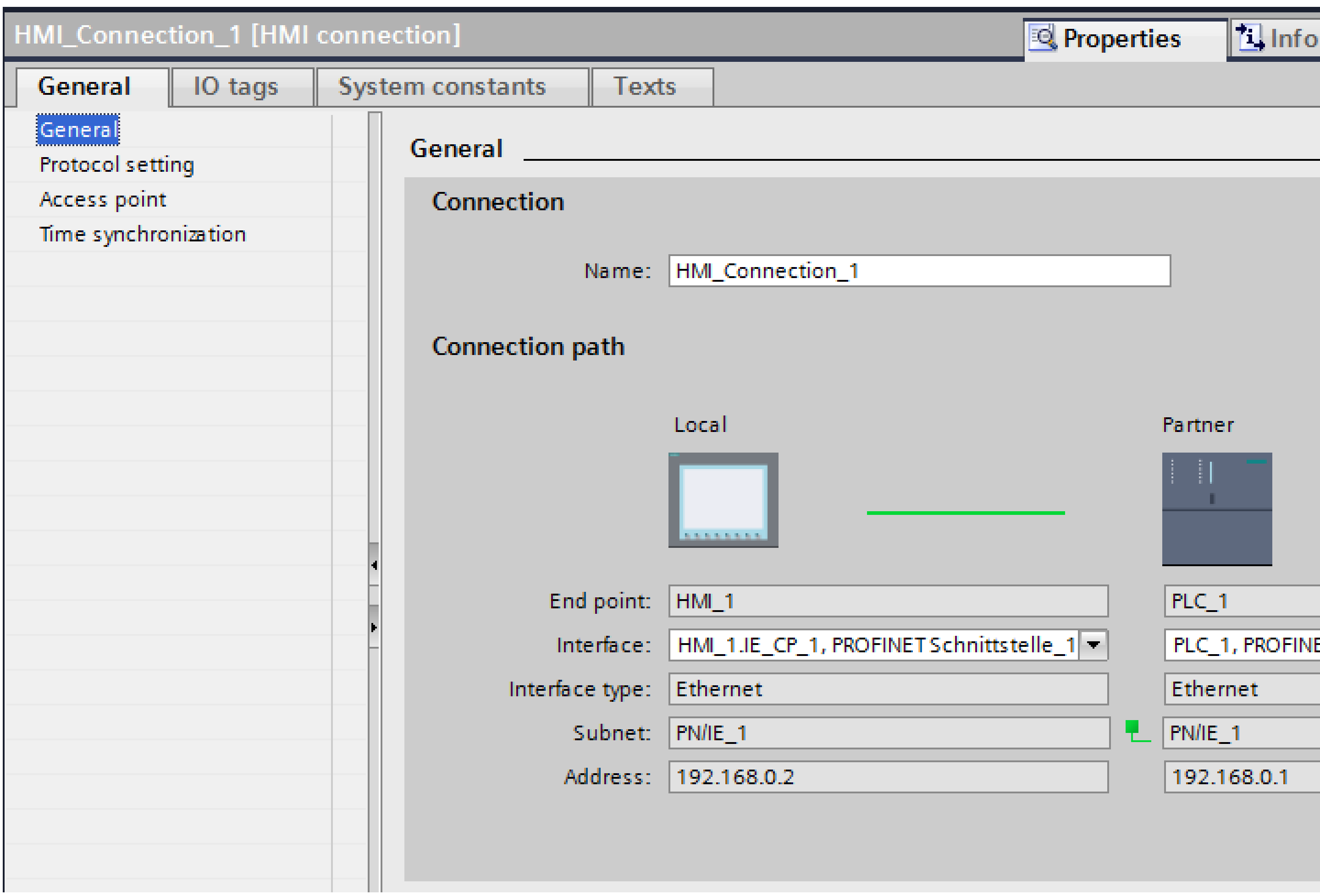
PROFINET parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and changing the HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

Displays whether the devices are networked together.

 - displayed if the devices are networked together.

 - displayed if the devices are not networked together.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFINET parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area cannot be edited.
- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the selected IP address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

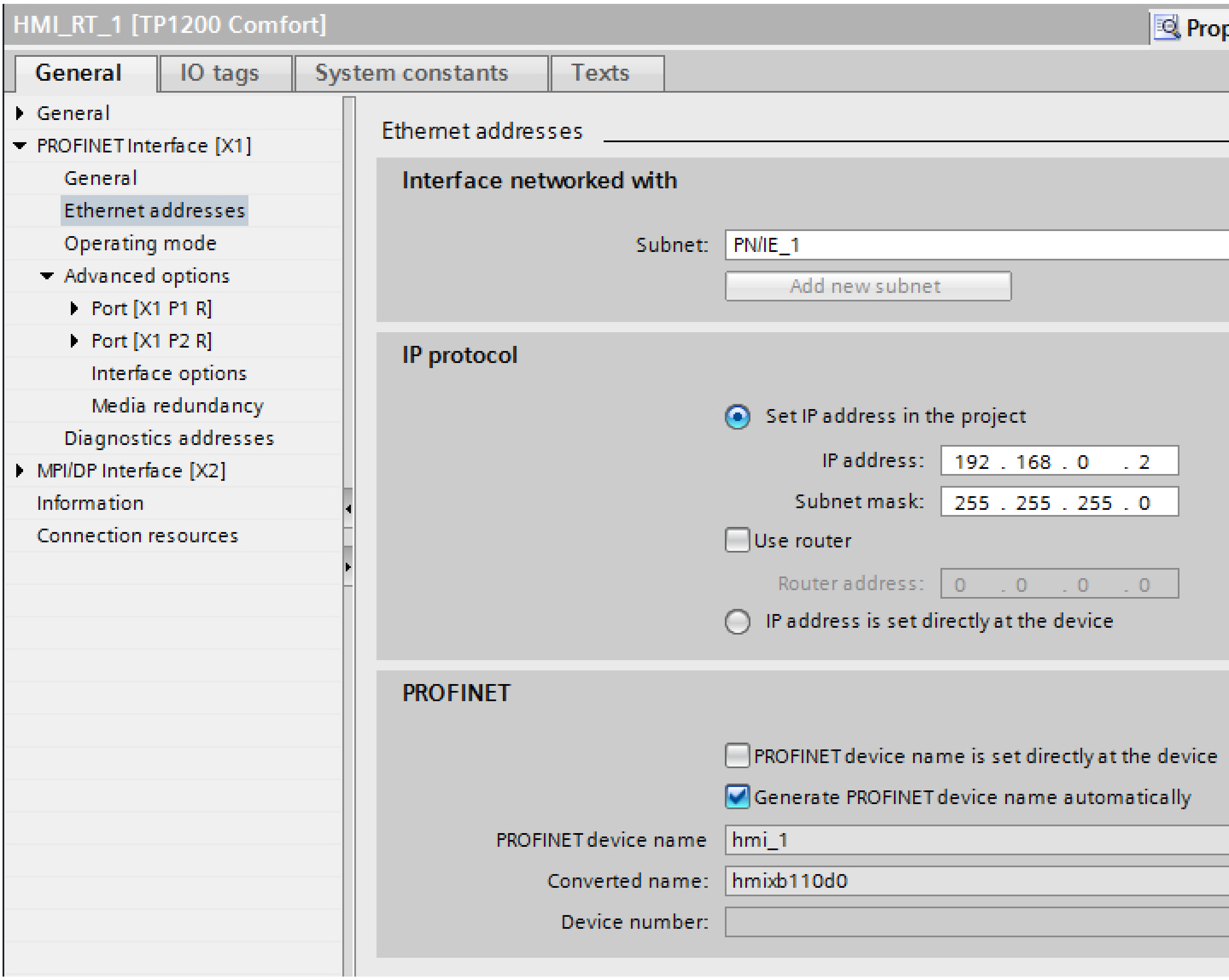
PROFINET parameters for the HMI device (Basic Panels)

PROFINET parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Set IP address in the project"
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

  Note

  The device is automatically restarted in the case of HMI devices with the Windows CE 3.0 operating system.

  HMI devices with Windows CE 3.0:

  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
- "Use IP router"
  If you are using an IP router, select "Use IP router" and enter the router address in the "Router address" field.
- "Set IP address using a different method"
  If the function "Set IP address using a different method" is activated, the IP address is not taken from the project. You have to enter the IP address directly in the Control Panel of the HMI device.

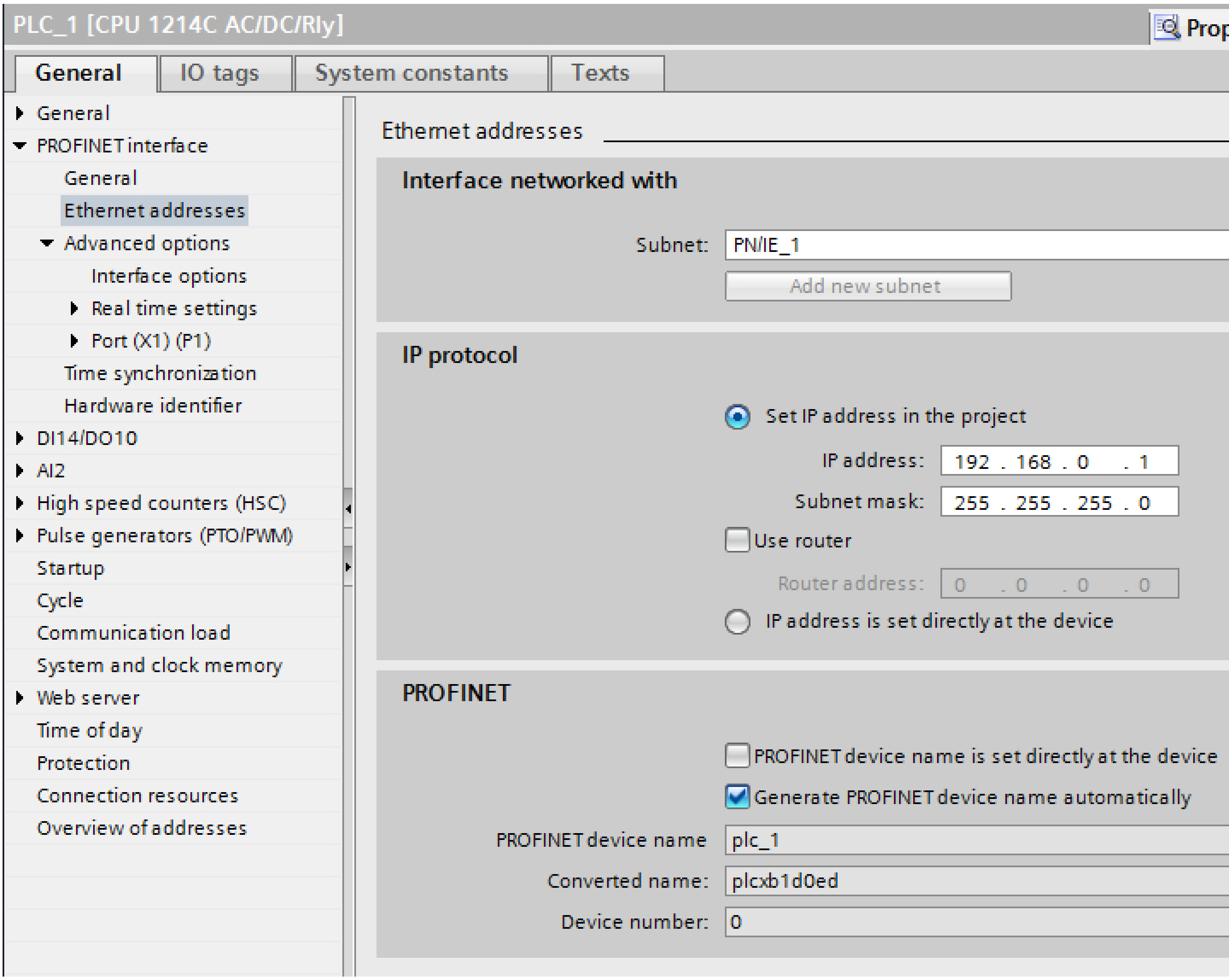
PROFINET parameters for the PLC (Basic Panels)

PROFINET parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFINET parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "IP address"
  You assign the IP address of the HMI device in the "IP address" area.
- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
  If you are using an IP router, select "Use IP router" and enter the router address in the field.

Configuring Industrial Ethernet (Basic Panels)

Rules for the network configuration

The Ethernet interfaces of the devices have a default IP address that you can change.

IP address

The IP parameters are visible if the communication-capable devices support the TCP/IP protocol.

The IP address consists of 4 decimal figures in the range of 0 to 255. The decimal figures are separated from one another by a dot.

Example: 140.80.0.2

The IP address consists of the following:

- The address of the (sub) net
- The address of the node (generally also called host or network node)

Subnet mask

The subnet mask splits these two addresses. It determines which part of the IP address addresses the network and which part of the IP address addresses the node.

The set bits of the subnet mask determine the network part of the IP address.

Example:

Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000

In the example given for the above IP address, the subnet mask shown here has the following meaning:

The first 2 bytes of the IP address identify the subnet - i.e. 140.80. The last two bytes address the node, thus 0.2.

It is generally true that:

- The network address results from AND linking the IP address and subnet mask.
- The node address results from AND NOT linking the IP address and subnet mask.

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. It determines 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 of between 224 and 255 is also possible for the first decimal number of the IP address (address class D etc). This is, however, 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 lower points of the subnet mask to "1". For each bit set to "1", the number of "private" networks doubles and the number of nodes they contain is halved. Externally, the network functions like an individual network as it did previously.

Example:

You have a subnet of address class B (e.g. 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 nodes with addresses between 129.80.001.xxx and 129.80.127.xxx are on one subnet, all nodes with addresses between 129.80.128.xxx and 129.80.255.xxx are on another subnet.

Router

The job of the routers is to connect the subnets. If an IP datagram is to be sent to another network, it first has to be conveyed to a router. To make this possible, in this case you have to enter the address of the router for each node in the subnet.

The IP address of a node in the subnet and the address of the router may only differ at the points at which there is a "0" in the subnet mask.

1.8.2.4 Setting port options (Basic Panels)

Setting the port options (Basic Panels)

Changing connection settings for the PROFINET IO port

You can change the network settings for the PROFINET IO port as required. By default, the settings are made automatically. In normal situations, this guarantees problem-free communication.

Possible settings for transmission rate / duplex

Depending on the selected device, you can make the following settings for "Transmission rate / duplex":

- Automatic setting
  Recommended default setting of the port. The transmission settings are automatically "negotiated" with the peer port. The "Enable autonegotiation" option is also enabled as a default, in other words, you can use cross cables or patch cables for the connection.
- TP/ITP at x Mbps full duplex (half duplex)
  Setting of the transmission rate and the full duplex/half duplex mode. The effectiveness depends on the "Enable autonegotiation" setting:
  – Autonegotiation enabled
    You can use both cross cable and patch cable.
  – Autonegotiation disabled
    Make sure that you use the correct cable (patch cable or cross cable)! The port is also monitored with this setting.
- Deactivated
  Depending on the module type, the drop down list box can contain the "- Disabled -" option. This option, for example, allows you to prevent access to an unused port for security reasons. With this setting, diagnostic events are not generated.

"Monitor" option

This option enables or disables port diagnostics. Examples of port diagnostics: The link status is monitored, in other words, the diagnostics are generated during link-down and the system reserve is monitored in the case of fiber optic ports.

Option "Enable autonegotiation "

The autonegotiation setting can only be changed if a concrete medium (for example, TP 100 Mbps full duplex) is selected. Whether or not a concrete medium can be set depends on the properties of the module.

If autonegotiation is disabled, this causes the port to be permanently specified, as for example, is necessary for a prioritized startup of the IO device.

You must make sure the partner port has the same settings because with this option the operating parameters of the connected network are not detected and the data transmission rate and transmission mode can accordingly not be optimally set.

Note

When a local port is connected, STEP 7 makes the setting for the partner port if the partner port supports the setting. If the partner port does not accept the setting, an error message is generated.

Wiring rules for disabled autonegotiation (Basic Panels)

Requirements

You have made the following settings for the port in question, for example, to accelerate the startup time of the IO device:

- Fixed transmission rate
- Autonegotiation incl. autocrossing disabled

The time for negotiating the transmission rate during startup has been saved.

If you have disabled autonegotiation, you must observe the wiring rules.

Wiring rules for disabled autonegotiation

PROFINET devices have the following two types of ports:

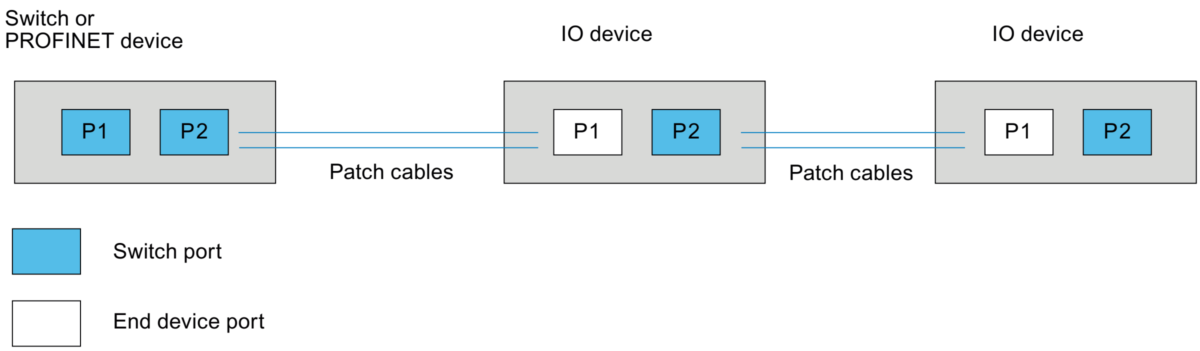
Type of port	PROFINET devices	Note
Switch port with crossed pin assignment	For IO devices: Port 2 For S7 CPUs with 2 ports: Ports 1 and 2	Crossed pin assignment means that the pin assignment for the ports for sending and receiving between the respective PROFINET devices is exchanged internally.
End device port with uncrossed pin assignment	For IO devices: Port 1 For S7 CPUs with one port: Port 1	-

Validity of the wiring rules

The cabling rules described in the following paragraph apply exclusively for the situation in which you have specified a fixed port setting.

Rules for cabling

You can connect several IO devices in line using a single cable type (patch cable). To do this, you connect port 2 of the IO device (distributed I/O) with port 1 of the next IO device. The following graphic gives an example with two IO devices.



Boundaries at the port (Basic Panels)

Requirements

To use boundaries, the respective device must have more than one port. If the PROFINET does not support boundary settings, they are not shown.

Enable boundaries

"Boundaries" are limits for transmission of certain Ethernet frames. The following boundaries can be set at a port:

- "End of discovery of accessible devices"
  No forwarding of DCP frames to identify accessible devices. Devices downstream from this port cannot be reached by the project tree under "Accessible devices". Devices downstream from this port cannot be reached by the CPU.
- "End of topology discovery"
  LLDP frames (Link Layer Discovery Protocol) are not forwarded for topology detection.
- "End of sync domain"
  No forwarding of sync frames transmitted to synchronize nodes within a sync domain.
  If you operate, for example, a PROFINET device with more than two ports in a ring, you should prevent the sync frame from being fed into the ring by setting a sync boundary (at the ports not inside the ring).
  Additional example: If you want to use several sync domains, configure a sync domain boundary for the port connected to a PROFINET device from the other sync domain.

Restrictions

The following restrictions must be observed:

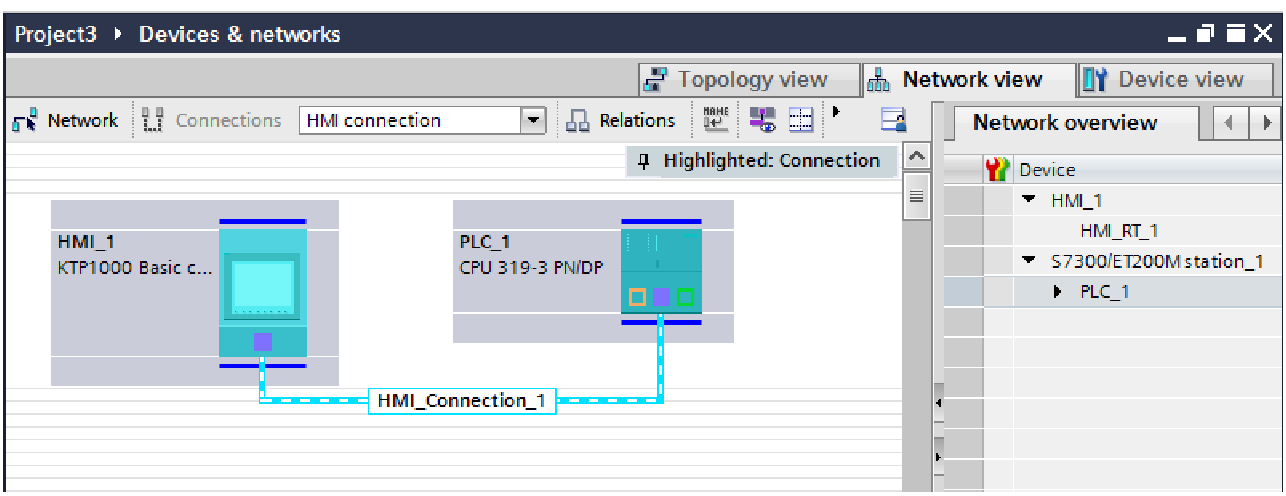
- The individual check boxes can only be used if the port supports the function in question.
- If a partner port has been determined for the port, the following check boxes cannot be used:
  - "End of discovery of accessible devices"
  - "End of topology discovery"
- If autonegotiation is disabled, none of the check boxes can be used.

1.8.3 Communication via PROFIBUS (Basic Panels)

1.8.3.1 Communication via PROFIBUS (Basic Panels)

HMI connections via PROFIBUS

If you have inserted an HMI device and a SIMATIC S7 300/400 into the project, you interconnect the two PROFIBUS interfaces in the "Devices & Networks" editor.



HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFIBUS in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device in the "Connections" editor of the HMI device.

1.8.3.2 Configuring an HMI connection via PROFIBUS (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC S7 300/400 via PROFIBUS in the "Devices & Networks" editor.

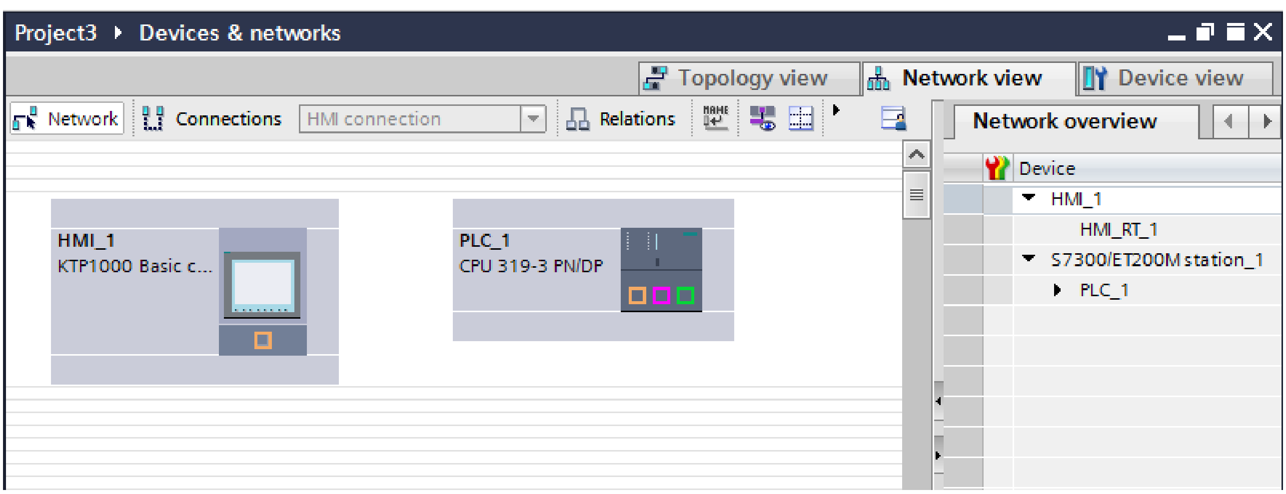
Requirements

The following communication partners are created in the "Devices & Networks" editor:

- HMI device with MPI/DP interface
- SIMATIC S7 300/400 with PROFIBUS interface

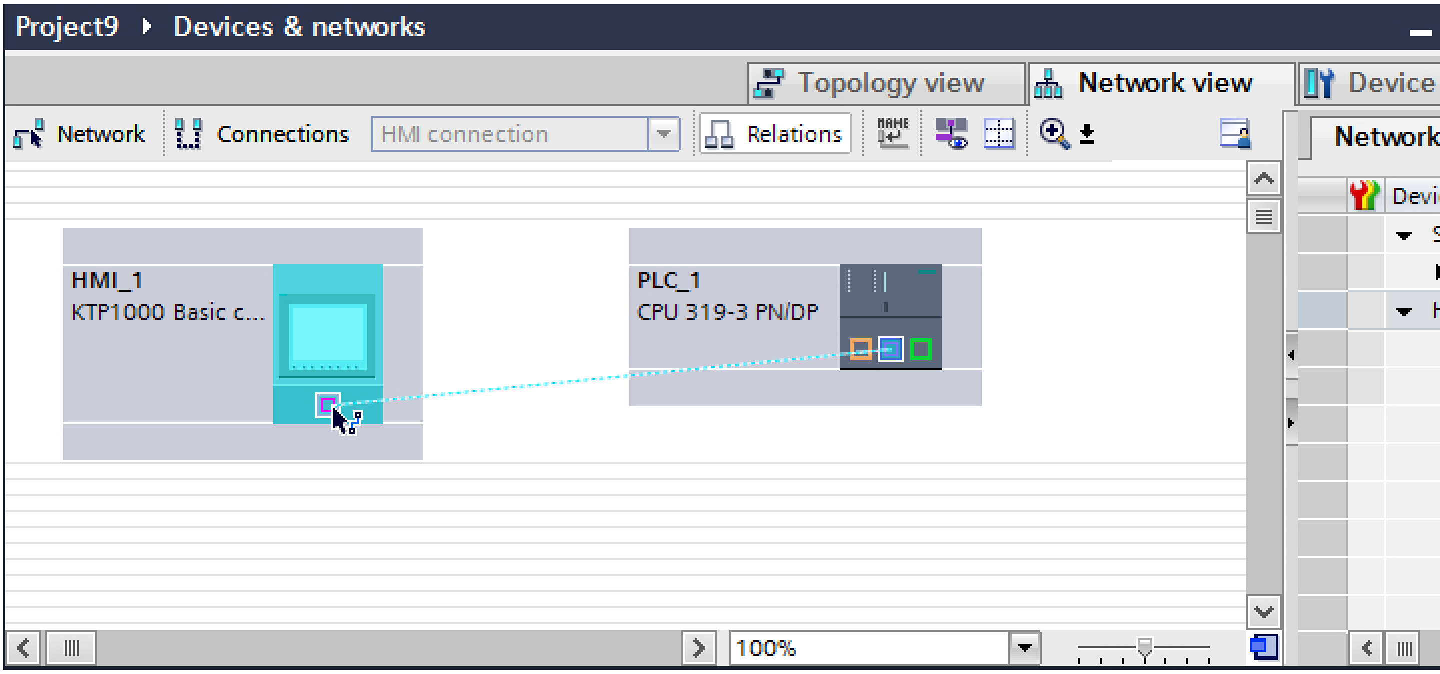
Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.



2. Click the "Connections" button.
   The devices available for connection are highlighted in color.
3. Click the HMI device interface.
4. Select the "PROFIBUS" interface type in the Inspector window under "Properties > General > HMI MPIDP > Parameters".

5. Click the interface of the PLC and use a drag-and-drop operation to draw a connection to the HMI device.



6. Click the connecting line.
7. Click "Highlight HMI connection" and select the HMI connection.
   The connection is displayed graphically in the Inspector window.
8. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project.
   See the chapter "PROFIBUS parameters (Page 194)" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created an HMI connection between an HMI device and a SIMATIC S7 300/400 via PROFIBUS.

1.8.3.3 PROFIBUS parameters (Basic Panels)

PROFIBUS parameters for the HMI connection (Basic Panels)

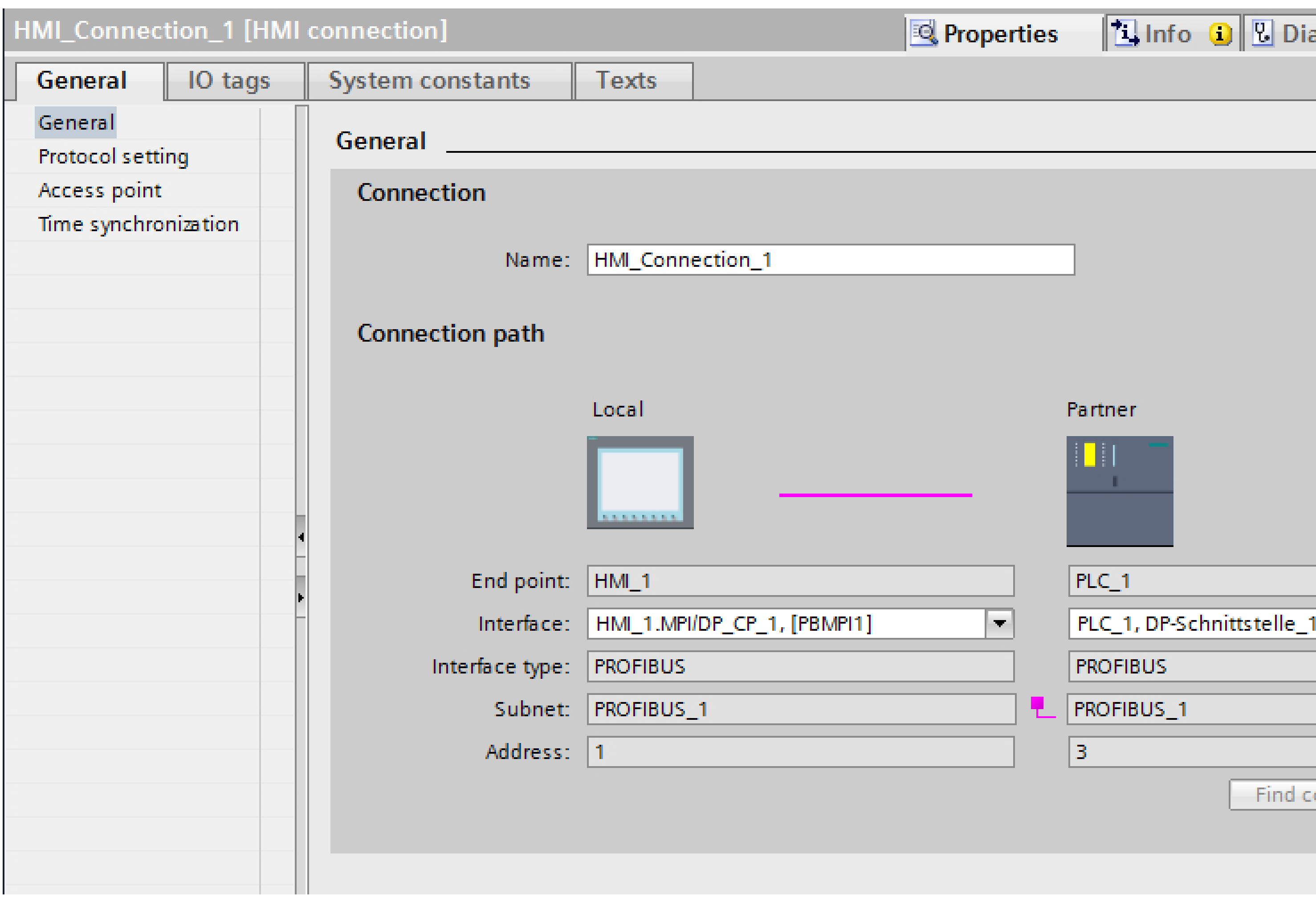
PROFIBUS parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and changing the HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

Displays whether the devices are networked together.

 - displayed if the devices are networked together.

 - displayed if the devices are not networked together.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFIBUS parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area is not editable.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area is not editable.
- "Subnet"
  Displays the selected subnet. This area is not editable.
- "Address"
  Displays the PROFIBUS address of the device. This area is not editable.
- "Find connection path" button
  Enables the subsequent specification of connections.

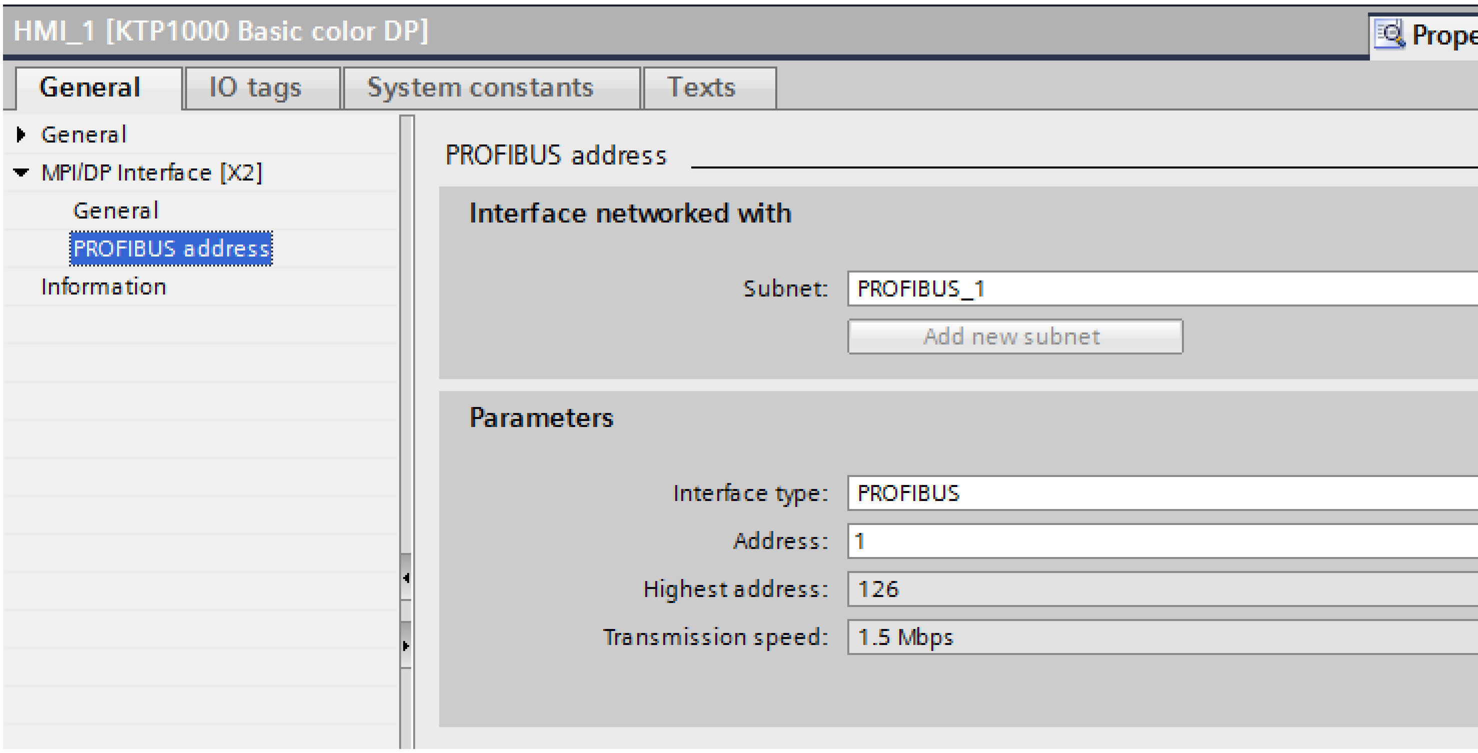
PROFIBUS parameters for the HMI device (Basic Panels)

PROFIBUS parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  You assign the interface type in the "Interface type" area. Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.
- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

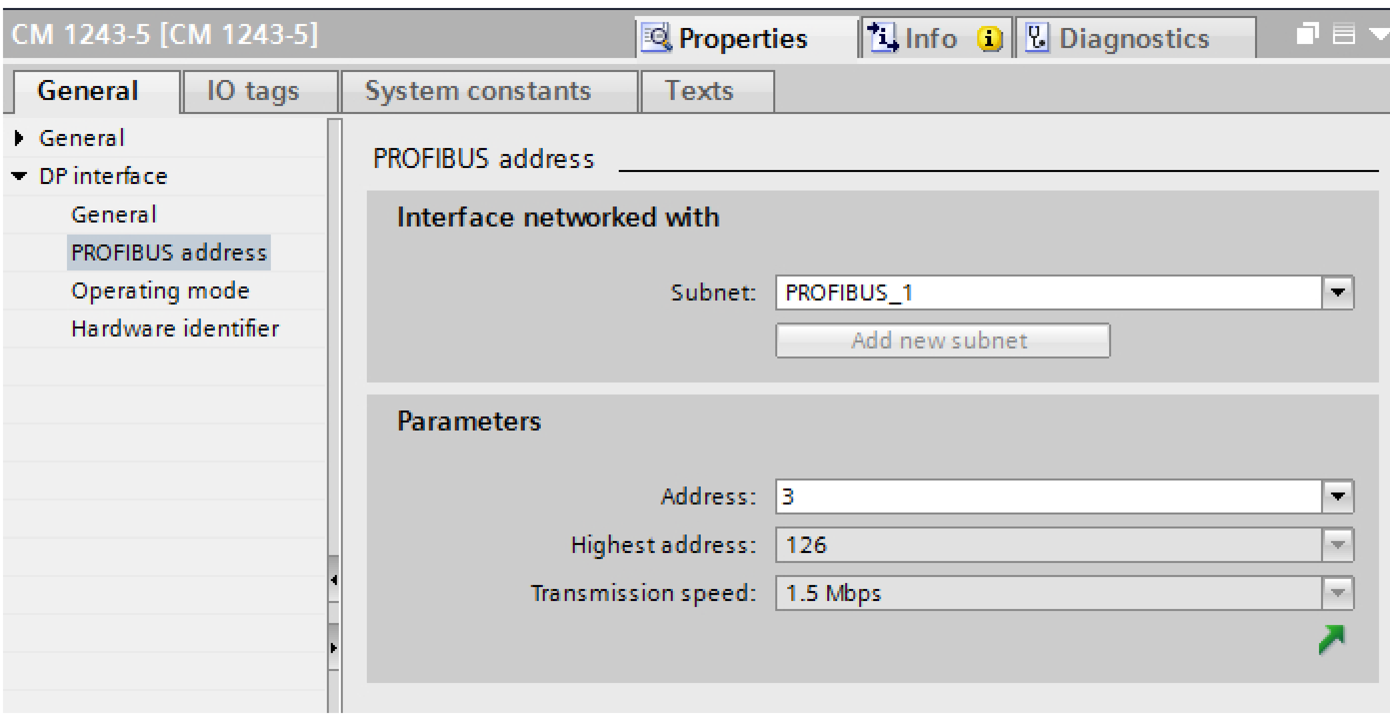
PROFIBUS parameters for the PLC (Basic Panels)

PROFIBUS parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFIBUS parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.

- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

Bus profiles with PROFIBUS (Basic Panels)

Introduction

Depending on the device types connected and protocols used on the PROFIBUS, different profiles are available. The profiles differ in terms of the setting options and calculation of bus parameters. The profiles are explained below.

Devices with different profiles on the same PROFIBUS subnet

The PROFIBUS subnet only functions without problem if the bus parameters of all devices have the same values.

Profiles and transmission rates

Profiles	Supported transmission speeds in Kbits/s
DP	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Standard	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Universal	9,6 19,2 93,75 187,5 500 1500

Meaning of profiles

Profile	Meaning
DP	Select the "DP" bus profile when the only devices connected to the PROFIBUS subnet are those which satisfy the requirements of standard EN 50170 Volume 2/3, Part 8-2 PROFIBUS. The bus parameter setting is optimized on these devices. This includes devices with DP master and DP slave interfaces of the SIMATIC S7 and distributed I/Os of other manufacturers.
Standard	Compared to the "DP" profile, the "Standard" profile also offers scope for devices of another project or devices which have not been configured here to be taken into account when calculating the bus parameters. The bus parameters are then calculated following a simple, non-optimized algorithm.
Universal	Select the "Universal" bus profile when individual devices on the PROFIBUS subnet use the PROFIBUS-FMS service. This includes the following devices for example: • CP 343-5 • PROFIBUS-FMS devices of other manufacturers As with the "Standard" profile, this profile allows you to take other devices into account when calculating the bus parameters.

1.8.4 Communication via MPI (Basic Panels)

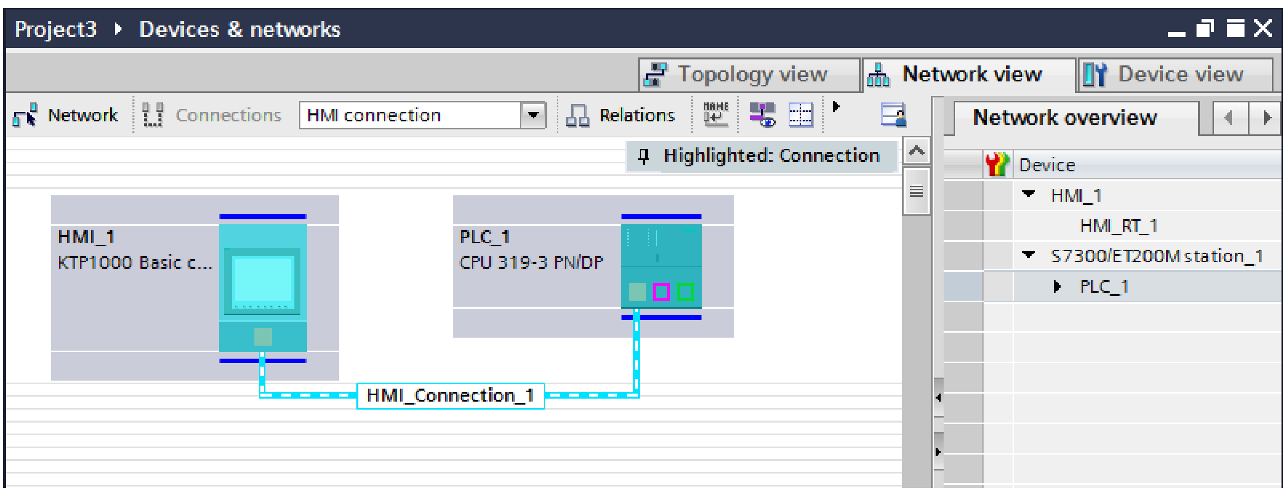
1.8.4.1 Communication via MPI (Basic Panels)

HMI connections via MPI

If you have inserted an HMI device and a SIMATIC S7 300/400 into the project, you interconnect the two MPI interfaces in the "Devices & Networks" editor.

HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFIBUS in the "Devices & Networks" editor.



Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device in the "Connections" editor of the HMI device.

1.8.4.2 Configuring an HMI connection via MPI (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC S7 300/400 via MPI in the "Devices & Networks" editor.

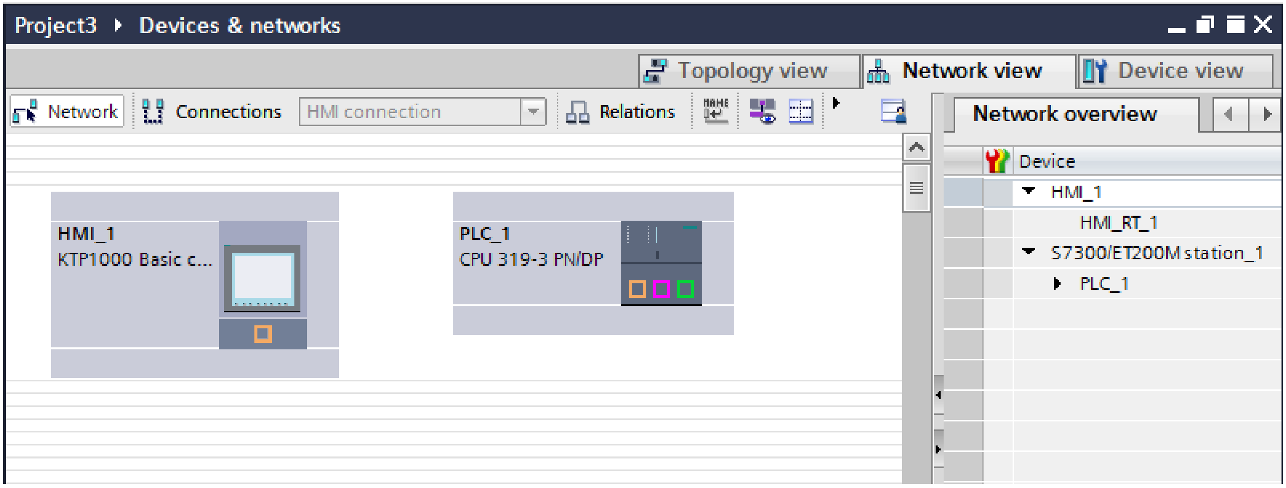
Requirements

The following communication partners are created in the "Devices & Networks" editor:

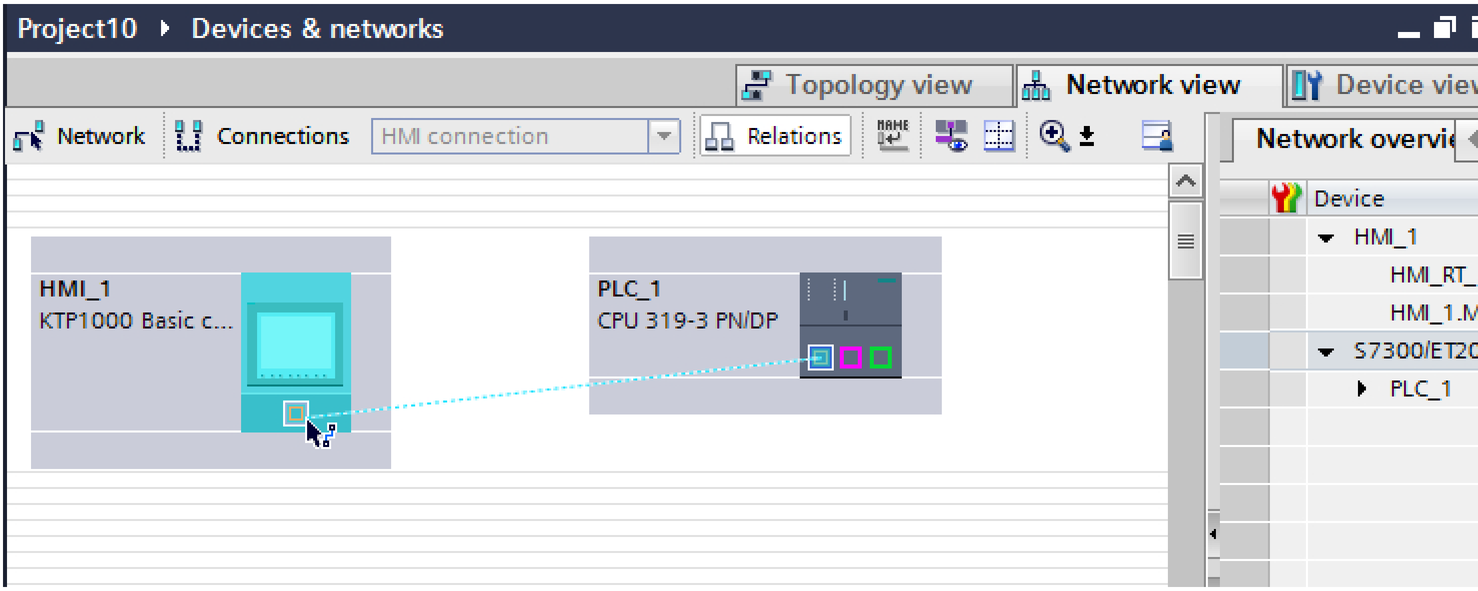
- HMI device with MPI/DP interface
- SIMATIC S7 300/400 with MPI/DP interface

Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.



2. Click the "Connections" button.
   The devices available for connection are highlighted in color.
3. Click the interface of the PLC and use a drag-and-drop operation to draw a connection to the HMI device.



4. Click the connecting line.
   The connection is displayed graphically in the Inspector window.
5. Click "Highlight HMI connection" and select the HMI connection.
6. Click the communication partners in the "Network view" and change the MPI parameters in the Inspector window according to the requirements of your project.
   See the chapter "MPI parameters (Page 201)" for additional details.

   Note

   The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. Use the table to monitor the connection parameters and change the connection partner. You can change the local name for the connection only in the table.

Result

You have created an HMI connection between an HMI device and a SIMATIC S7 300/400 via MPI.

1.8.4.3 MPI parameters (Basic Panels)

MPI parameters for the HMI connection (Basic Panels)

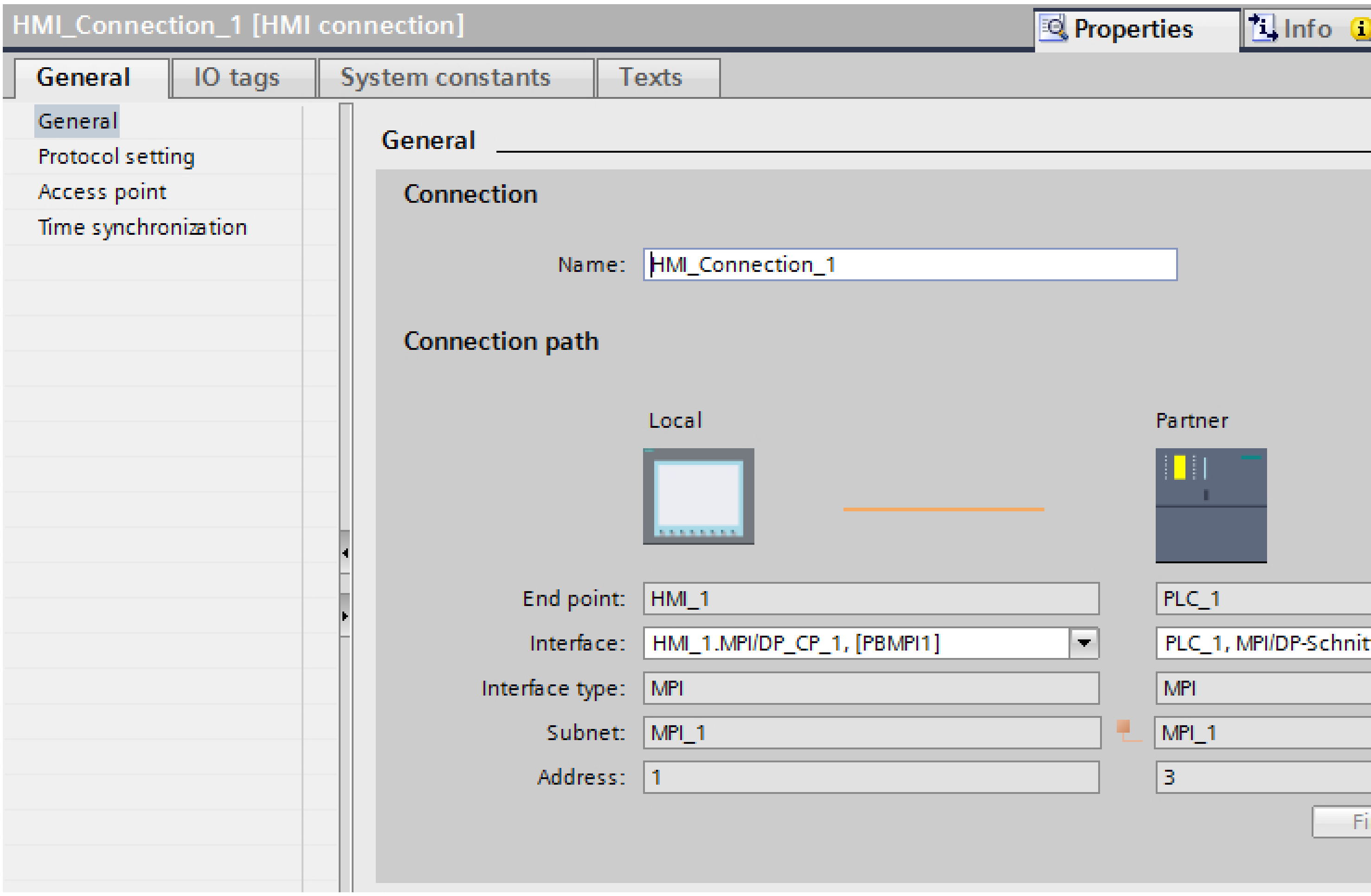
MPI parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and changing the HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

Displays whether the devices are networked together.

 - displayed if the devices are networked together.

 - displayed if the devices are not networked together.

"Connection path"

The communication partners of the selected HMI connection and the associated MPI parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the name of the device. This area is not editable.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area is not editable.
- "Subnet"
  Displays the selected subnet. This area is not editable.
- "Address"
  Displays the MPI address of the device. This area is not editable.
- "Find connection path" button
  Enables the subsequent specification of connections.

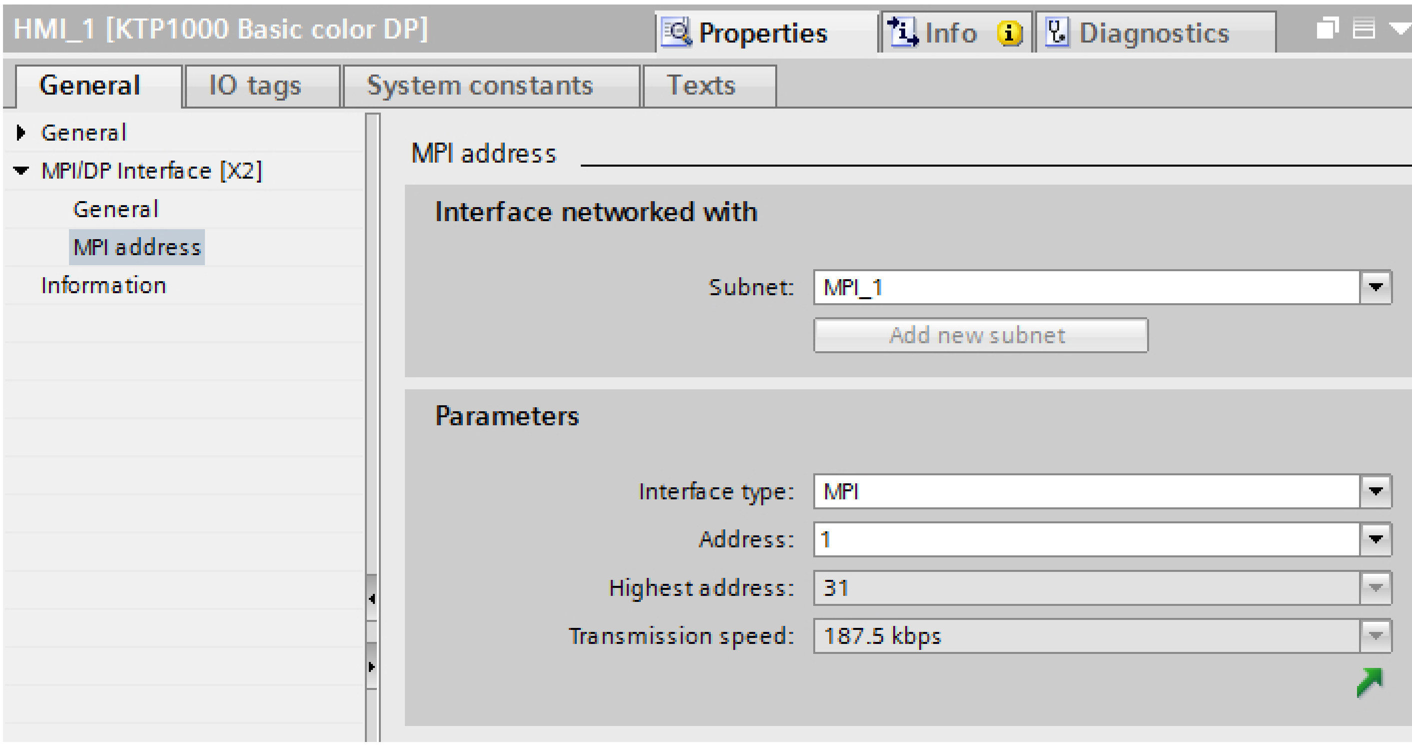
MPI parameters for the HMI device (Basic Panels)

MPI parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing MPI parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  You assign the interface type in the "Interface type" area. Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the MPI address of the HMI device in the "Address" area. The MPI address must be unique throughout the MPI network.
- "Highest address"
  The "Highest address" area displays the highest address of the MPI network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

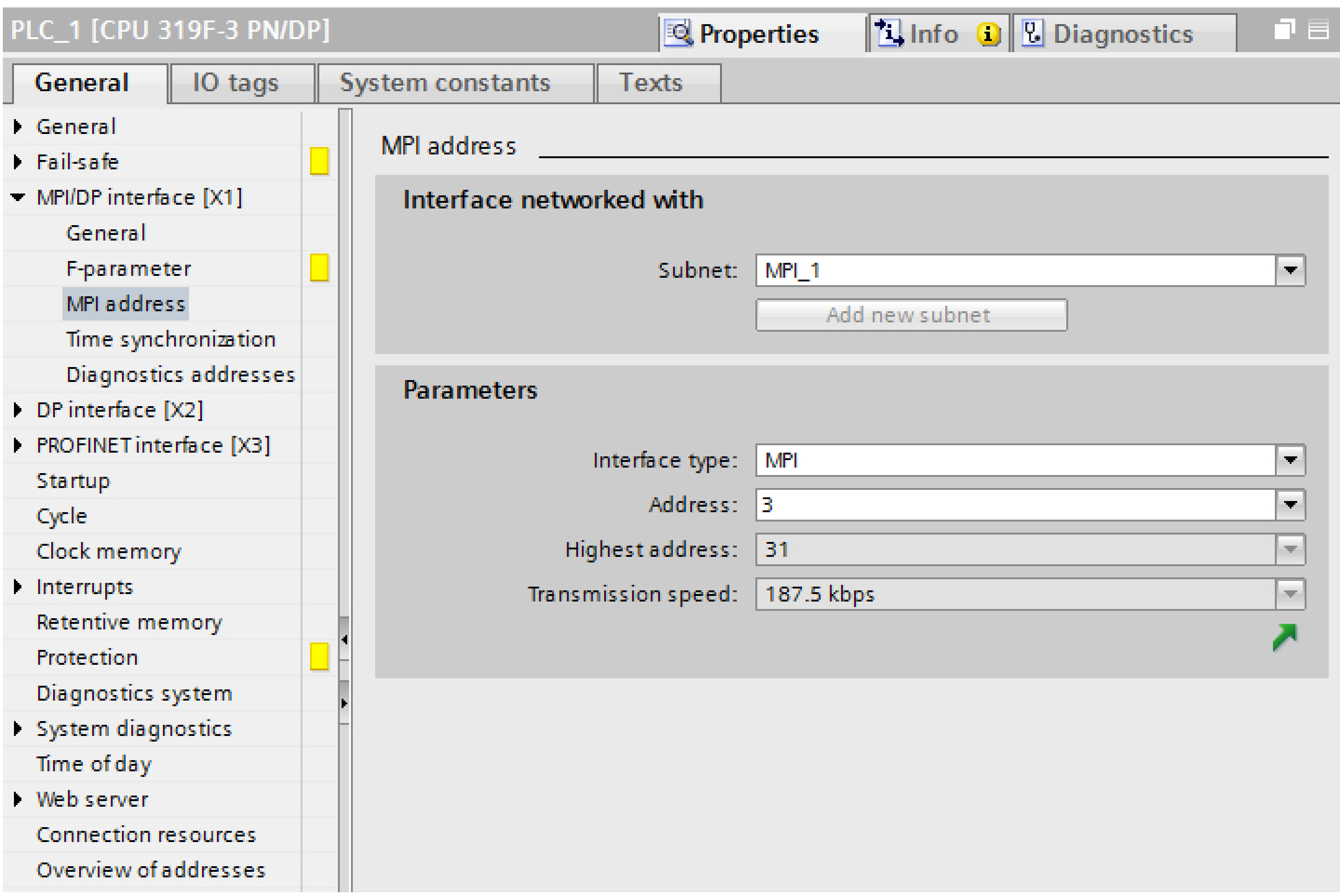
MPI parameters for the PLC (Basic Panels)

MPI parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFIBUS parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the MPI address of the HMI device in the "Address" area. The MPI address must be unique throughout the MPI network.
- "Highest address"
  The "Highest address" area displays the highest address of the MPI network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

Addressing of the PLC via MPI (Basic Panels)

Introduction

Each communication partner must be assigned an MPI network address.

Each S7 module which supports communication functions and is operated the SIMATIC S7-300/400 PLC is assigned a unique MPI address. Only one CPU may be used per rack.

Note

HMI devices cannot be operated with incorrect addressing

Always avoid redundant addressing on the MPI bus.

MPI address of the communication partner of a SIMATIC S7-300

When assigning addresses, you have to distinguish between communication partners with and without separate MPI address.

- If the communications partner has its own MPI address, you only need to define the MPI address.
- If the communication partners do not have a separate MPI address, specify the MPI address of the communications partner used for the connection. In addition, define the slot and rack of a communication partner without its own MPI address.

MPI address of the communication partner of a SIMATIC S7-400

Only S7 modules with an MPI connector are assigned an MPI address. Modules without an MPI connector are addressed indirectly:

- MPI address of the module to which the HMI is connected.
- The slot and the rack of the module with which the HMI device communicates.

1.8.5 Data exchange (Basic Panels)

1.8.5.1 Data exchange using area pointers (Basic Panels)

General information on area pointers (Basic Panels)

Introduction

You use an area pointer to access a data area in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations.

The PLC and the HMI device trigger defined interactions based on the evaluation of stored data.

Configuration of area pointers

Before you use an area pointer, you enable it under "Connections > Area pointer". You then assign the area pointer parameters.

You can find more detailed information on configuring area pointers in:

Configuring area pointers (Page 58)

"Screen number" area pointer (Basic Panels)

Function

The HMI device saves information about the screen called on the HMI device to the "Screen number" area pointer.

This allows the transfer of the current screen contents from the HMI device to the PLC. The PLC can trigger specific reactions such as the call of a different screen.

Use

Configure and enable the area pointer in "Communication > Connections" before you put it into use. You can create only **one** instance of the "Screen number" area pointer and only on **one** PLC.

The screen number is always transferred to the PLC when a new screen is activated or when the focus within a screen changes from one screen object to another.

Structure

The area pointer is a data area in the memory of the PLC with a fixed length of 5 words.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1st word	Current screen type															
2nd word	Current screen number															
3rd word	Reserved															
4th word	Current field number															
5th word	Reserved															

- Current screen type
  "1" for root screen or
  "4" for permanent window
- Current screen number
  1 to 32767
- Current field number
  1 to 32767

"Date/time" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the HMI device to the PLC.

The PLC writes control job "41" or "40" to the job mailbox.

When it evaluates the control job, the HMI device writes its current date and the time in the data area configured in the "Date/time" area pointer. All definitions are coded in BCD format.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

Note

You cannot use the "Date/Time PLC" area pointer if you have configured the "Date/Time" area pointer.

Note

Symbolic addressing is not possible if you are using the "Date/Time" area pointer.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

The date/time data area has the following structure:

Data word	Most significant byte		Least significant byte		
	7	0	7	0	
n+0	Reserved		Hour (0 to 23)		Time
n+1	Minute (0 to 59)		Second (0 to 59)		
n+2	Reserved		Reserved		
n+3	Reserved		Weekday (1 to 7, 1=Sunday)		Date
n+4	Day (1 to 31)		Month (1 to 12)		
n+5	Year (80 to 99/0 to 29)		Reserved		

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

"Date/time PLC" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the PLC to the HMI device. Use this area pointer if the PLC is the time master.

The PLC loads the data area of the area pointer. All definitions are coded in BCD format.

The HMI device reads the data cyclically within the configured acquisition cycle and synchronizes itself.

Note

Set an acquisition cycle of sufficient length for the Date/time area pointer in order to avoid any negative impact on HMI device performance.
Recommended: Acquisition cycle of 1 minute, if the process allows this.

You cannot use the "Date/Time" area pointer if you have configured the "Date/Time PLC" area pointer.

"Date/Time PLC" is a global area pointer and may be configured only once per project.

Note

You cannot use the "Date/Time" area pointer if you have configured the "Date/Time PLC" area pointer.

The date/time data area has the following structure:

DATE_AND_TIME format (in BCD code)

<table>
<tr><th rowspan="2">Data word</th><th colspan="3">Most significant byte</th><th colspan="3">Least significant byte</th></tr>
<tr><th>7</th><th>......</th><th>0</th><th>7</th><th>......</th><th>0</th></tr>
<tr><td>n+0</td><td colspan="3">Year (80 to 99/0 to 29)</td><td colspan="3">Month (1 to 12)</td></tr>
<tr><td>n+1</td><td colspan="3">Day (1 to 31)</td><td colspan="3">Hour (0 to 23)</td></tr>
<tr><td>n+2</td><td colspan="3">Minute (0 to 59)</td><td colspan="3">Second (0 to 59)</td></tr>
<tr><td>n+3</td><td colspan="3">Reserved</td><td colspan="2">Reserved</td><td>Weekday (1 to 7, 1=Sun-day)</td></tr>
<tr><td>n+4 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
<tr><td>n+5 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
</table>

1) The two data words must exist in the data area to ensure that the data format matches WinCC flexible and to avoid reading false information.

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

"Coordination" area pointer (Basic Panels)

Function

The "Coordination" area pointer is used to implement the following functions:

- Detecting the startup of the HMI device in the control program
- Detecting the current operating mode of the HMI device in the control program
- Detecting whether the HMI device is ready to communicate in the control program

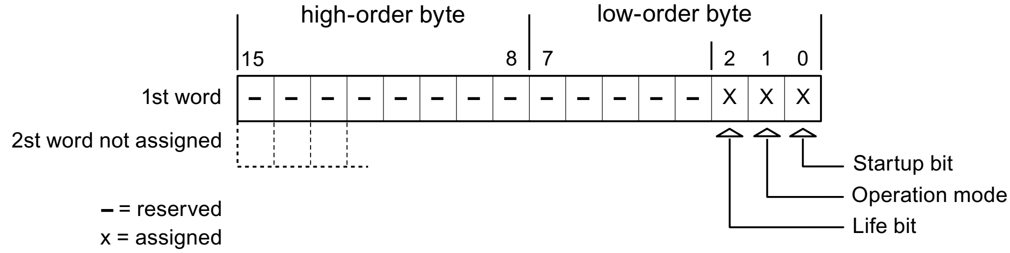
By default, the "Coordination" area pointer has the length of one word and cannot be changed.

Application

Note

The HMI device always writes the entire coordination area when updating the area pointer. The control program can for this reason not make changes to the coordination area.

Assignment of bits in the "Coordination" area pointer



Startup bit

The startup bit is set briefly to "0" by the HMI device during startup. It sets the bit permanently to "1" when startup is completed.

Operating mode

The operating mode bit is set to 1 as soon as the user switches the HMI device offline. The status of the operating mode bit is "0" during normal operation of the HMI device. You can determine the current operating mode of the HMI device by reading this bit in the control program.

Life bit

The HMI device inverts the life bit at intervals of approximately one second. You can check whether or not there is still a connection to the HMI device by querying this bit in the control program.

"Project ID" area pointer (Basic Panels)

Function

When Runtime starts, a check can be carried out as to whether the HMI device is connected to the correct PLC. This check is important when operating with several HMI devices.

For this, the HMI device compares a value stored on the PLC with the value specified in configuration. This ensures compatibility of the configuration data with the PLC program. If there is no concordance, a system event is given on the HMI device and Runtime is stopped.

Use

Note

HMI connections cannot be switched "online".

The HMI connection in which the "Project ID" area pointer is used must be switched "online".

To use this area pointer, set up the following during the configuration:

- Define the version of the configuration. Values between 1 and 255 are possible.
  You enter the version in the editor "Runtime settings > General" in the "Identification" area.
- Data address of the value for the version that is stored in the PLC:
  You enter the data address in the editor "Communication > Connections" under "Address".

Connection failure

A connection failure to a device on which the "Project ID" area pointer is configured results in all the other connections of the device being switched to "offline".

This behavior has the following requirements:

- You have configured several connections in a project.
- You are using the "Project ID" area pointer in at least one connection.

Causes which may set connections "offline":

- The PLC is not available.
- The connection has been switched offline in the Engineering System.

"Job mailbox" area pointer (Basic Panels)

Function

The PLC can use the job mailbox to transfer jobs to the HMI device to trigger corresponding actions on the HMI device. These functions include, for example:

- Display screen
- Set date and time

Data structure

The first word of the job mailbox contains the job number. Depending on the job mailbox, up to three parameters can be transferred.

Word	Most significant byte	Least significant byte
n+0	0	Job number
n+1	Parameter 1	
n+2	Parameter 2	
n+3	Parameter 3	

The HMI device evaluates the job mailbox if the first word of this job is not equal to zero. This means that the parameters must be entered in the job mailbox first, followed by the job number.

When the HMI device accepts the job mailbox, the first word is set to 0 again. The execution of the job mailbox is generally not completed at this point in time.

Job mailboxes

All job mailboxes and their parameters are listed below. The "No." column contains the job number of the job mailbox. Job mailboxes can only be triggered by the PLC when the HMI device is online.

No.	Function	
14	**Set time (BCD-coded)**	
	Parameter 1	Left byte: - Right byte: hours (0-23)
	Parameter 2	Left byte: minutes (0-59) Right byte: seconds (0-59)
	Parameter 3	-
15	**Setting the date (BCD coded)** [3) 4)]	
	Parameter 1	Left byte: - Right byte: weekday (1-7: Sunday-Saturday)
	Parameter 2	Left byte: day (1-31) Right byte: month (1-12)
	Parameter 3	Left byte: year
23	**User logon**	
	Logs the user on with the name "PLC user" at the HMI device with the group number transferred in Parameter 1. The logon is possible only when the transferred group number exists in the project.	
	Parameter 1	Group number 1 to 255
	Parameter 2, 3	-
24	**User logoff**	
	Logs off the current user. (The function corresponds to the "logoff" system function)	
	Parameter 1, 2, 3	-
40	**Transfer date/time to PLC**	
	(in the S7 format DATE_AND_TIME) An interval of at least 5 seconds must be maintained between two successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
41	**Transfer date/time to PLC**	
	(In OP/MP format) An interval of at least 5 seconds must be maintained between successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
46	**Update tag**	
	Causes the HMI device to read the current value of the tags from the PLC whose update ID matches the value transferred in Parameter 1. (Function corresponds to the "UpdateTag" system function.)	
	Parameter 1	1 - 100
49	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Warnings" class from the alarm buffer.	
	Parameter 1, 2, 3	-

No.	Function	
14	**Set time (BCD-coded)**	
50	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Errors" class from the alarm buffer.	
	Parameter 1, 2, 3	-
51	**Screen selection** [2]	
	Parameter 1	Screen number
	Parameter 2	-
	Parameter 3	Field number
69	**Reading data record from PLC** [1]	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	0: Do not overwrite existing data record 1: Overwrite existing data record
70	**Writing data record to PLC** [1]	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	-

1)	Only for devices supporting recipes.
2)	OP 73, OP 77A and TP 177A HMI devices also execute the "Screen selection" job mailbox if the on-screen keyboard is active.
3)	The weekday is ignored on HMI device KTP 600 BASIC PN.
4)	The weekday is ignored when you configure the "Date/Time PLC" area pointer.

"Data record" area pointer (Basic Panels)

"Data record" area pointer (Basic Panels)

Function

When data records are transferred between the HMI device and PLC, both partners access common communications areas on the PLC.

Data transfer types

There are two ways of transferring data records between the HMI device and PLC:

- Transfer without synchronization
- Transfer with synchronization via the data mailbox

Data records are always transferred directly, which means that the tag values are read straight from an address or written straight to an address configured for this tag without being redirected via an interim memory.

Initiating the transfer of data records

There are three ways of triggering the transfer:

- Operator input in the recipe view
- Job mailboxes
  The transfer of data records can also be triggered by the PLC.
- Triggering by configured functions

If the transfer of data records is triggered by a configured function or by a job mailbox, the recipe view on the HMI device remains operable. The data records are transferred in the background.

Simultaneous processing of several transfer requests is, however, not possible. In this case, the HMI device rejects the other transfer requests with a system alarm.

Transfer without synchronization (Basic Panels)

If you select asynchronous transfer of data records between the HMI device and PLC, there is no coordination over the common data areas. It is therefore unnecessary to set up a data area during configuration.

Asynchronous data record transfer can be a useful alternative, for example when:

- The system is capable of excluding the risk of uncontrolled overwriting of data by the communication peer.
- The PLC does not require information about the recipe number and data record number.
- The transfer of data records is triggered by the operator of the HMI device.

Reading values

When a read job is triggered, the values are read from the PLC addresses and transferred to the HMI device.

- Triggering by the operator in the recipe view:
  The values are downloaded to the HMI device. You can then, for example, process, edit, or save these values in the HMI device.
- Triggering by a function or job mailbox:
  The values are saved immediately to the data volume.

Writing values

When a write job is triggered, the values are written to the PLC addresses.

- Triggering by the operator in the recipe view:
  The current values are written to the PLC.
- Triggering by a function or job mailbox:
  The current values are written to the PLC from the data medium.

Transfer with synchronization (Basic Panels)

If you select synchronous transfer, both communication partners set status bits in the common data area. You can use this mechanism to prevent uncontrolled overwriting of data in either direction in your control program.

Application

Synchronous data record transfer can be a useful solution, for example, when:

- The PLC is the "active partner" in the transfer of data records.
- The PLC evaluates the information about the recipe number and data record number.
- The transfer of data records is triggered by means of a Job mailbox.

Requirements

In order to synchronize transfer of data records between the HMI device and the PLC, the following requirements must be met during configuration:

- An area pointer has been set up: "Communication > Connections" editor in "Area pointer".
- The PLC with which the HMI device synchronizes transfer of data records is specified in the recipe:
  "Recipes" editor in the inspector window the option "Coordinated transfer of data records" under "General > Synchronization > Settings"

Structure of the data area

The data area has a fixed length of 5 words. Structure of the data area:

	15		0
1. Word	Current recipe number (1 - 999)		
2. Word	Current data record number (0 - 65535)		
3. Word	Reserved		
4. Word	Status (0, 2, 4, 12)		
5. Word	Reserved		

- Status
  The status word (word 4) can adopt the following values:

Value		Meaning
Decimal	Binary	
0	0000 0000	Transfer permitted, data mailbox free
2	0000 0010	Transferring
4	0000 0100	Transfer completed without error
12	0000 1100	Transfer completed with error

Sequence of a transfer started by the operator in the recipe view (Basic Panels)

Reading from the PLC started by the operator in the recipe view

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe number to be read and the status "Transferring" in the data record and sets the data record number to 0.	Abort with system event.
3	The HMI device reads the values from the PLC and displays them in the recipe view. If the recipes have synchronized tags, the values from the PLC are also written to the tags.	
4	The HMI device sets the status "Transfer completed."	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC started by the operator in the recipe view

Step	Action	
	Check: Status word = 0?	
1	Yes	No
	The HMI device enters the recipe and data record number to be written and the status "Transferring" in the data record.	Abort with system event.
2	The HMI device writes the current values to the PLC. If the recipes have synchronized tags, the changed values are synchronized between the recipe view and tags and then written to the PLC.	
3	The HMI device sets the status "Transfer completed."	
4	If required, the control program can now evaluate the transferred data.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Sequence of the transfer triggered by a job mailbox (Basic Panels)

The transfer of data records between the HMI device and the PLC can be initiated by either one of these stations.

The two job mailboxs No. 69 and No. 70 are available for this type of transfer.

No. 69: Read data record from PLC ("PLC → DAT")

Job mailbox no. 69 transfers data records from the PLC to the HMI device. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	69
Word 2	Recipe number (1-999)	
Word 3	Data record number (1 to 65535)	
Word 4	Do not overwrite existing data record: 0 Overwrite existing data record: 1	

No. 70: Write data record to PLC ("DAT → PLC")

Job mailbox No. 70 transfers data records from the HMI device to the PLC. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	70
Word 2	Recipe number (1-999)	
Word 3	Data record number (1-65,535)	
Word 4	—	

Sequence when reading from the PLC with job mailbox "PLC → DAT" (no. 69)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data record.	Abort without return message.
3	The HMI device reads the values from the PLC and saves these to the data record defined in the job mailbox.	
4	• If "Overwrite" was selected in the job, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "Do not overwrite" was selected in the job, and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Sequence of writing to the PLC with job mailbox "DAT → PLC" (no. 70)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data mailbox.	Abort without return message.
3	The HMI device fetches the values of the data record specified in the function from the data medium and writes the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Sequence of the transfer when triggered by a configured function (Basic Panels)

Reading from the PLC using a configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system event.
3	The HMI device reads the values from the PLC and stores them in the data record specified in the function.	
4	• If "Yes" was selected for the "Overwrite" function, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "No" was selected for the "Overwrite" function and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC by means of configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system event.
3	The HMI device fetches the values of the data record specified in the function from the data medium and transfers the values to the PLC.	

Step	Action	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Possible causes of error when transferring data records (Basic Panels)

Possible causes of error

The section below shows possible error causes which lead to the cancellation of data record transfer:

- Tag address not set up on the PLC
- Overwriting data records not possible
- Recipe number does not exist
- Data record number does not exist

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Reaction to an aborted transfer due to errors

If the transfer of data records is aborted due to errors, the HMI device reacts as follows:

- Triggering by the operator in the recipe view
  Information in the status bar of the recipe view and output of system alarms
- Triggered by function
  Output of system alarms
- Triggering by job mailbox
  No return message on the HMI device

You can nonetheless evaluate the status of the transfer by querying the status word in the data record.

1.8.5.2 Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

1.8.5.3 Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configure alarms

Several steps are needed to configure alarms, such as operational messages, error alarms, and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a SIMATIC communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
SIMATIC S7 300/400	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIME

How the bit positions are counted

For connections with a SIMATIC communication driver, the following counting method applies:

How the bit positions are counted	Byte 0								Byte 1							
	Most significant byte								Least significant byte							
In SIMATIC S7 PLCs	7							0	7							0
In WinCC you configure:	15							8	7							0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

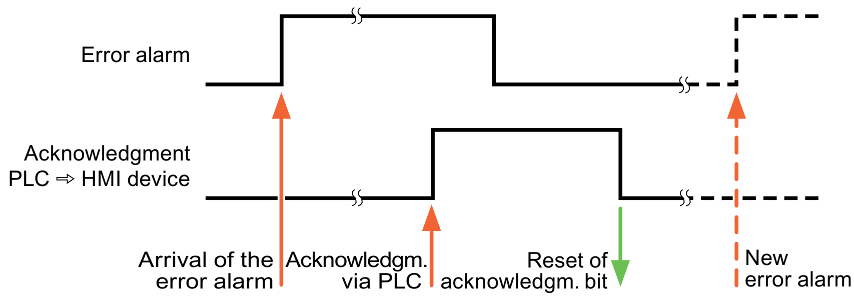
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

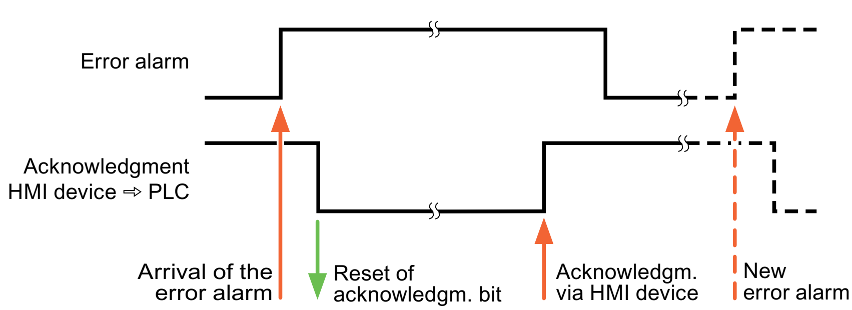
If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.8.6 Performance features of communication (Basic Panels)

1.8.6.1 S7-300/400 device dependency (Basic Panels)

Communication with the SIMATIC S7-300/400 controller

If you use devices from an earlier version of the TIA Portal with TIA Portal V13, it may not be possible to configure integrated connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC S7-300/400
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V12.0

HMI devices	SIMATIC S7-300/400
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V13.0

HMI devices	SIMATIC S7-300/400
KTP400 Basic PN	Yes
KTP700 Basic PN	Yes
KTP700 Basic DP	Yes

HMI devices	SIMATIC S7-300/400
KTP900 Basic PN	Yes
KTP1200 Basic PN	Yes
KTP1200 Basic DP	Yes

1.8.6.2 Permitted data types for SIMATIC S7 300/400 (Basic Panels)

Permitted data types for connections with SIMATIC S7 300/400

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Length
BOOL	1-bit
BYTE	1 byte
WORD	2 bytes
DWORD	4 bytes
CHAR	1 byte
INT	2 byte
DINT	4 bytes
REAL	4 bytes
TIME	4 bytes
DATE	2 bytes
TIME_OF_DAY, TOD	4 bytes
S5TIME	2 bytes
COUNTER	2 bytes
TIMER	2 bytes
DATE_AND_TIME	8 bytes
STRING	(2+n) bytes, n = 0 to 254

1.8.7 Creating connections in the "Connections" editor (Basic Panels)

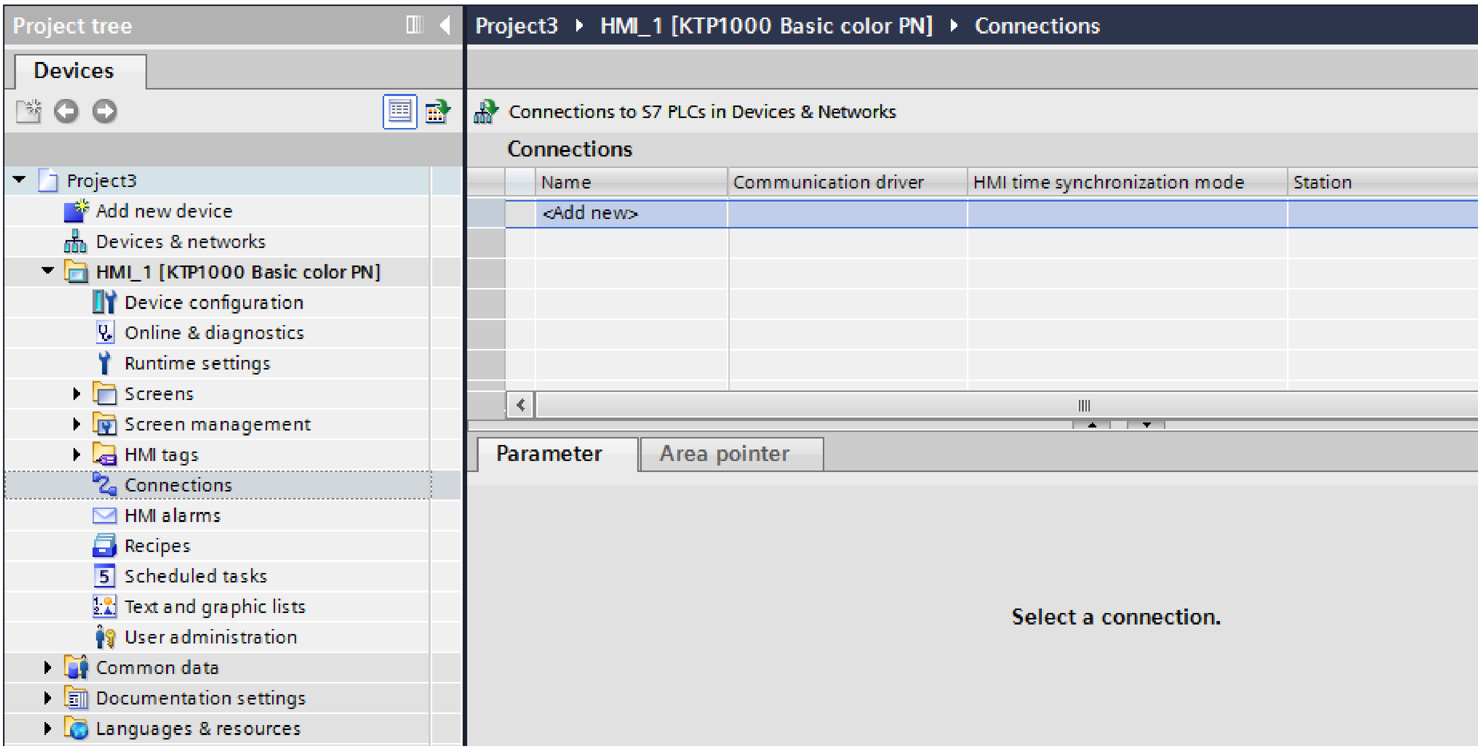
1.8.7.1 Creating a PROFINET connection (Basic Panels)

Requirements

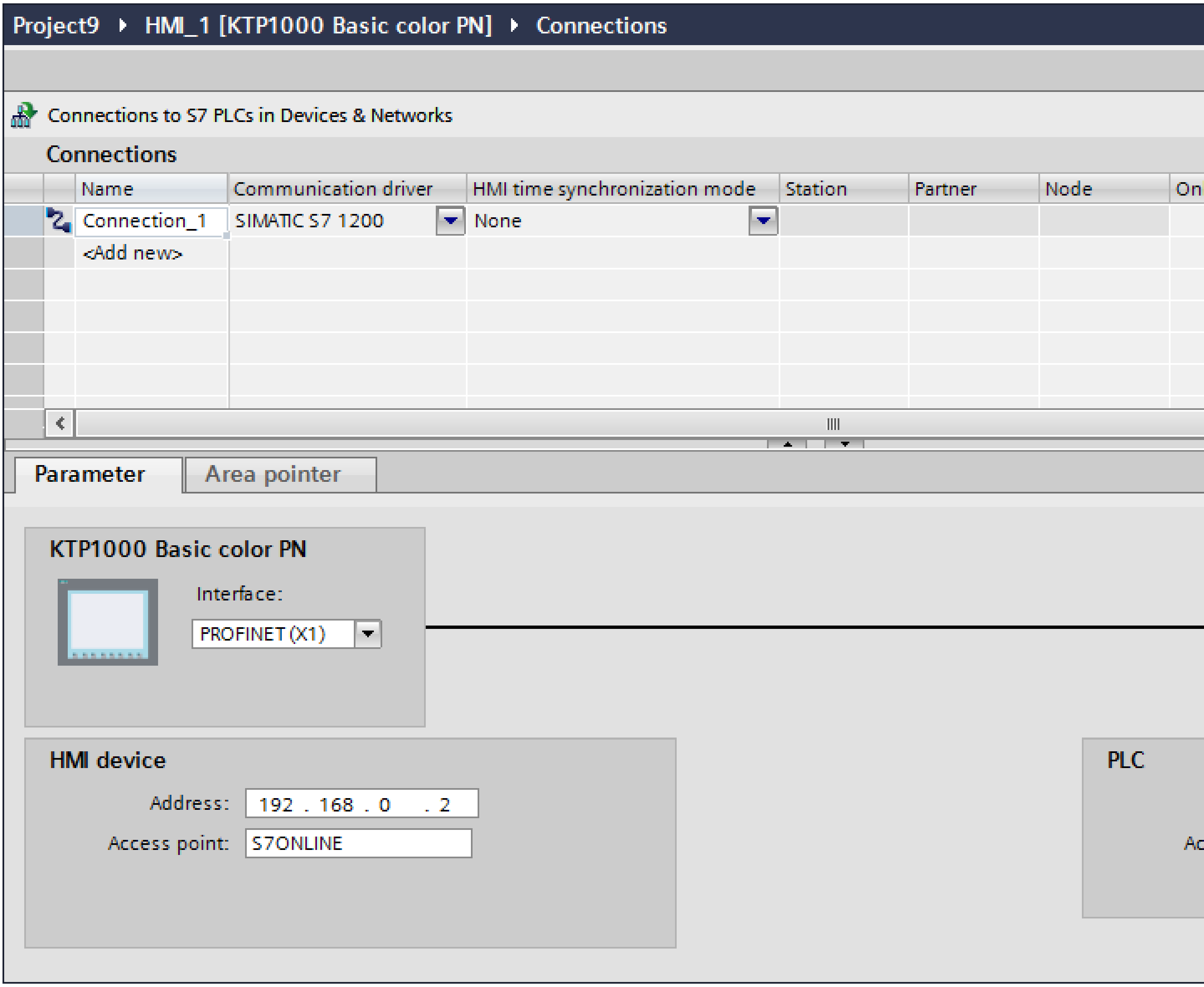
- A project is open.
- An HMI device with a PROFINET interface has been created.

Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. Select the driver in the "Communication driver" column.



4. Click the name of the connection.
5. Select a PROFINET interface of the HMI device in the Inspector window under "Parameters > Interface".
6. Set the IP addresses of the communication partners in the Inspector window:
   – HMI device: "Parameters > HMI device > Address"
   – PLC: "Parameters > PLC > Address"

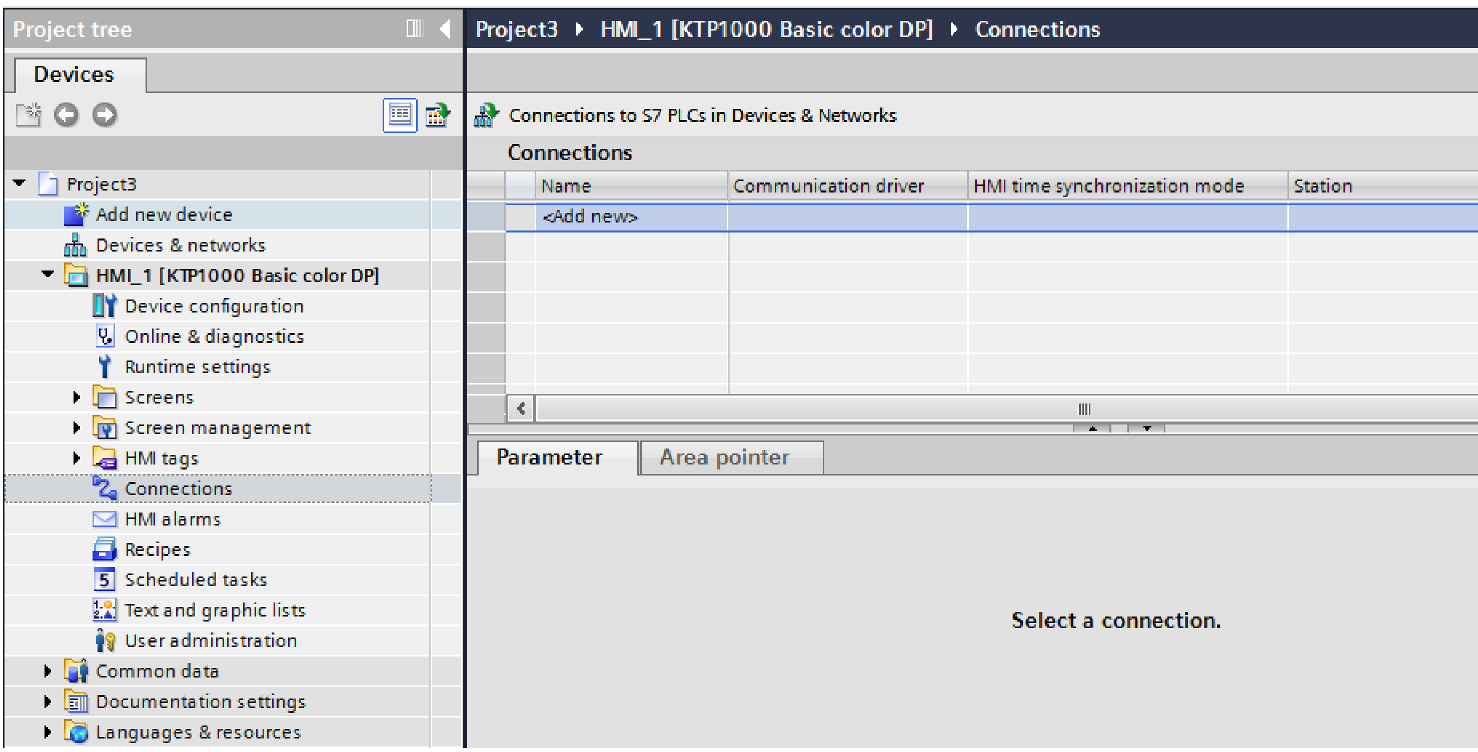
1.8.7.2 Creating a PROFIBUS connection (Basic Panels)

Requirements

- A project is open.
- An HMI device with a PROFIBUS interface has been created.

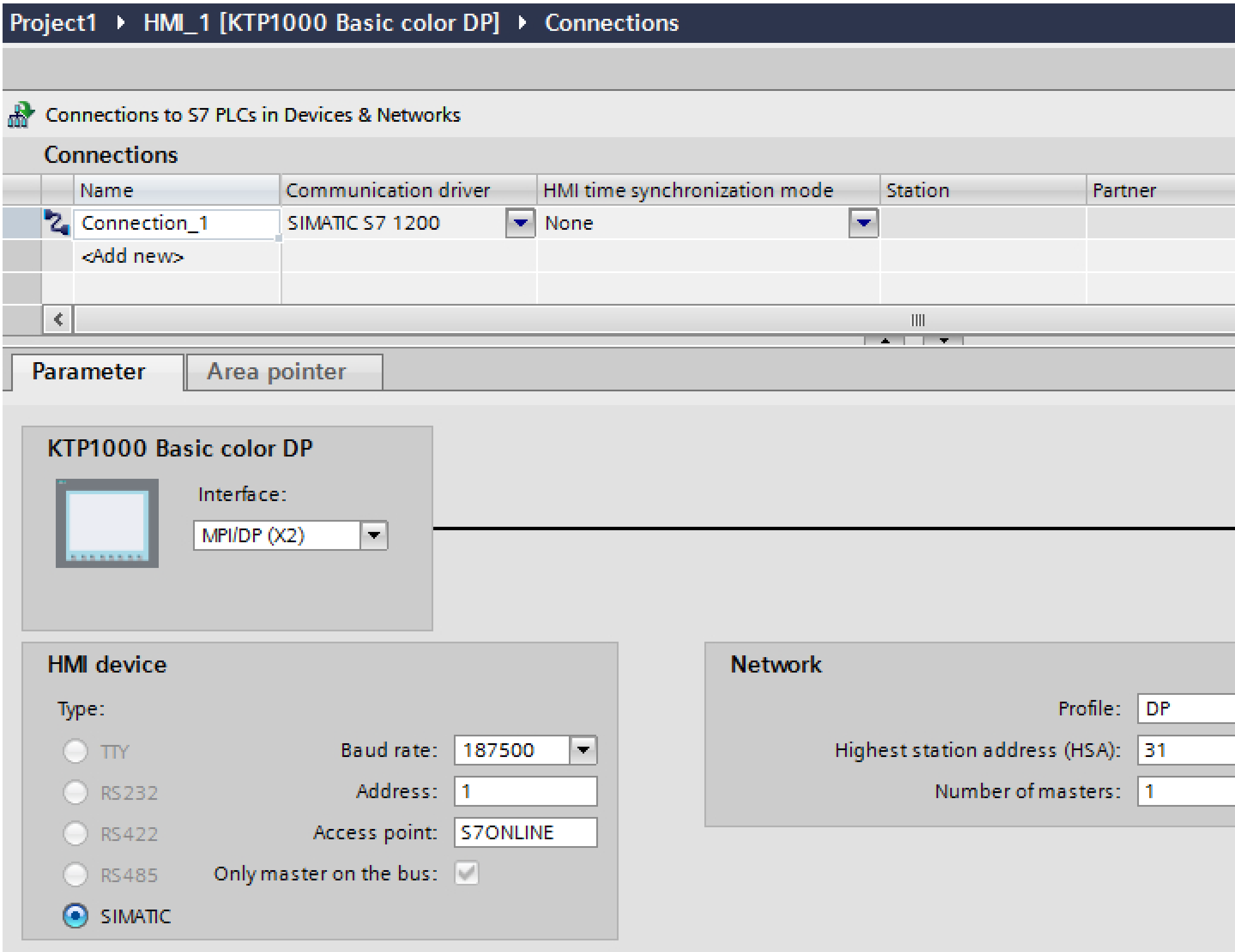
Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. Select the driver in the "Communication driver" column.
4. Click the name of the connection.
5. Select the "MPI/DP" interface in the Inspector window under "Parameters > Interface".

6. Select the "DP" profile in the Inspector window under "Parameters > Network".



7. Set the addresses of the communication partners in the inspector window:
   – HMI device: "Parameters > HMI device > Address"
   – PLC: "Parameters > PLC > Address"

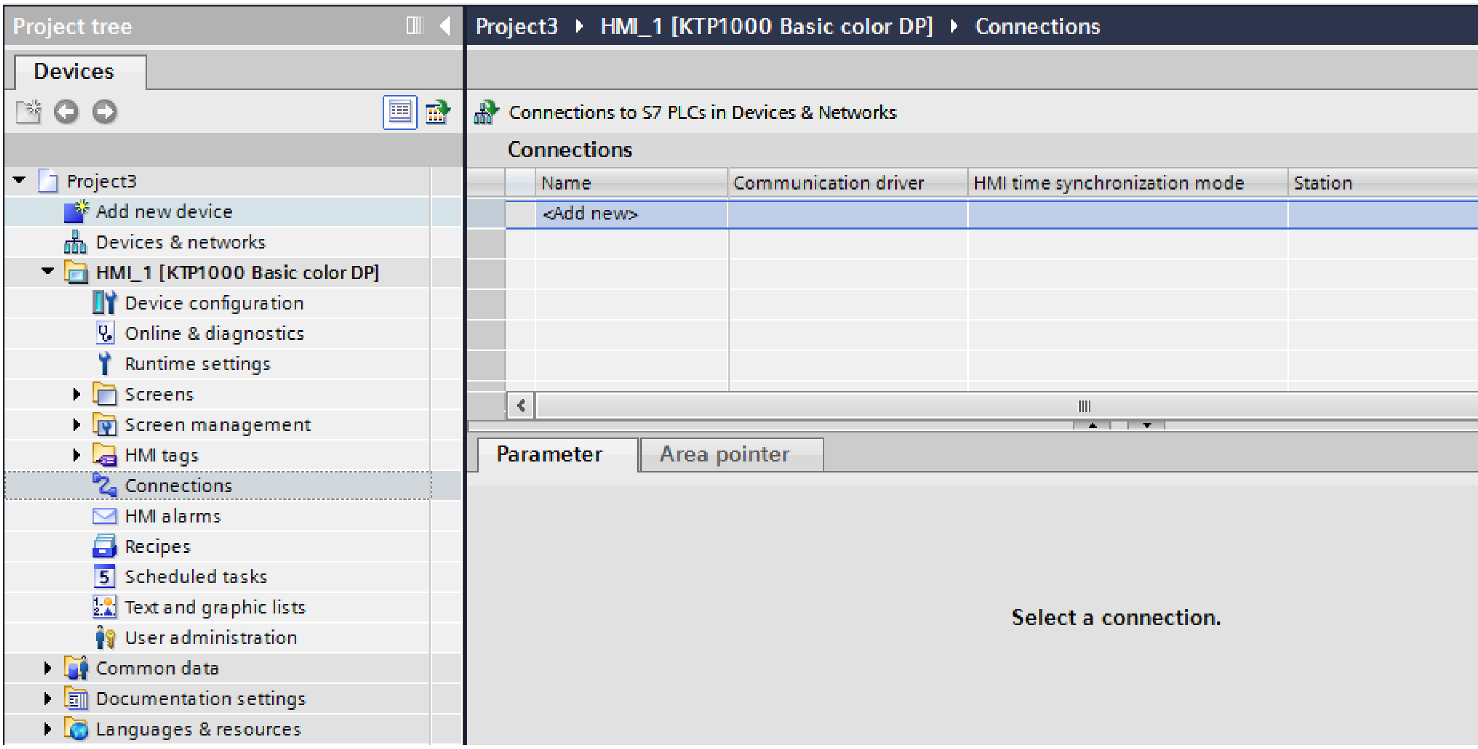
1.8.7.3 Creating an MPI connection (Basic Panels)

Requirements

- A project is open.
- An HMI device with an MPI interface has been created.

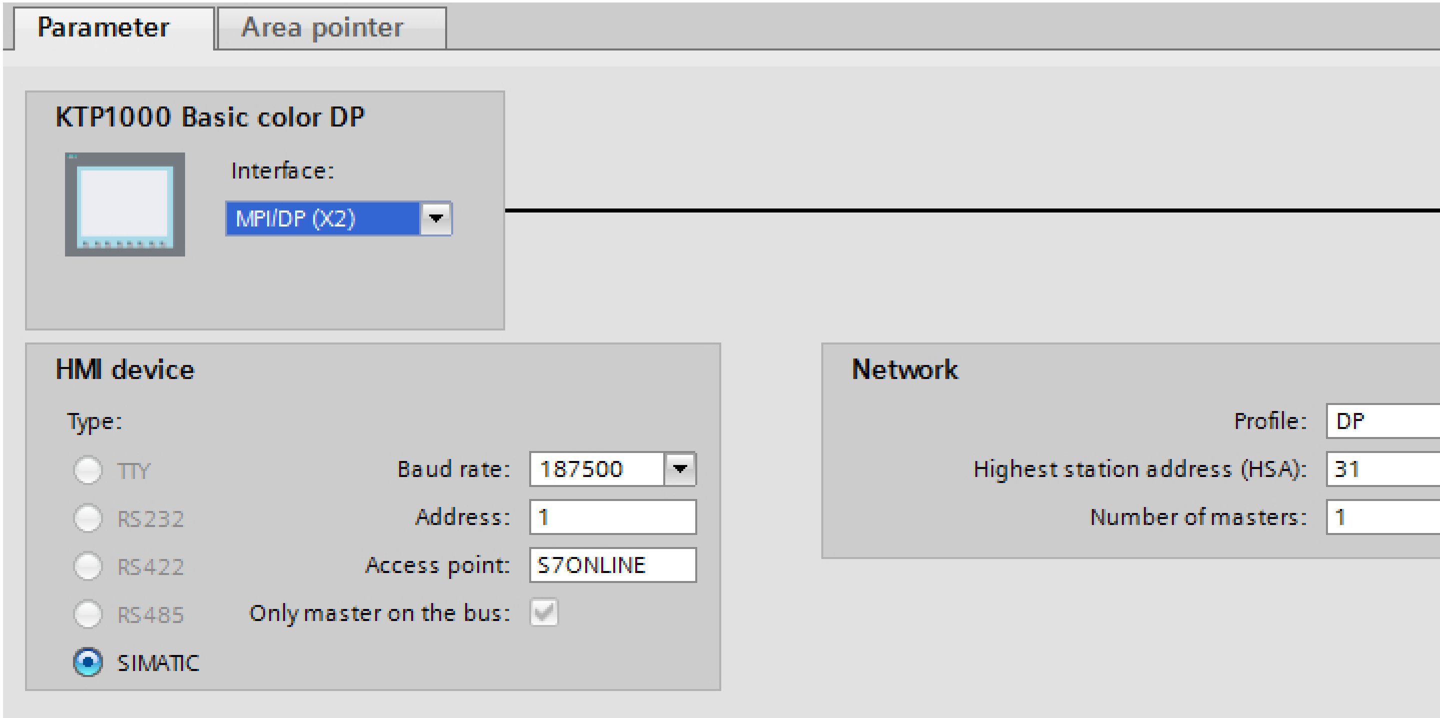
Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. Select the driver in the "Communication driver" column.
4. Click the name of the connection.
5. Select the "MPI/DP" interface in the Inspector window under "Parameters > Interface".

6. Select the "MPI" profile in the Inspector window under "Parameters > Network".



7. Set the addresses of the communication partners in the inspector window:
   - HMI device: "Parameters > HMI device > Address"
   - PLC: "Parameters > PLC > Address"

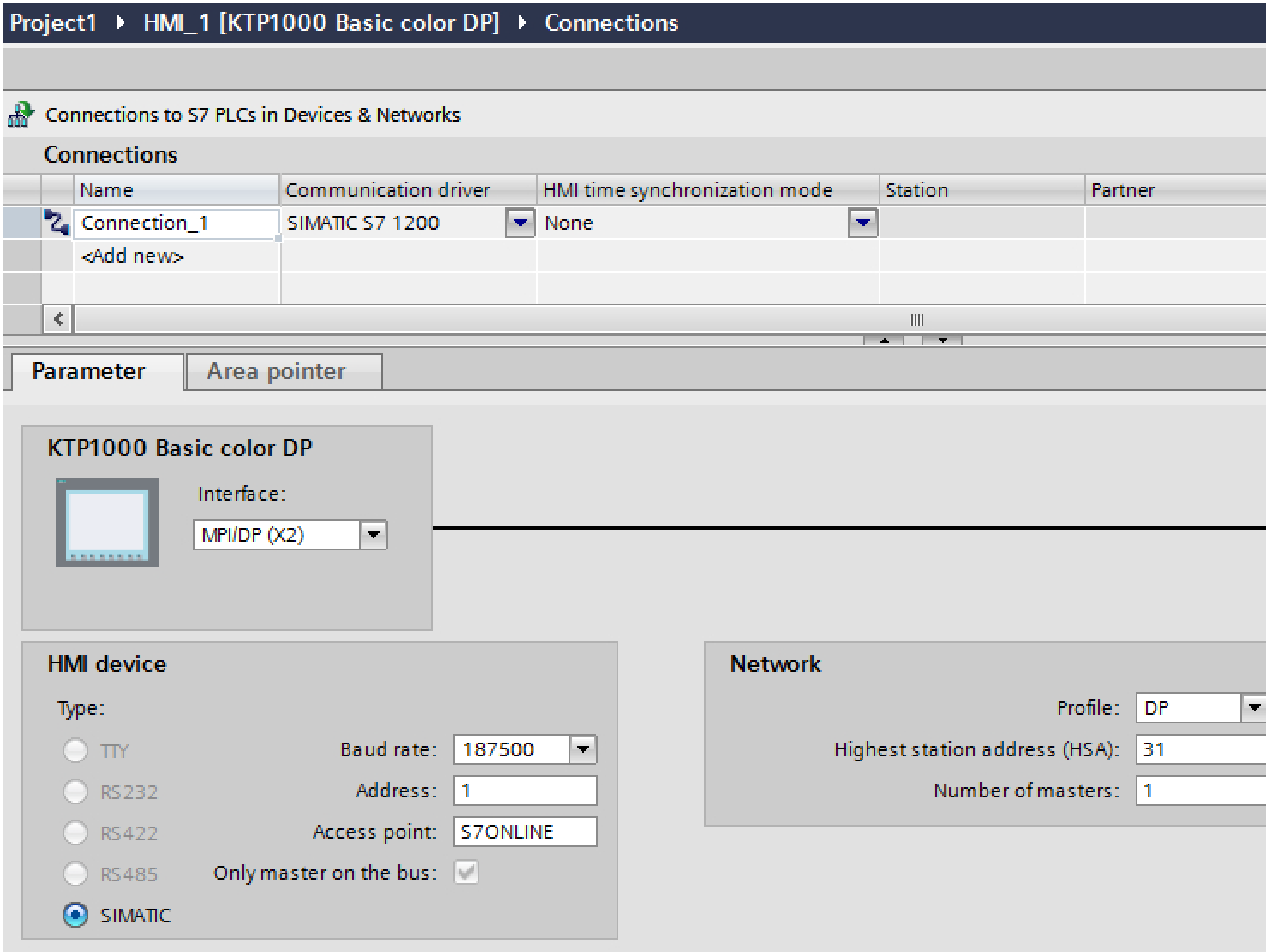
1.8.7.4 Parameters for the connection (Basic Panels)

Parameters for the connection (SIMATIC S7 300/400) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device", "Network", and "PLC" areas are available for assigning parameters according to the interface used.


Project1 ▸ HMI_1 [KTP1000 Basic color DP] ▸ Connections
Connections to S7 PLCs in Devices & Networks
Connections
Name
Communication driver
HMI time synchronization mode
Station
Partner
Connection_1
SIMATIC S7 1200
None
<Add new>
Parameter
Area pointer
KTP1000 Basic color DP
Interface:
MPI/DP (X2)
HMI device
Type:
TTY
RS232
RS422
RS485
SIMATIC
Baud rate:
187500
Address:
1
Access point:
S7ONLINE
Only master on the bus:
Network
Profile:
DP
Highest station address (HSA):
31
Number of masters:
1

Ethernet parameters (Basic Panels)

Parameters for the HMI device

You set the parameters for the HMI device in the network under "HMI device".. The changes are not transferred automatically to the HMI device. You must change the settings in the Control Panel of the HMI device.

- "Interface"
  If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC.

  Note

  The IP address in the Control Panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

  The IP address already set up in the Control Panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

  The IP address is transferred to the HMI device during project transfer.
  To set up the IP address of the HMI device:
  - Click the HMI device.
  - Open the "Device configuration" editor.
  - Click the Ethernet interface.
  - Assign the IP address in the inspector window under:
    "General > PROFINET interface > Ethernet addresses"
- "Address"
  You assign the IP address of the HMI device in the "Address" area.
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  Under "Address", set the IP address of the S7 module to which the HMI device is connected.
- "Expansion slot"
  Defines the number of the expansion slot of the CPU to be addressed.

- "Rack"
  Defines the rack number of the CPU to be addressed.
- "Cyclic operation"

  Note

  The setting "Cyclic operation" cannot be configured for the SIMATIC S7 1200 PLC.

  When cyclic operation is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This increases system performance.
  Disable cyclic mode if you are operating several HMI devices in parallel.

PROFIBUS parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the PROFIBUS network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  You set the PROFIBUS address of the HMI device under "Address". The PROFIBUS address must be unique in the PROFIBUS network.
- "Sole master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master).
  In S7 200, you must set an HMI device as the master.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the network

Under "Network", you set the parameters for the PROFIBUS network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "DP", "Universal", or "Standard". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual PROFIBUS address. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud in the OP 73 or the OP 77A, the highest station address must be less than or equal 63.

- "Number of masters"
  For "Number of masters", set the number of masters in the PROFIBUS network. This information is necessary to correctly calculate the bus parameters.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the PROFIBUS address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Cyclic operation"

  Note

  The setting "Cyclic operation" cannot be configured for the SIMATIC S7 1200 PLC.

  When cyclic operation is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This increases system performance. Disable cyclic mode if you are operating several HMI devices in parallel. This setting is not required for SIMATIC S7-200 PLCs.

MPI parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the MPI network.

- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  For "Address", you set the MPI address of the HMI device. The MPI address must be unique throughout the MPI network.
- "Sole master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master). If you have connected only slaves to the HMI device, you must therefore disable the "Sole master on bus" safety feature.
  In S7 200, you must set an HMI device as the master.

Parameters for the network

Under "Network", you set the parameters for the MPI network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "MPI". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual MPI address. The setting must be identical throughout the network.
- "Number of masters"
  This setting is not required for MPI.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the MPI address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Cyclic mode"
  When cyclic mode is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This improves system performance. Disable cyclic mode if you are operating several HMI devices in parallel. This setting is not required for SIMATIC S7-200.

Cyclic operation (Basic Panels)

Handling the "Cyclic operation" selection

If "Cyclic operation" is enabled, the HMI device sends a message frame to the CPU at the beginning of communication indicating that certain tags are required on a recurring basis.

The CPU then always transmits the data at the same cyclic interval. This saves the HMI device from having to output new requests for the data.

If cyclic operation is disabled, the HMI device sends a request whenever information is required.

Additional properties:

- Cyclic operation reduces data transmission load at the HMI device. The PLC resources are used to relieve load on the HMI device.
- The PLC only supports a certain number of cyclic services. The HMI device handles the operation if the PLC cannot provide any further resources for cyclic services.
- The HMI device generates the cycle if the PLC does not support the cyclic mode.
- Screen tags are not integrated in cyclic operation.
- Cyclic mode is only set up at the restart of Runtime.
- The HMI device transfers several jobs to the PLC if cyclic mode is enabled, depending on the PLC.
- The HMI device only transfers one job to the PLC if cyclic mode is disabled.

1.9 Communicating with the SIMATIC S7-1500 Software Controller (Basic Panels)

1.9.1 Communication with SIMATIC S7-1500 Software Controller (Basic Panels)

Introduction

This section describes the communication between a HMI device and a SIMATIC S7-1500 Software Controller.

You can configure the following communication channels for the SIMATIC S7-1500 Software Controller :

- PROFINET

HMI connection for communication

You configure connections between the HMI device and SIMATIC S7-1500 Software Controller in the "Devices & Networks" editor.

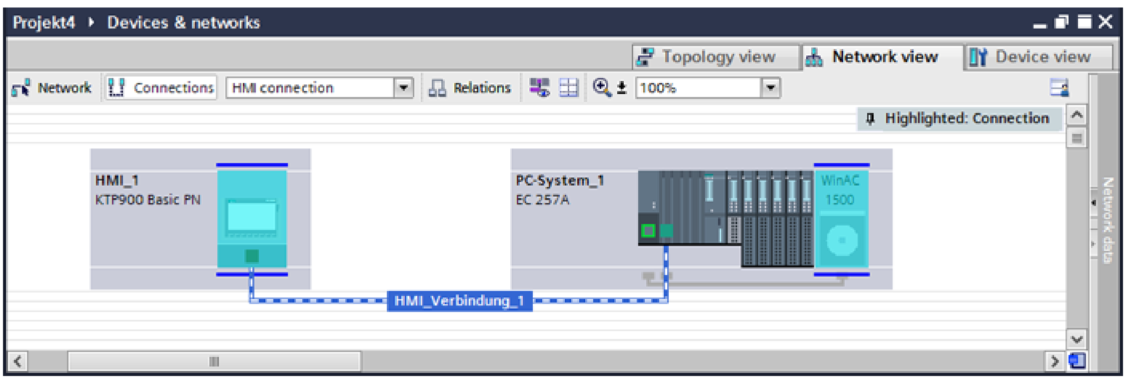
1.9.2 Communication via PROFINET (Basic Panels)

1.9.2.1 Configuring an HMI connection (Basic Panels)

Communication via PROFINET (Basic Panels)

HMI connections via PROFINET

If you have inserted an HMI device and a SIMATIC WinAC 1500 into the project, interconnect the two PROFINET interfaces in the "Devices & Networks" editor.



You can also connect multiple HMI devices to a single SIMATIC WinAC 1500 and multiple SIMATIC WinAC 1500 to a single HMI device.

The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used.

Additional information is available in the documentation for the respective HMI device.

HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFINET in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device via PROFINET in the "Connections" editor of the HMI device.

Configuring an HMI connection via PROFINET (Basic Panels)

Introduction

You configure a HMI connection between HMI devices and a SIMATIC S7-1500 Software Controller via PROFINET or Ethernet in the "Devices & Networks" editor.

⚠ CAUTION
Communication via Ethernet In Ethernet-based communication, the end user is responsible for the security of his data network. Targeted attacks can overload the device and interfere with proper functioning.

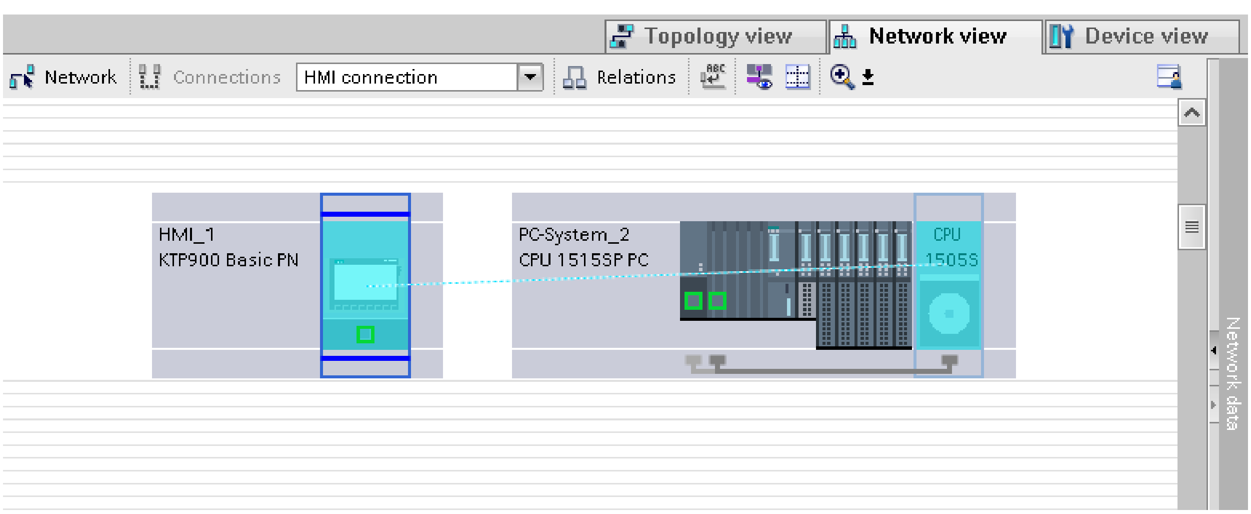
Requirements

The following communication partners are created in the "Devices & Networks" editor:

- HMI device with PROFINET or Ethernet interface
- SIMATIC S7-1500 Software Controller on PC system with PROFINET interface.

Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button and select "HMI connection" for the connection type.
   The devices available for connection are highlighted in color.
3. Click the HMI device and use a drag-and-drop operation to draw a connection to the SIMATIC S7-1500 Software Controller.



4. Click the connecting line.

5. Click "Highlight HMI connection" and select the HMI connection.
   The connection is displayed graphically in the Inspector window.
6. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project.
   See the section "AUTOHOTSPOT" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created a connection between an HMI device and a SIMATIC S7-1500 Software Controller. The IP address and subnet mask connection parameters are configured.

1.9.2.2 PROFINET parameters (Basic Panels)

PROFINET parameters for the HMI connection (Basic Panels)

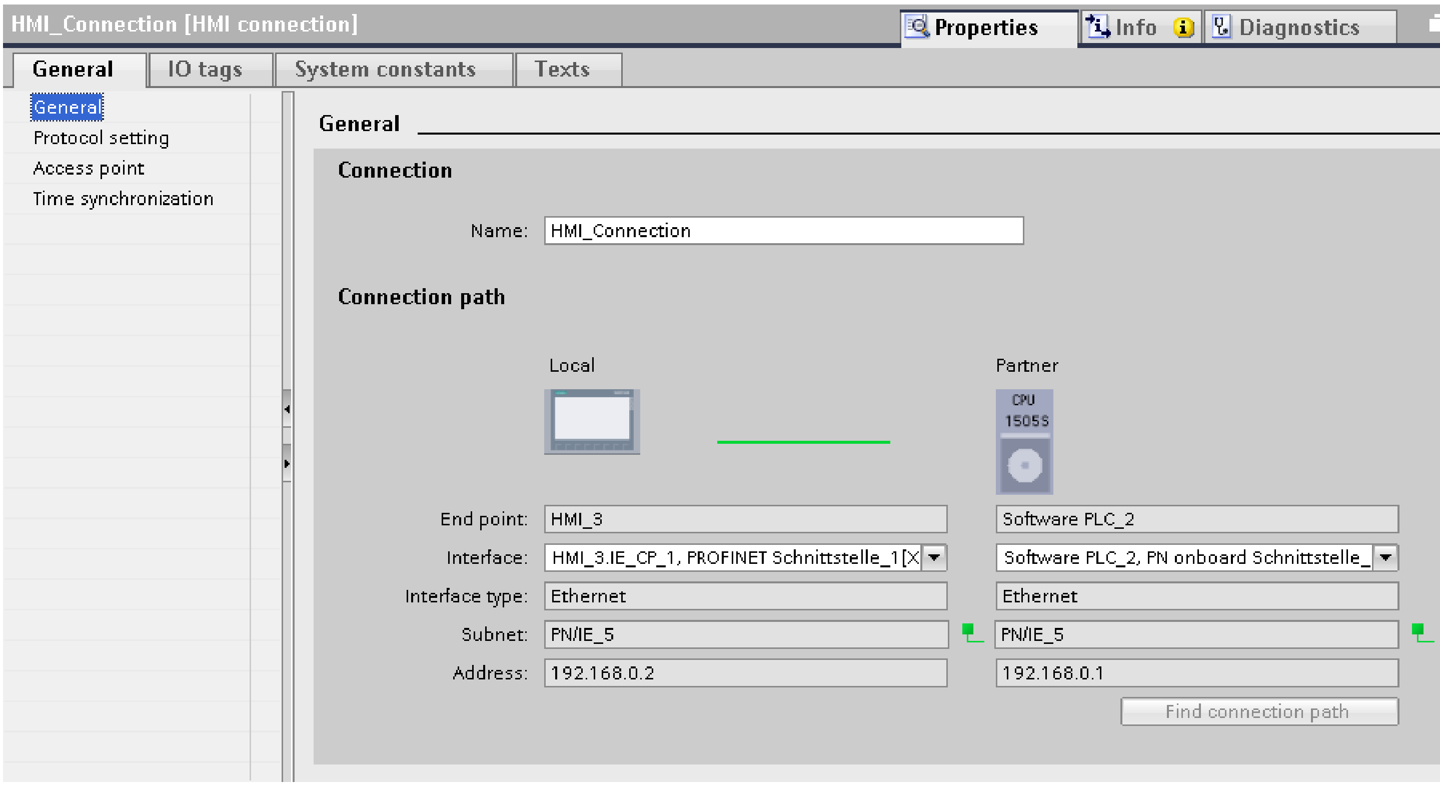
PROFINET parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and changing the HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

Shows the name of the HMI connection.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFINET parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area cannot be edited.
- "Subnet"
  Displays the selected subnet. This area cannot be edited.

- "Address"
  Displays the selected IP address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

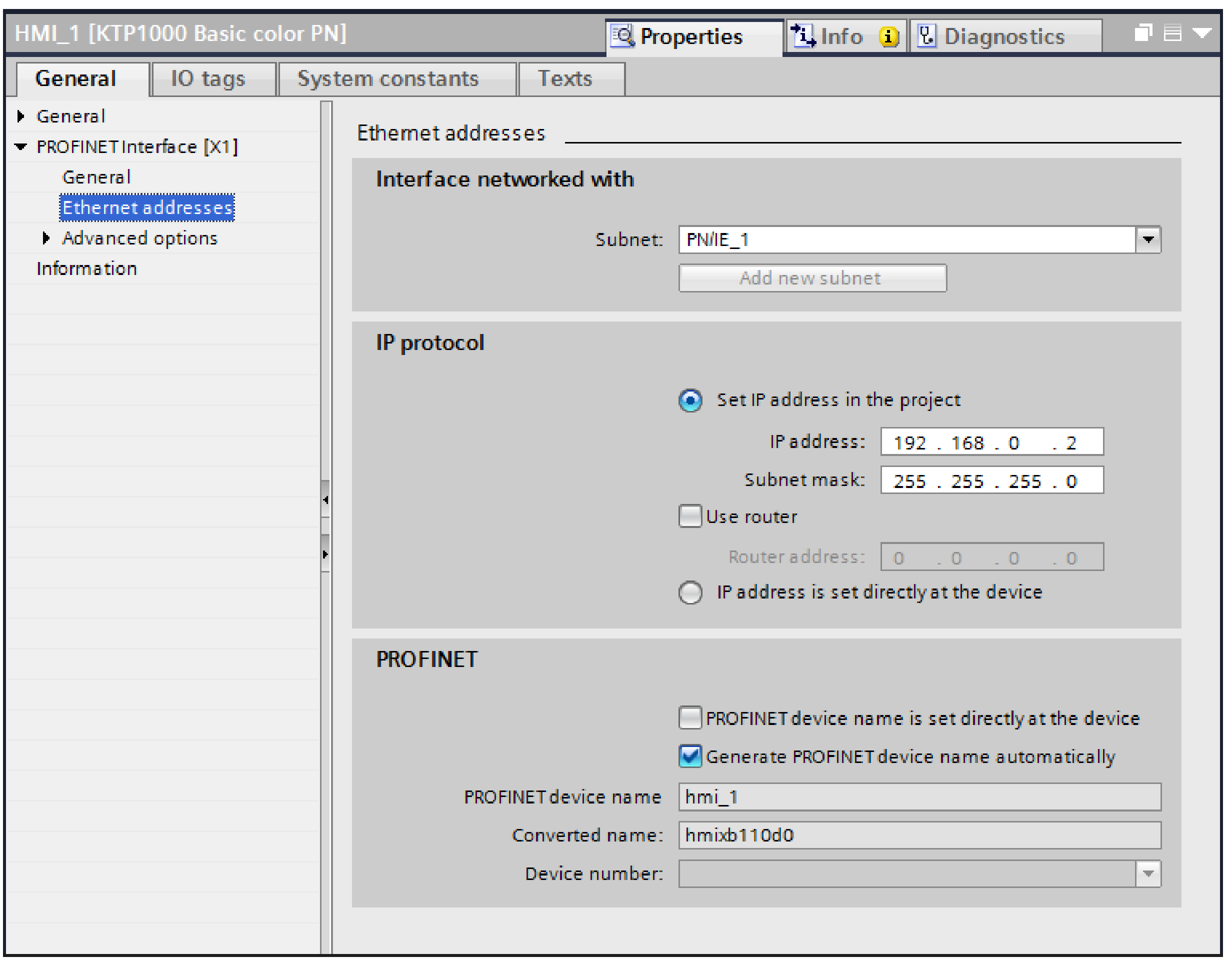
PROFINET parameters for the HMI device (Basic Panels)

PROFINET parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Set IP address in the project"
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

  Note

  The device is automatically restarted in the case of HMI devices with the Windows CE 3.0 operating system.

  HMI devices with Windows CE 3.0:

  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
- "Use IP router"
  If you are using an IP router, select "Use IP router" and enter the router address in the "Router address" field.
- "Set IP address using a different method"
  If the function "Set IP address using a different method" is activated, the IP address is not taken from the project. You have to enter the IP address directly in the Control Panel of the HMI device.

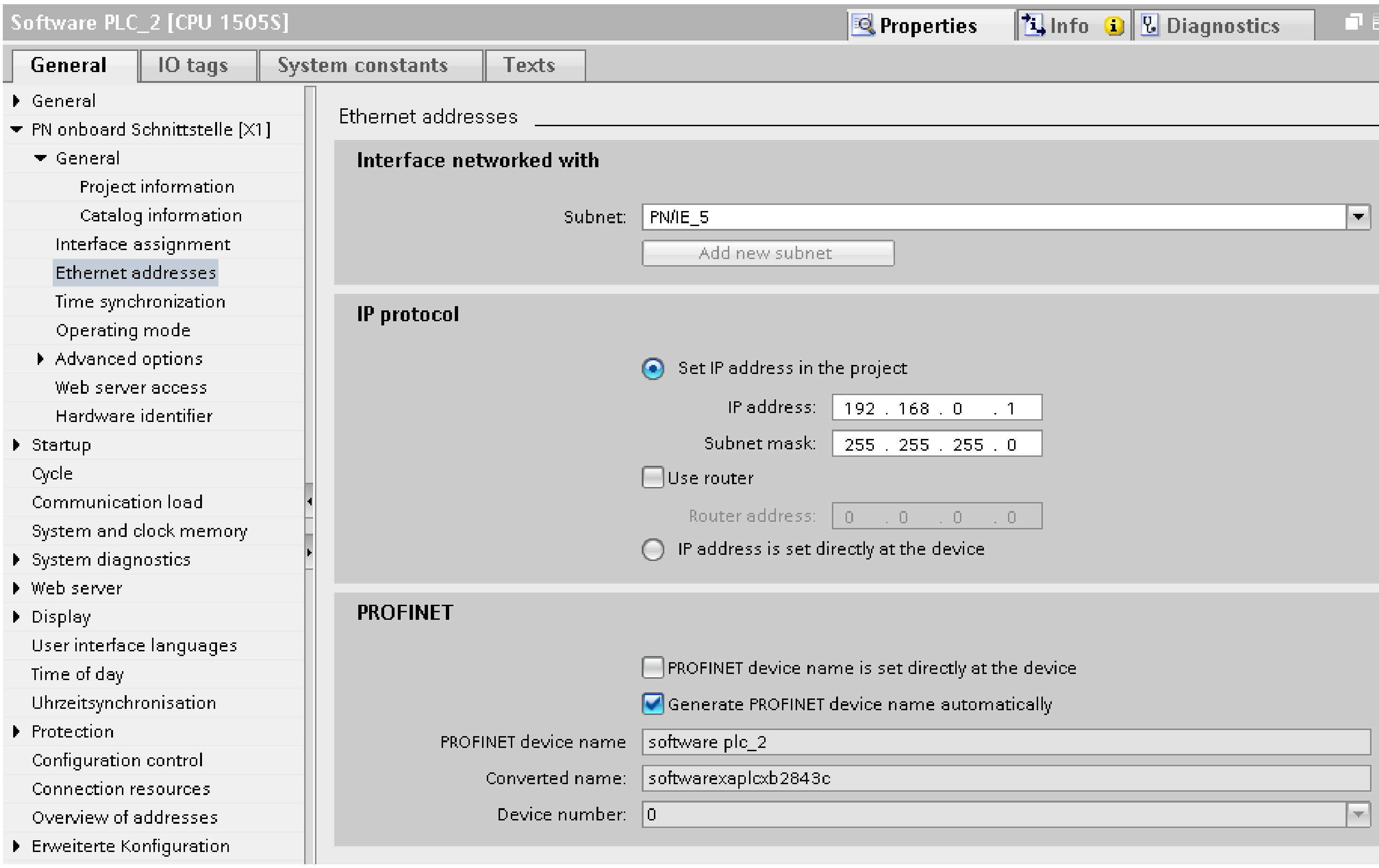
PROFINET parameters for the PLC (Basic Panels)

PROFINET parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFINET parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "IP address"
  You assign the IP address of the HMI device in the "IP address" area.
- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
  If you are using an IP router, select "Use IP router" and enter the router address in the field.

Protection of communication (Basic Panels)

Security levels (Basic Panels)

If you want to protect the PLC and HMI device communication, you can assign protection levels for the communication.

For an SIMATIC S7-1500 Software Controller, you can enter multiple passwords to set up different access rights for different user groups.

The passwords are entered in a table, so that exactly one protection level is assigned to each password.

The effect of the password is given in the "Protection" column.

Example

You select the "Complete protection" protection level for a standard CPU (i.e., not an F-CPU) when configuring it.

Afterwards, you enter a separate password for every protection level above it in the table.

For users who do not know any of the passwords, the CPU is completely protected. Not even HMI access is possible.

For users who know one of the assigned passwords, the effect depends on the table row in which the password occurs:

- The password in row 1 (no protection) allows access as if the CPU were completely unprotected. Users who know this password have unrestricted access to the CPU.
- The password in row 2 (write protection) allows access as if the CPU were write-protected. Despite knowing the password, users who know this password only have read access to the CPU.
- The password in row 3 (read and write protection) allows access as if the CPU were read-protected and write-protected, so that only HMI access is possible for users who know this password.

Access password for the HMI connection (Basic Panels)

Introduction

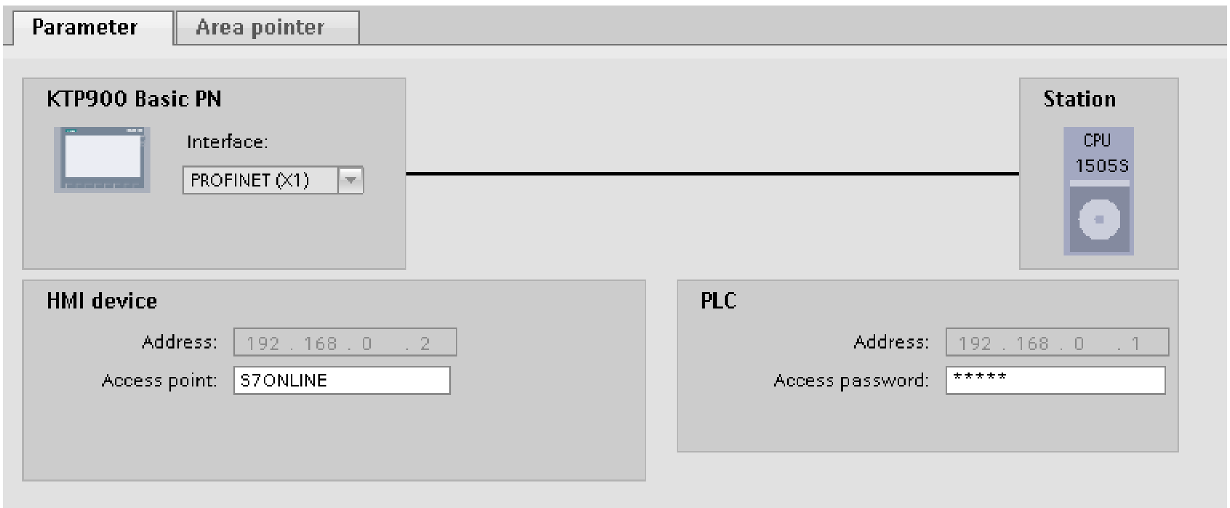
Communication with a PLC with the"Complete protection" protection level is protected by a password. The password is stored in the properties of the PLC.

Enter the password from the PLC in the "Access password" area.

Communication to the PLC is denied if an incorrect password or no password is entered.

Entering the access password

Enter the access password for the HMI connection in the "Connections" editor.



1.9.2.3 Setting port options (Basic Panels)

Setting the port options (Basic Panels)

Changing connection settings for the PROFINET IO port

You can change the network settings for the PROFINET IO port as required. By default, the settings are made automatically. In normal situations, this guarantees problem-free communication.

Possible settings for transmission rate / duplex

Depending on the selected device, you can make the following settings for "Transmission rate / duplex":

- Automatic setting
  Recommended default setting of the port. The transmission settings are automatically "negotiated" with the peer port. The "Enable autonegotiation" option is also enabled as a default, in other words, you can use cross cables or patch cables for the connection.
- TP/ITP at x Mbps full duplex (half duplex)
  Setting of the transmission rate and the full duplex/half duplex mode. The effectiveness depends on the "Enable autonegotiation" setting:
  - Autonegotiation enabled
    You can use both cross cable and patch cable.
  - Autonegotiation disabled
    Make sure that you use the correct cable (patch cable or cross cable)! The port is also monitored with this setting.
- Deactivated
  Depending on the module type, the drop down list box can contain the "- Disabled -" option. This option, for example, allows you to prevent access to an unused port for security reasons. With this setting, diagnostic events are not generated.

"Monitor" option

This option enables or disables port diagnostics. Examples of port diagnostics: The link status is monitored, in other words, the diagnostics are generated during link-down and the system reserve is monitored in the case of fiber optic ports.

Option "Enable autonegotiation "

The autonegotiation setting can only be changed if a concrete medium (for example, TP 100 Mbps full duplex) is selected. Whether or not a concrete medium can be set depends on the properties of the module.

If autonegotiation is disabled, this causes the port to be permanently specified, as for example, is necessary for a prioritized startup of the IO device.

You must make sure the partner port has the same settings because with this option the operating parameters of the connected network are not detected and the data transmission rate and transmission mode can accordingly not be optimally set.

Note

When a local port is connected, STEP 7 makes the setting for the partner port if the partner port supports the setting. If the partner port does not accept the setting, an error message is generated.

Wiring rules for disabled autonegotiation (Basic Panels)

Requirements

You have made the following settings for the port in question, for example, to accelerate the startup time of the IO device:

- Fixed transmission rate
- Autonegotiation incl. autocrossing disabled

The time for negotiating the transmission rate during startup has been saved.

If you have disabled autonegotiation, you must observe the wiring rules.

Wiring rules for disabled autonegotiation

PROFINET devices have the following two types of ports:

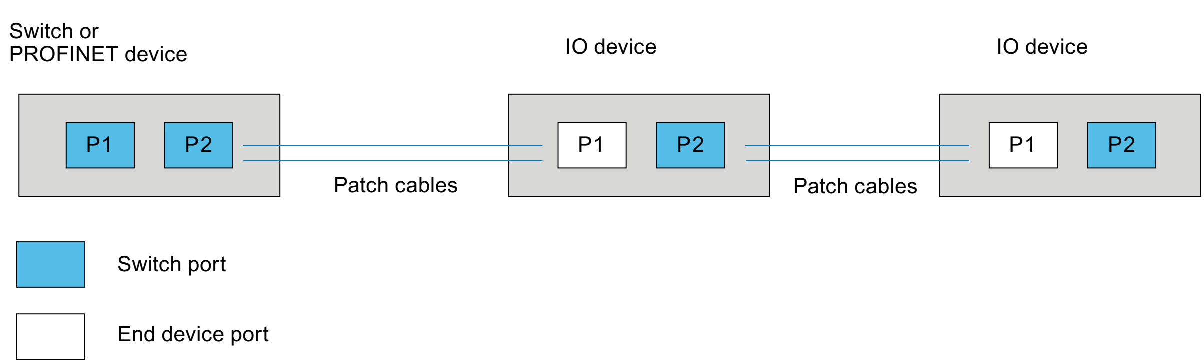
Type of port	PROFINET devices	Note
Switch port with crossed pin assignment	For IO devices: Port 2 For S7 CPUs with 2 ports: Ports 1 and 2	Crossed pin assignment means that the pin assignment for the ports for sending and receiving between the respective PROFINET devices is exchanged internally.
End device port with uncrossed pin assignment	For IO devices: Port 1 For S7 CPUs with one port: Port 1	-

Validity of the wiring rules

The cabling rules described in the following paragraph apply exclusively for the situation in which you have specified a fixed port setting.

Rules for cabling

You can connect several IO devices in line using a single cable type (patch cable). To do this, you connect port 2 of the IO device (distributed I/O) with port 1 of the next IO device. The following graphic gives an example with two IO devices.



Boundaries at the port (Basic Panels)

Requirements

To use boundaries, the respective device must have more than one port. If the PROFINET does not support boundary settings, they are not shown.

Enable boundaries

"Boundaries" are limits for transmission of certain Ethernet frames. The following boundaries can be set at a port:

- "End of discovery of accessible devices"
  No forwarding of DCP frames to identify accessible devices. Devices downstream from this port cannot be reached by the project tree under "Accessible devices". Devices downstream from this port cannot be reached by the CPU.
- "End of topology discovery"
  LLDP frames (Link Layer Discovery Protocol) are not forwarded for topology detection.
- "End of sync domain"
  No forwarding of sync frames transmitted to synchronize nodes within a sync domain.
  If you operate, for example, a PROFINET device with more than two ports in a ring, you should prevent the sync frame from being fed into the ring by setting a sync boundary (at the ports not inside the ring).
  Additional example: If you want to use several sync domains, configure a sync domain boundary for the port connected to a PROFINET device from the other sync domain.

Restrictions

The following restrictions must be observed:

- The individual check boxes can only be used if the port supports the function in question.
- If a partner port has been determined for the port, the following check boxes cannot be used:
  - "End of discovery of accessible devices"
  - "End of topology discovery"
- If autonegotiation is disabled, none of the check boxes can be used.

1.9.3 Performance features of communication (Basic Panels)

1.9.3.1 WinAC 1500 device dependency (Basic Panels)

Device dependency

If you use devices from an earlier version of the TIA Portal with TIA Portal V13 SP1, it may not be possible to configure connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC WinAC 1500
KP300 Basic	No
KP400 Basic	No
KTP400 Basic PN	No
KTP600 Basic DP	No
KTP600 Basic PN	No
KTP1000 Basic DP	No
KTP1000 Basic PN	No
TP1500 Basic PN	No

Basic Panels V12.0

HMI devices	SIMATIC WinAC 1500
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V13.0

HMI devices	SIMATIC WinAC 1500
KTP400 Basic	Yes
KTP700 Basic	Yes
KTP900 Basic	Yes

Basic Panels V13.0.1

HMI devices	SIMATIC WinAC 1500
KTP400 Basic	Yes
KTP700 Basic	Yes
KTP900 Basic	Yes

1.9.4 Configuring time synchronization (Basic Panels)

1.9.4.1 Time synchronization (Basic Panels)

Introduction

To have the same time of day throughout the plant, you can synchronize the time on various plant components using time synchronization. WinCC time synchronization is operated as a master-slave system.

One system component must be a clock for all components of a plant to work with identical time. The component functioning as the clock is referred to as the time master. The components that receive the time are time slaves.

Properties of time synchronization

- The HMI device can define the time as master or can accept the time of the PLC as slave.
- In "master mode", the time is synchronized at each connection setup.
- In "slave mode", the time is synchronized at each connection setup and than at cyclic intervals of 10 minutes.
- The first time synchronization is performed on the HMI device immediately after the start of runtime.
- Time synchronization is only performed on the HMI device during operation of runtime.

1.9.4.2 Time synchronization restrictions (Basic Panels)

Approved HMI devices

You can configure the time synchronization between a SIMATIC S7-1200 or SIMATIC S7-1500 and an HMI device with the following HMI devices:

Device	Operating system
Basic Panels	-
TP177 4"	Windows CE 5.0
Multi Panel 177	Windows CE 5.0
Multi Panel 277	Windows CE 5.0
Multi Panel 377	Windows CE 5.0
Mobile Panel 277	Windows CE 5.0
Mobile 277 IWLAN V2	Windows CE 5.0
Comfort Panels	Windows CE 6.0
PC systems with WinCC RT Advanced	Microsoft Windows XP Microsoft Windows 7

Configuration restrictions

- If an HMI device has several connections to SIMATIC S7-1200 or SIMATIC S7-1500, you can only configure one connection as "slave".
- If you have enabled time synchronization for the HMI device as "slave", you can no longer use the global area pointer "Date/time PLC".
- An HMI device can only request the time from a PLC with "Complete protection" security type configuration if the correct "Access password" is configured.
  Configure the "Access password" for communication with a PLC with "Complete protection" security type in the "Connections" editor of the HMI device.
  This "Access password" must match the password configured on the PLC. The PLC password is assigned in the PLC properties at: "General > Security"
- Basic Panels can only be configured as "Slave".
- If you use Basic Panels for the configuration, it is not possible to use time synchronization via NTP and the "Date/time PLC" area pointer simultaneously.
- Time synchronization with SIMATIC S7-1200 (V1.0) controllers is not possible.
- Time synchronization between the HMI device TP177 4" and SIMATIC S7-1200 (V4.0) controllers is not possible.
- Time synchronization between the HMI device TP177 4" and SIMATIC S7-1500 is not possible.

1.9.4.3 Configuring time synchronization for integrated connections

Introduction

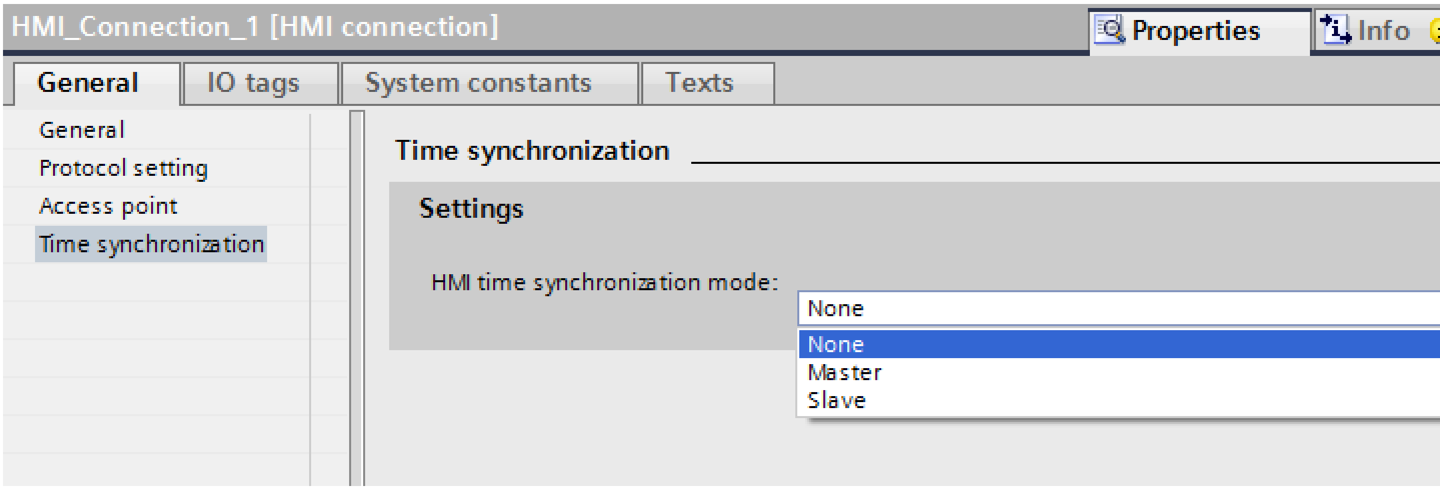
You configure time synchronization for an integrated connection in the "Devices & Networks" editor.

Requirements

- An HMI connection between an HMI device and a SIMATIC S7 1200 or SIMATIC S7 1500 has been configured.
- The HMI device must support the "time synchronization" function.
- The "Devices & Networks" editor is open.

Procedure

1. Click the line of the HMI connection in the "Devices & networks" editor.
2. Select the following in the inspector window under "General > Time synchronization > Settings":
   - None: No time synchronization is used.
   - Master: The HMI device sets the time.
   - Slave: The PLC sets the time.



1.9.4.4 Configuring time synchronization for non-integrated connections (Basic Panels)

Introduction

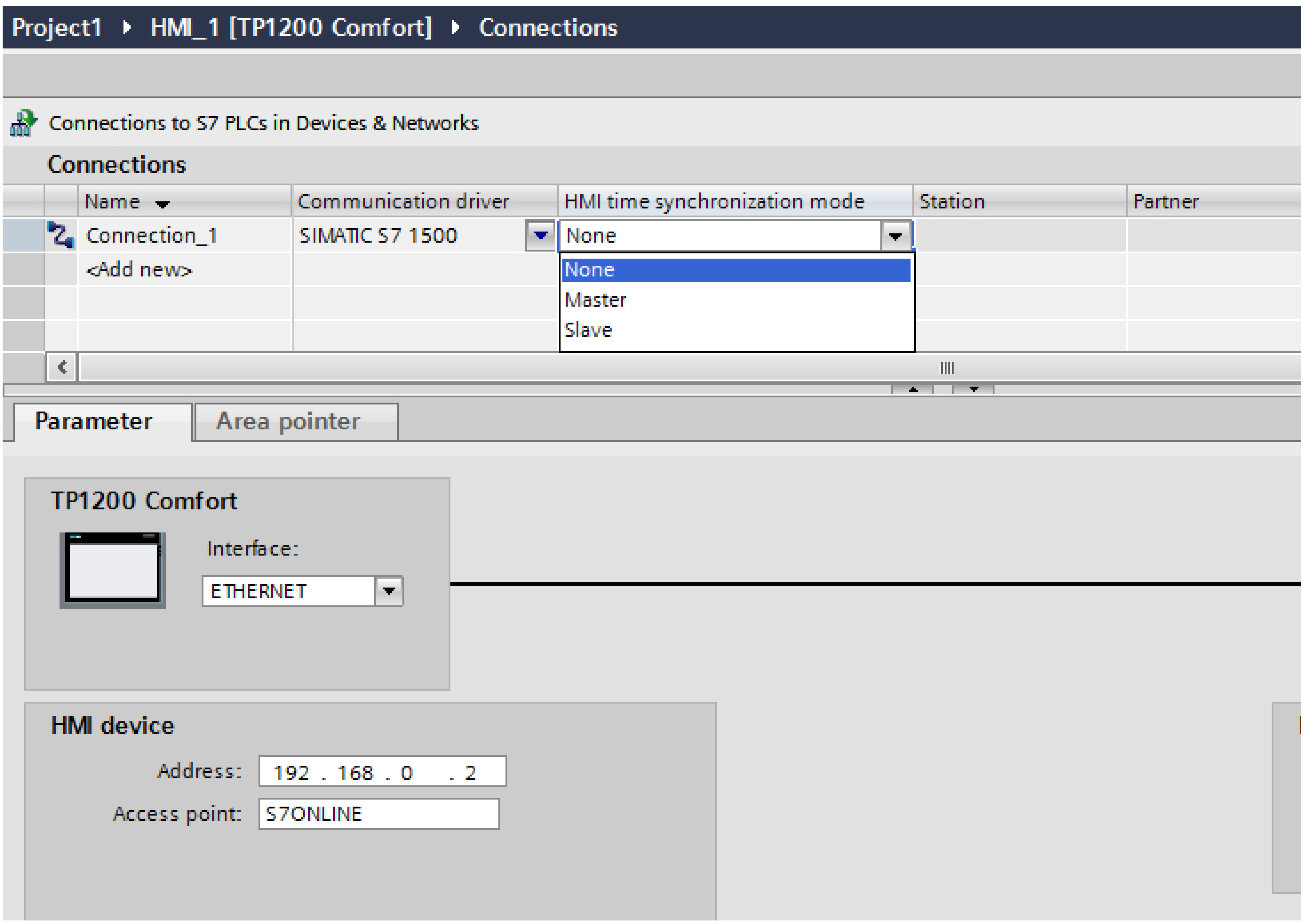
You configure time synchronization for a non-integrated connection in the "Connections" editor.

Requirements

- An HMI device which supports the "time synchronization" function has been created.
- "Connections" editor is open.

Procedure

1. Double-click "<Add>".
2. In the "Communication drivers" column, select the "SIMATIC S7 1500" PLC.
3. Select the following in the "HMI time synchronization mode" column:
   – None: No time synchronization is used.
   – Master: The HMI device sets the time.
   – Slave: The PLC sets the time.



1.10 Communicating with SIMATIC ET 200 CPU (Basic Panels)

1.10.1 Communication with SIMATIC ET 200 CPU (Basic Panels)

Introduction

This section describes the communication between an HMI device and the SIMATIC ET 200 CPU controller.

You can configure the following communication channels for the SIMATIC ET 200 CPU:

- PROFINET
- PROFIBUS

HMI connection for communication

Configure connections between the HMI device and a SIMATIC ET 200 CPU in the "Devices & Networks" editor. If you have configured a HMI device with a serial port, you must configure a PROFIBUS-capable communication module to the ET 200 CPU.

1.10.2 Communication via PROFINET (Basic Panels)

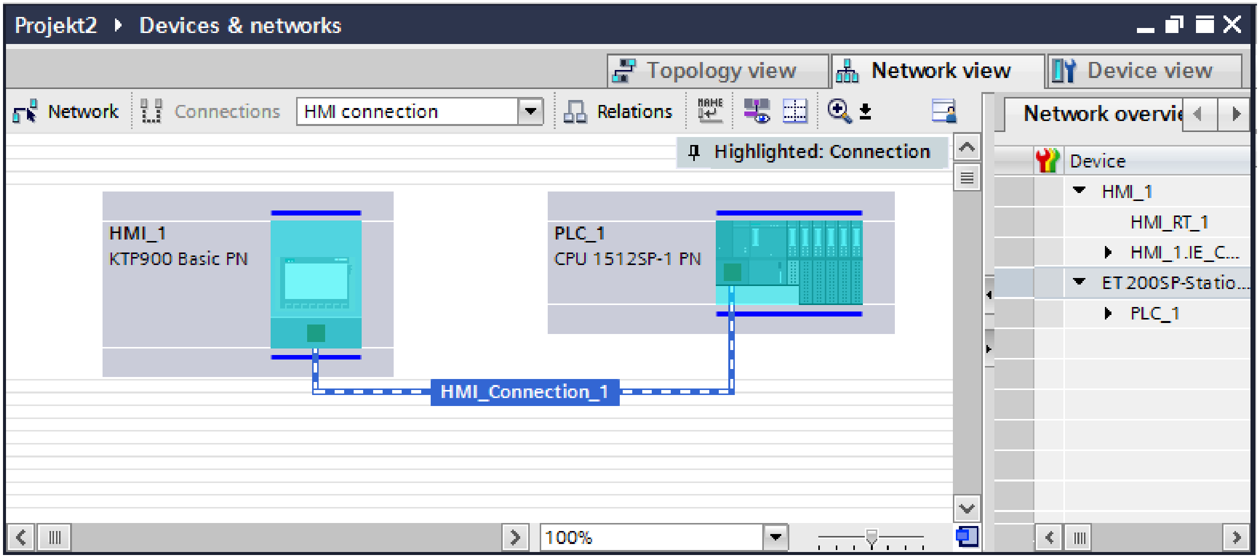
1.10.2.1 Communication via PROFINET (Basic Panels)

HMI connections via PROFINET

If you have inserted an HMI device and a SIMATIC ET 200 CPU into the project, you interconnect the two PROFINET interfaces in the "Devices & Networks" editor.

You can also connect multiple HMI devices to a single SIMATIC ET 200 CPU and multiple SIMATIC ET 200 CPU to a single HMI device.

The maximum number of communication partners that you can connect to an HMI device is dependent on the HMI device used.



Additional information is available in the documentation for the respective HMI device.

HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFINET in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device via PROFINET in the "Connections" editor of the HMI device.

1.10.2.2 Configuring an HMI connection via PROFINET (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC ET 200 CPU via PROFINET or Ethernet in the "Devices & Networks" editor.

⚠ CAUTION
Communication via Ethernet In Ethernet-based communication, the end user is responsible for the security of his data network. Targeted attacks can overload the device and interfere with proper functioning.

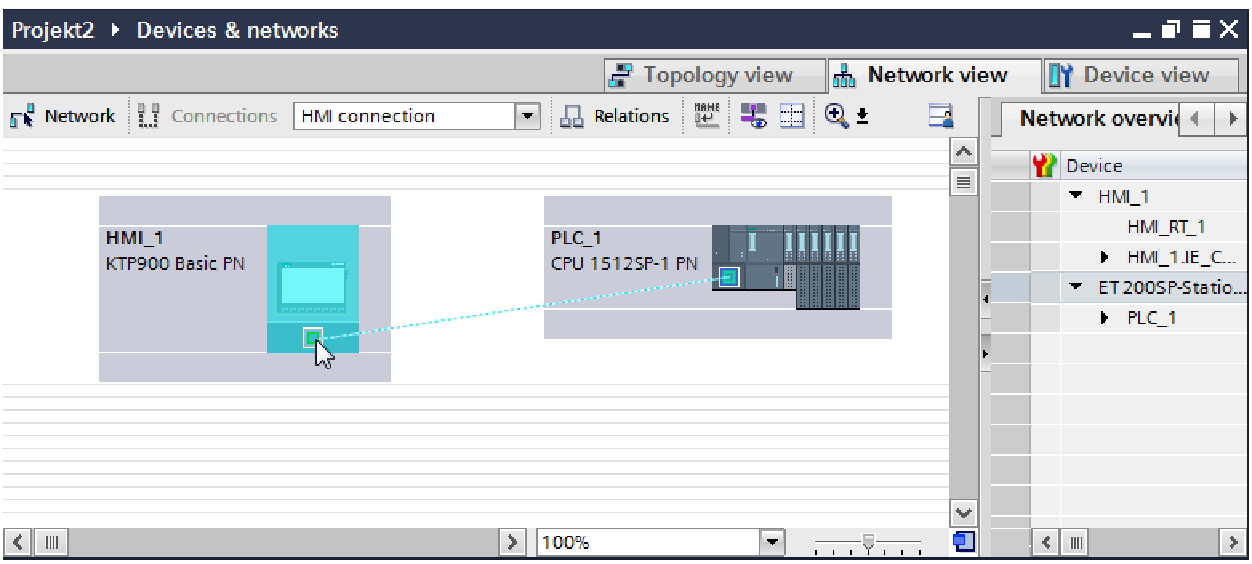
Requirements

The following communication partners are created in the "Devices & Networks" editor:

- SIMATIC ET 200 CPU
- HMI device with PROFINET or Ethernet interface

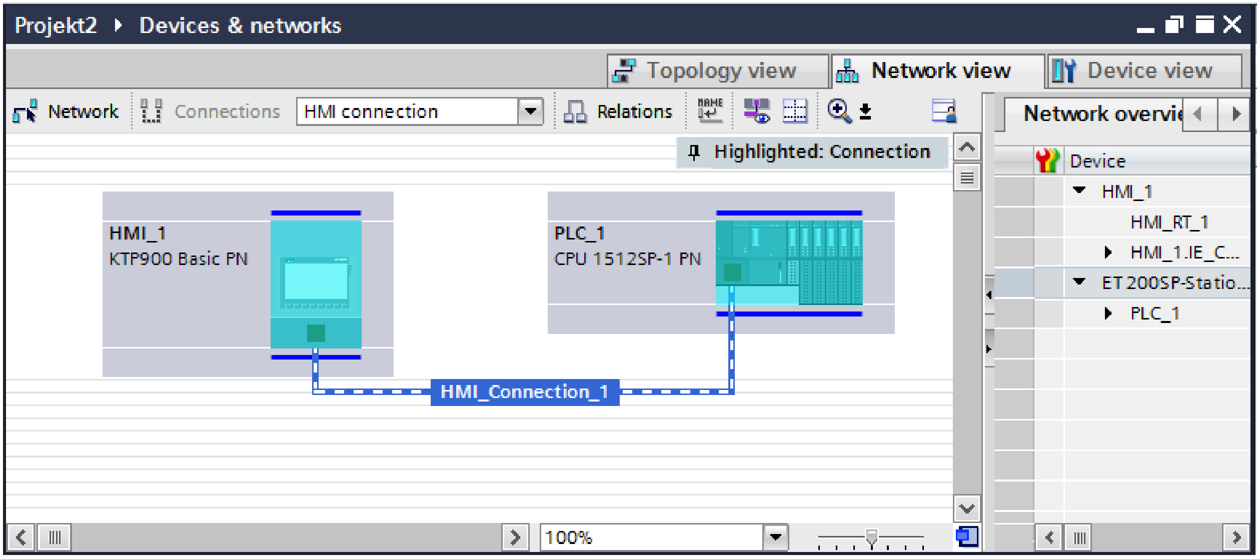
Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button and select "HMI connection" for the connection type.
   The devices available for connection are highlighted in color.
3. Click the PROFINET interface of the PLC and use a drag-and-drop operation to draw a connection to the PROFINET or Ethernet interface of the HMI device.



4. Click the connecting line.

5. Click "Highlight HMI connection" and select the HMI connection.



The connection is displayed graphically in the Inspector window.

6. Click the communication partners in the "Network view" and change the PROFINET parameters in the Inspector window according to the requirements of your project. See the section "PROFINET parameters (Page 258)" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created a connection between an HMI device and a SIMATIC ET 200 CPU.

The IP address and subnet mask connection parameters are configured.

1.10.2.3 PROFINET parameters (Basic Panels)

PROFINET parameters for the HMI connection (Basic Panels)

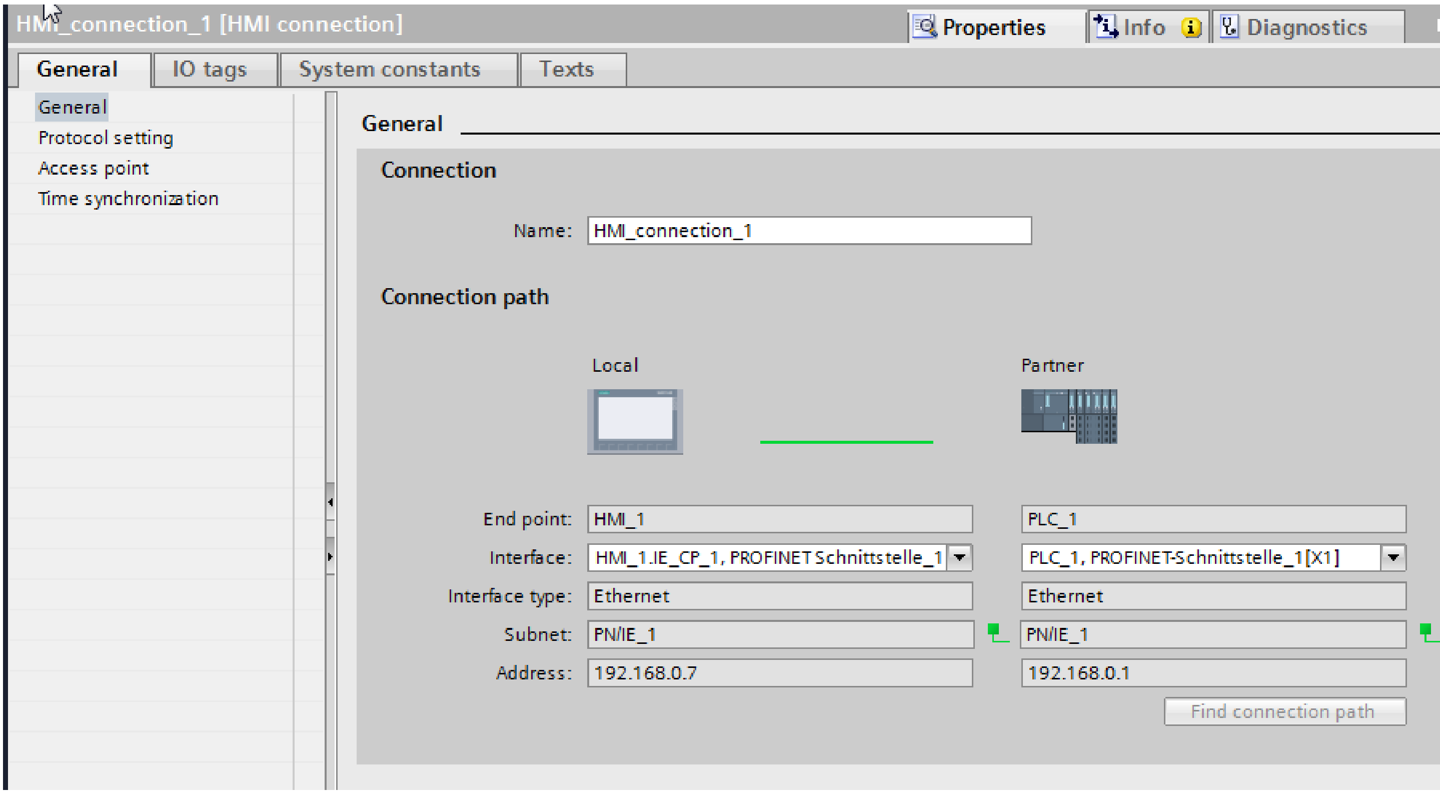
PROFINET parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and editing HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



Connection

The "Connection" area displays the HMI connection created for communication between the devices.

You can edit the name of the HMI connection in this area.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFINET parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"
  Displays the selected interface type. This area cannot be edited.

- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the selected IP address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

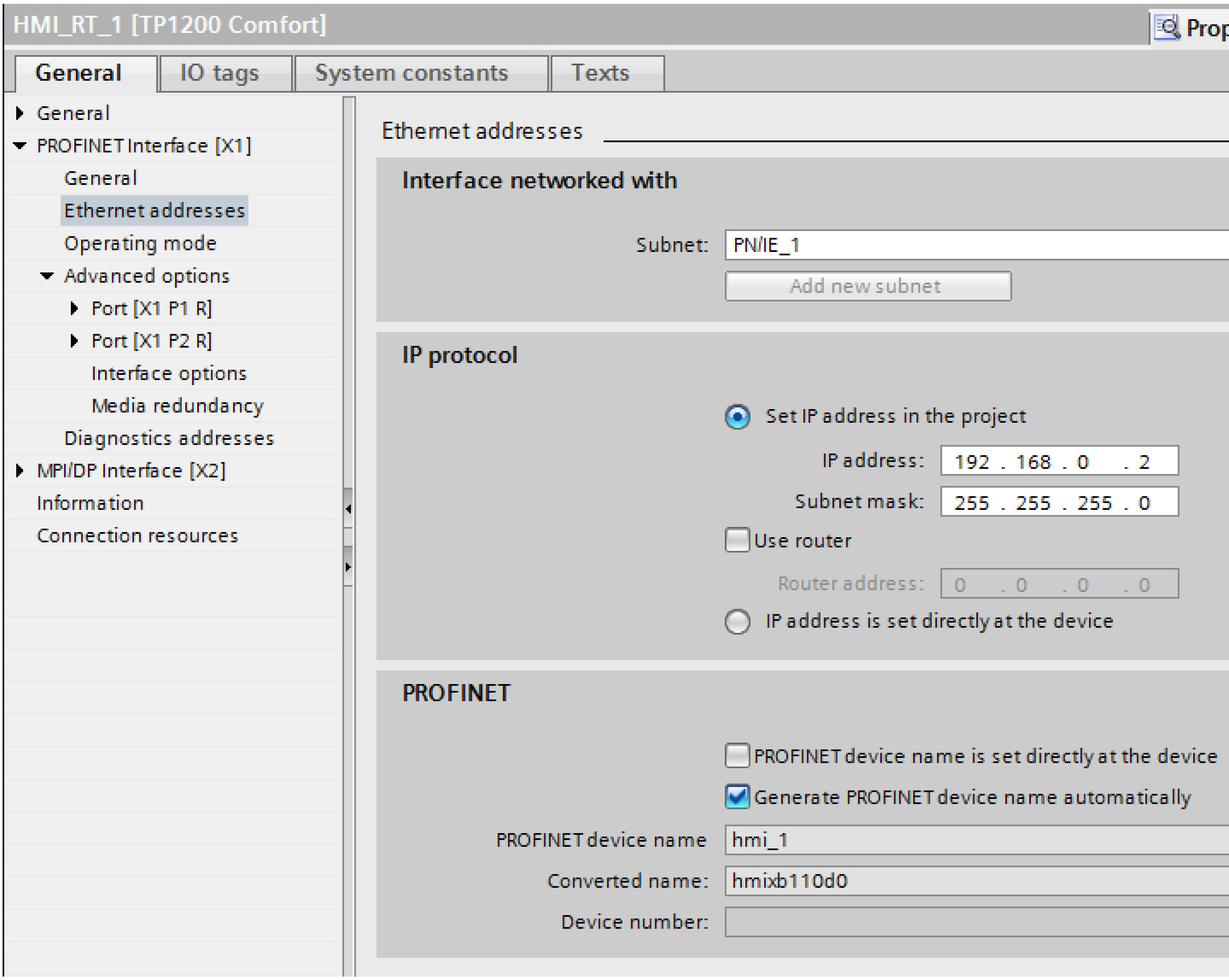
PROFINET parameters for the HMI device (Basic Panels)

PROFINET parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFINET parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Set IP address in the project"
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.

  Note

  The device is automatically restarted in the case of HMI devices with the Windows CE 3.0 operating system.

  HMI devices with Windows CE 3.0:
  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
- "Use IP router"
  If you are using an IP router, select "Use IP router" and enter the router address in the "Router address" field.
- "Set IP address using a different method"
  If the function "Set IP address using a different method" is activated, the IP address is not taken from the project. You have to enter the IP address directly in the Control Panel of the HMI device.

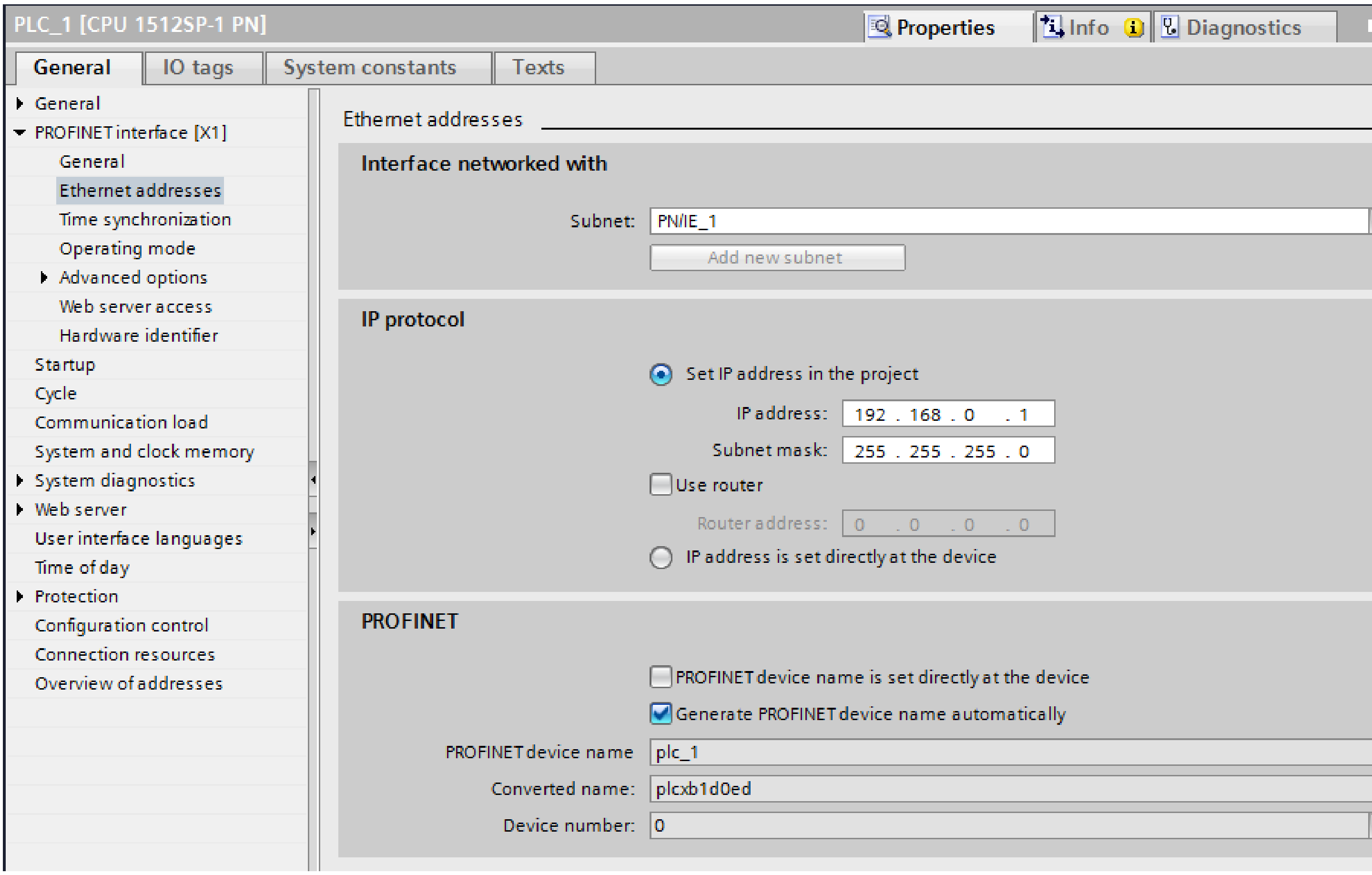
PROFINET parameters for the PLC (Basic Panels)

PROFINET parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFINET parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"IP protocol"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "IP address"
  You assign the IP address of the HMI device in the "IP address" area.
- "Subnet mask"
  You assign data of the subnet mask in the "Subnet mask" area.
  If you are using an IP router, select "Use IP router" and enter the router address in the field.

Configuring Industrial Ethernet (Basic Panels)

Rules for the network configuration

The Ethernet interfaces of the devices have a default IP address that you can change.

IP address

The IP parameters are visible if the communication-capable devices support the TCP/IP protocol.

The IP address consists of 4 decimal figures in the range of 0 to 255. The decimal figures are separated from one another by a dot.

Example: 140.80.0.2

The IP address consists of the following:

- The address of the (sub) net
- The address of the node (generally also called host or network node)

Subnet mask

The subnet mask splits these two addresses. It determines which part of the IP address addresses the network and which part of the IP address addresses the node.

The set bits of the subnet mask determine the network part of the IP address.

Example:

Subnet mask: 255.255.0.0 = 11111111.11111111.00000000.00000000

In the example given for the above IP address, the subnet mask shown here has the following meaning:

The first 2 bytes of the IP address identify the subnet - i.e. 140.80. The last two bytes address the node, thus 0.2.

It is generally true that:

- The network address results from AND linking the IP address and subnet mask.
- The node address results from AND NOT linking the IP address and subnet mask.

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. It determines 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 of between 224 and 255 is also possible for the first decimal number of the IP address (address class D etc). This is, however, 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 lower points of the subnet mask to "1". For each bit set to "1", the number of "private" networks doubles and the number of nodes they contain is halved. Externally, the network functions like an individual network as it did previously.

Example:

You have a subnet of address class B (e.g. 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 nodes with addresses between 129.80.001.xxx and 129.80.127.xxx are on one subnet, all nodes with addresses between 129.80.128.xxx and 129.80.255.xxx are on another subnet.

Router

The job of the routers is to connect the subnets. If an IP datagram is to be sent to another network, it first has to be conveyed to a router. To make this possible, in this case you have to enter the address of the router for each node in the subnet.

The IP address of a node in the subnet and the address of the router may only differ at the points at which there is a "0" in the subnet mask.

1.10.2.4 Setting port options (Basic Panels)

Setting the port options (Basic Panels)

Changing connection settings for the PROFINET IO port

You can change the network settings for the PROFINET IO port as required. By default, the settings are made automatically. In normal situations, this guarantees problem-free communication.

Possible settings for transmission rate / duplex

Depending on the selected device, you can make the following settings for "Transmission rate / duplex":

- Automatic setting
  Recommended default setting of the port. The transmission settings are automatically "negotiated" with the peer port. The "Enable autonegotiation" option is also enabled as a default, in other words, you can use cross cables or patch cables for the connection.
- TP/ITP at x Mbps full duplex (half duplex)
  Setting of the transmission rate and the full duplex/half duplex mode. The effectiveness depends on the "Enable autonegotiation" setting:
  – Autonegotiation enabled
    You can use both cross cable and patch cable.
  – Autonegotiation disabled
    Make sure that you use the correct cable (patch cable or cross cable)! The port is also monitored with this setting.
- Deactivated
  Depending on the module type, the drop down list box can contain the "- Disabled -" option. This option, for example, allows you to prevent access to an unused port for security reasons. With this setting, diagnostic events are not generated.

"Monitor" option

This option enables or disables port diagnostics. Examples of port diagnostics: The link status is monitored, in other words, the diagnostics are generated during link-down and the system reserve is monitored in the case of fiber optic ports.

Option "Enable autonegotiation "

The autonegotiation setting can only be changed if a concrete medium (for example, TP 100 Mbps full duplex) is selected. Whether or not a concrete medium can be set depends on the properties of the module.

If autonegotiation is disabled, this causes the port to be permanently specified, as for example, is necessary for a prioritized startup of the IO device.

You must make sure the partner port has the same settings because with this option the operating parameters of the connected network are not detected and the data transmission rate and transmission mode can accordingly not be optimally set.

Note

When a local port is connected, STEP 7 makes the setting for the partner port if the partner port supports the setting. If the partner port does not accept the setting, an error message is generated.

Wiring rules for disabled autonegotiation (Basic Panels)

Requirements

You have made the following settings for the port in question, for example, to accelerate the startup time of the IO device:

- Fixed transmission rate
- Autonegotiation incl. autocrossing disabled

The time for negotiating the transmission rate during startup has been saved.

If you have disabled autonegotiation, you must observe the wiring rules.

Wiring rules for disabled autonegotiation

PROFINET devices have the following two types of ports:

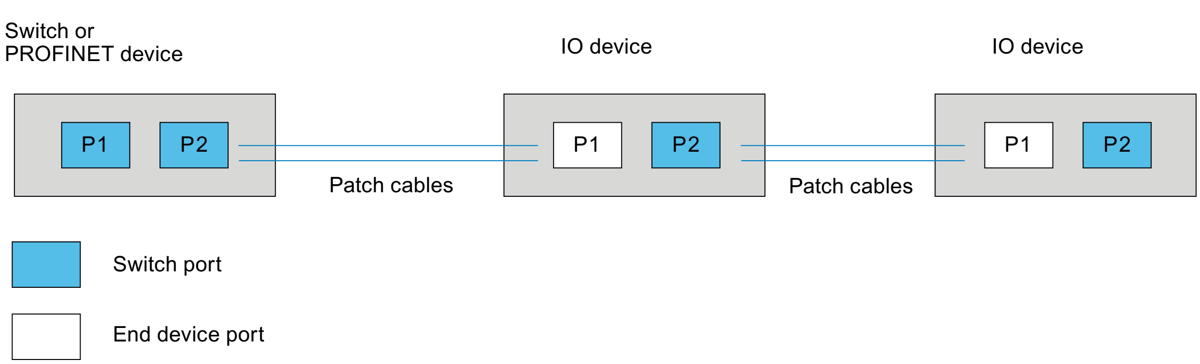
Type of port	PROFINET devices	Note
Switch port with crossed pin assignment	For IO devices: Port 2 For S7 CPUs with 2 ports: Ports 1 and 2	Crossed pin assignment means that the pin assignment for the ports for sending and receiving between the respective PROFINET devices is exchanged internally.
End device port with uncrossed pin assignment	For IO devices: Port 1 For S7 CPUs with one port: Port 1	-

Validity of the wiring rules

The cabling rules described in the following paragraph apply exclusively for the situation in which you have specified a fixed port setting.

Rules for cabling

You can connect several IO devices in line using a single cable type (patch cable). To do this, you connect port 2 of the IO device (distributed I/O) with port 1 of the next IO device. The following graphic gives an example with two IO devices.



Boundaries at the port (Basic Panels)

Requirements

To use boundaries, the respective device must have more than one port. If the PROFINET does not support boundary settings, they are not shown.

Enable boundaries

"Boundaries" are limits for transmission of certain Ethernet frames. The following boundaries can be set at a port:

- "End of discovery of accessible devices"
  No forwarding of DCP frames to identify accessible devices. Devices downstream from this port cannot be reached by the project tree under "Accessible devices". Devices downstream from this port cannot be reached by the CPU.
- "End of topology discovery"
  LLDP frames (Link Layer Discovery Protocol) are not forwarded for topology detection.
- "End of sync domain"
  No forwarding of sync frames transmitted to synchronize nodes within a sync domain.
  If you operate, for example, a PROFINET device with more than two ports in a ring, you should prevent the sync frame from being fed into the ring by setting a sync boundary (at the ports not inside the ring).
  Additional example: If you want to use several sync domains, configure a sync domain boundary for the port connected to a PROFINET device from the other sync domain.

Restrictions

The following restrictions must be observed:

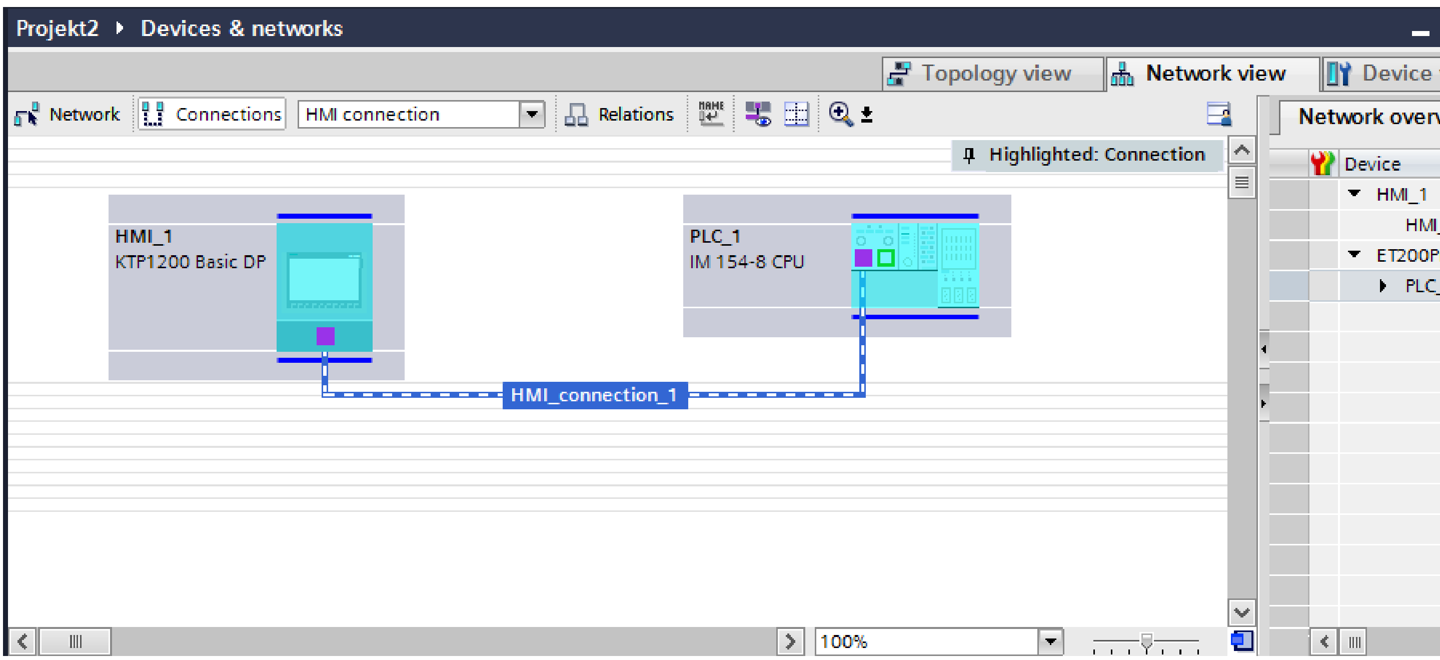
- The individual check boxes can only be used if the port supports the function in question.
- If a partner port has been determined for the port, the following check boxes cannot be used:
  - "End of discovery of accessible devices"
  - "End of topology discovery"
- If autonegotiation is disabled, none of the check boxes can be used.

1.10.3 Communication via PROFIBUS (Basic Panels)

1.10.3.1 Communication via PROFIBUS (Basic Panels)

HMI connections via PROFIBUS

If you want to connect a SIMATIC ET 200 CPU to a HMI device via PROFIBUS, you must configure a PROFIBUS-capable communication module to a slot of the controller first.



HMI connection in the "Devices & Networks" editor

You configure the HMI connection between the PLC and the HMI device via PROFIBUS in the "Devices & Networks" editor.

Connection in the "Connections" editor

Alternatively, you configure the connection between the PLC and HMI device in the "Connections" editor of the HMI device.

1.10.3.2 Configuring an HMI connection via PROFIBUS (Basic Panels)

Introduction

You configure an HMI connection between HMI devices and a SIMATIC ET 200 CPU via PROFIBUS in the "Devices & Networks" editor.

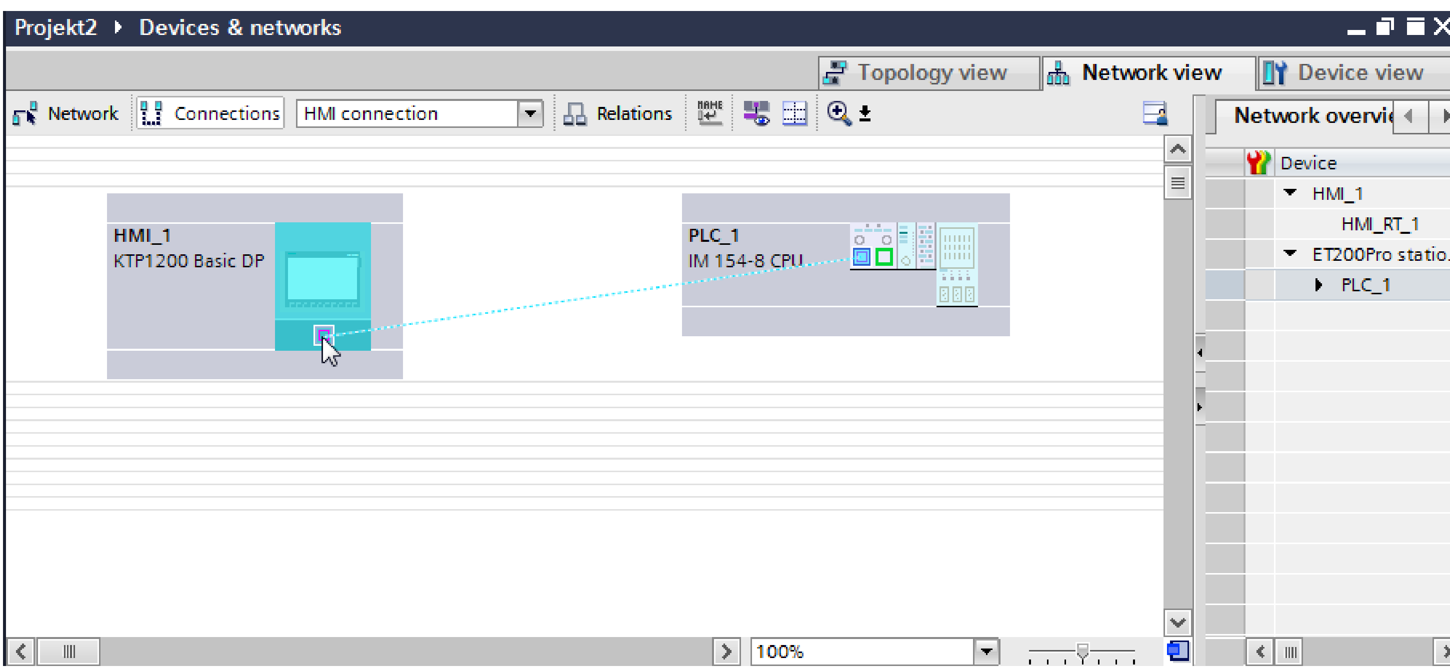
Requirements

The following communication partners are created in the "Devices & Networks" editor:

- HMI device with MPI/DP interface
- SIMATIC ET 200 CPU with DP interface

Procedure

1. Double-click the "Devices & Networks" item in the project tree.
   The available communication partners in the project are displayed graphically in the network view.
2. Click the "Connections" button.
   The devices available for connection are highlighted in color.
3. Click the HMI device interface.
4. Select the "PROFIBUS" interface type in the Inspector window under "Properties > General > PROFIBUS address/ MPI address > Parameters".
5. Click the interface of the CPU and use a drag-and-drop operation to draw a connection to the HMI device.



6. Click the name of the connection.
   The connection is displayed graphically in the Inspector window.

7. Click "Highlight HMI connection" and select the HMI connection.
8. Click the communication partners in the "Network view" and change the PROFIBUS parameters in the Inspector window according to the requirements of your project. See the section "PROFIBUS parameters (Page 271)" for additional details.

Note

The created HMI connection is also shown in the tabular area of the editor on the "Connections" tab. You check the connection parameters in the table.

You can change the local name for the connection only in the table.

Result

You have created an HMI connection between an HMI device and a SIMATIC ET 200 CPU via PROFIBUS.

1.10.3.3 PROFIBUS parameters (Basic Panels)

PROFIBUS parameters for the HMI connection (Basic Panels)

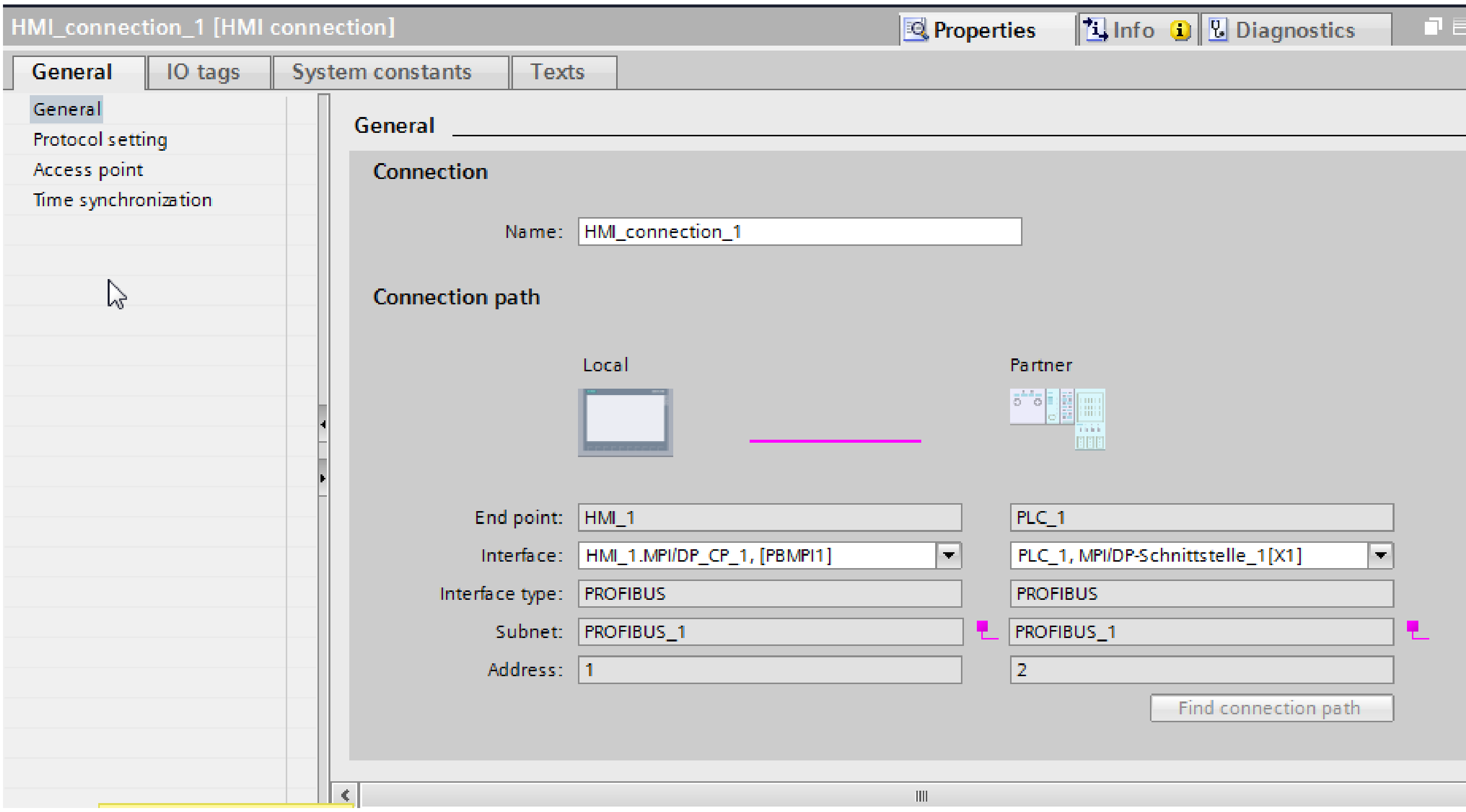
PROFIBUS parameters for the HMI connection

An overview of the configured HMI connection parameters can be found in the properties for the HMI connection.

Only limited changes are possible in this Inspector window.

Displaying and editing HMI connection parameters

1. Click the HMI connection in the "Devices & Networks" editor.
2. Change the parameters of the HMI connection in the Inspector window under "Properties > General > General".



"Connection"

The "Connection" area displays the HMI connection created for communication between the devices.

You can edit the name of the HMI connection in this area.

"Connection path"

The communication partners of the selected HMI connection and the associated PROFIBUS parameters are displayed in the "Connection path" area. Some of the areas displayed cannot be edited in this dialog.

- "End point"
  Displays the device name. This area cannot be edited.
- "Interface"
  Displays the selected interface of the device. You can choose between several interfaces, depending on the device.
- "Interface type"

Displays the selected interface type. This area cannot be edited.

- "Subnet"
  Displays the selected subnet. This area cannot be edited.
- "Address"
  Displays the PROFIBUS address of the device. This area cannot be edited.
- "Find connection path" button
  Enables the subsequent specification of connections.

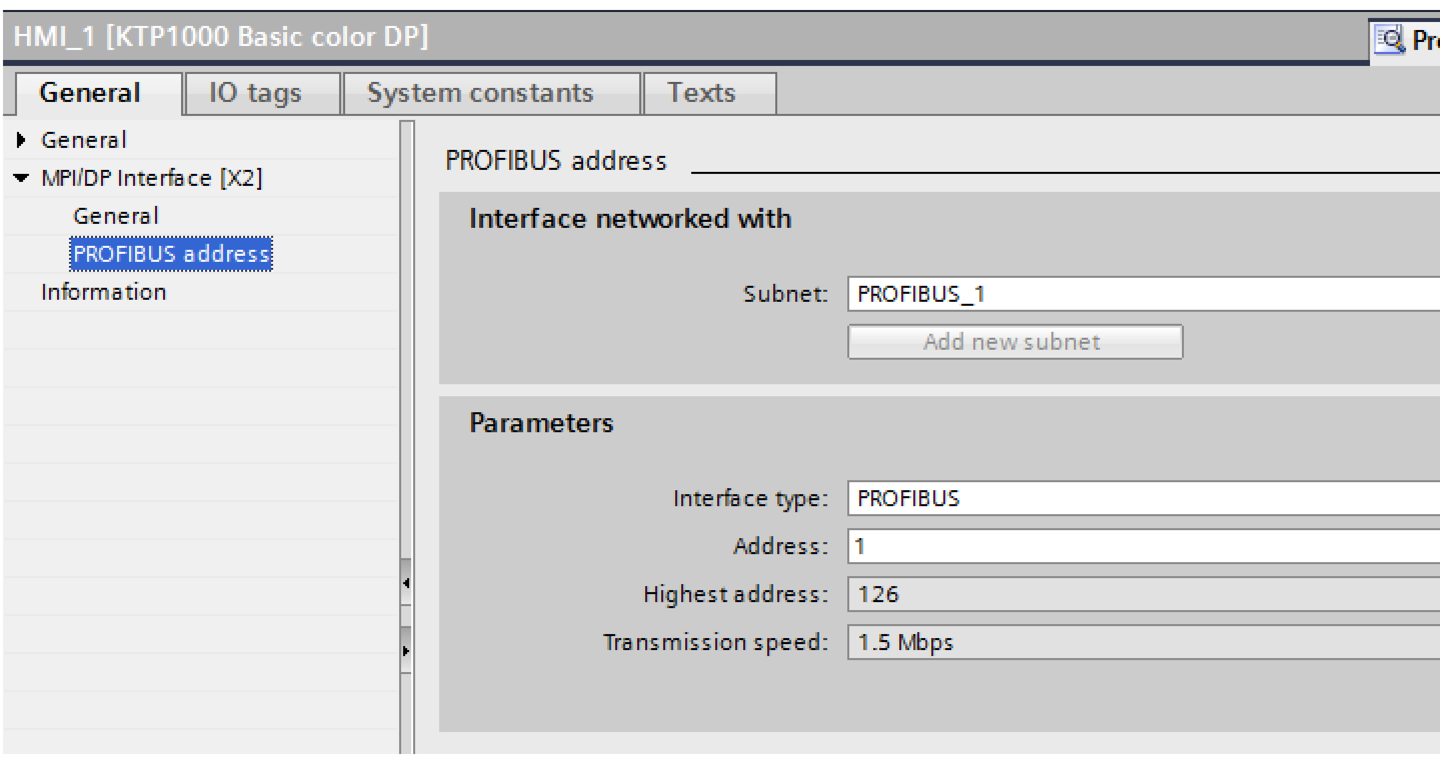
PROFIBUS parameters for the HMI device (Basic Panels)

PROFIBUS parameters for the HMI device

An overview of the configured HMI device parameters can be found in the properties for the HMI device.

Displaying and changing PROFIBUS parameters of the HMI device

1. Click the HMI device in the "Devices & Networks" editor.
2. Change the parameters of the HMI device in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Interface networked with" area, select the subnet of the HMI connection via which the HMI device is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.
- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

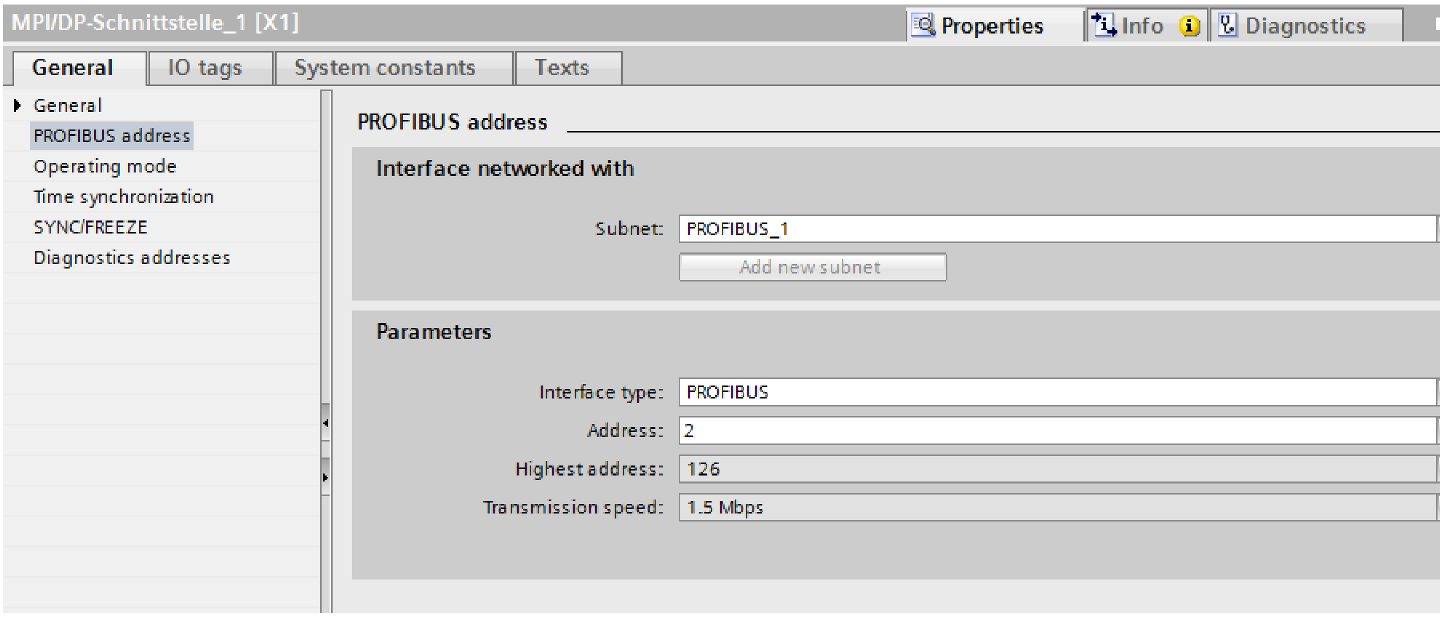
PROFIBUS parameters for the PLC (Basic Panels)

PROFIBUS parameters for the PLC

An overview of the configured parameters can be found in the properties for the PLC.

Displaying and changing PROFIBUS parameters of the PLC

1. Click the PLC in the "Devices & Networks" editor.
2. Change the parameters of the PLC in the Inspector window under "Properties > General > General".



"Interface networked with"

In the "Subnet" area, select the subnet of the HMI connection via which the PLC is connected to the network. You use the "Add new subnet" button to create a new subnet.

"Parameters"

- "Interface type"
  Depending on the HMI device type, you have various interfaces to choose from.
- "Address"
  You assign the PROFIBUS address of the HMI device in the "Address" area. The PROFIBUS address must be unique throughout the PROFIBUS network.
- "Highest address"
  The "Highest address" area displays the highest address of the PROFIBUS network.
- "Transmission speed"
  The "Transmission speed" is determined by the slowest device connected to the network. The setting is identical throughout the network.

1.10.4 Performance features of communication (Basic Panels)

1.10.4.1 Device dependencies of SIMATIC ET 200 CPU (Basic Panels)

Communication with SIMATIC ET 200 CPU controller

If you use devices from an earlier version of TIA Portal with TIA Portal V12 SP1, it may not be possible to configure integrated connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC ET 200 CPU
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V12.0

HMI devices	SIMATIC ET 200 CPU
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V13.0

HMI devices	SIMATIC ET 200 CPU
KTP400 Basic PN	Yes
KTP700 Basic PN / DP	Yes

HMI devices	SIMATIC ET 200 CPU
KTP900 Basic PN	Yes
KTP1200 Basic PN / DP	Yes

Basic Panels V13.0.1

HMI devices	SIMATIC ET 200 CPU
KTP400 Basic PN	Yes
KTP700 Basic PN / DP	Yes
KTP900 Basic PN	Yes
KTP1200 Basic PN / DP	Yes

Panels V11.0

HMI devices	SIMATIC ET 200 CPU
OP 73	Yes
OP 77A	Yes
OP 77B	Yes
TP 177A	Yes
TP 177A Portrait	Yes
TP 177B 4"	Yes
TP 177B 6" mono	Yes
TP 177B 6"	Yes
OP 177B 6" mono	Yes
OP 177B 6"	Yes
TP 277 6"	Yes
OP 277 6"	Yes

Multi Panels V11.0

HMI devices	SIMATIC ET 200 CPU
MP 177 6" Touch	Yes
MP 277 8" Key	Yes
MP 277 10" Key	Yes
MP 277 10" Touch	Yes
MP 377 12" Key	Yes
MP 377 12" Touch	Yes
MP 377 15" Touch	Yes
MP 377 19" Touch	Yes

Multi Panels V12.0

HMI devices	SIMATIC ET 200 CPU
MP 177 6" Touch	Yes
MP 277 8" Key	Yes
MP 277 10" Key	Yes
MP 277 10" Touch	Yes
MP 377 12" Key	Yes
MP 377 12" Touch	Yes
MP 377 15" Touch	Yes
MP 377 19" Touch	Yes

Mobile Panels V11.0

HMI devices	SIMATIC ET 200 CPU
Mobile Panel 177 6" DP	Yes
Mobile Panel 177 6" PN	Yes
Mobile Panel 277 8"	Yes
Mobile Panel 277 8" IWLAN V2	Yes
Mobile Panel 277F 8" IWLAN V2	Yes
Mobile Panel 277F 8" IWLAN V2 (RFID tag)	Yes
Mobile Panel 277 10"	Yes

Mobile Panels V12.0

HMI devices	SIMATIC ET 200 CPU
Mobile Panel 177 6" DP	Yes
Mobile Panel 177 6" PN	Yes
Mobile Panel 277 8"	Yes
Mobile Panel 277 8" IWLAN V2	Yes
Mobile Panel 277F 8" IWLAN V2	Yes
Mobile Panel 277F 8" IWLAN V2 (RFID tag)	Yes
Mobile Panel 277 10"	Yes

Mobile Panels V13.0.1

HMI devices	SIMATIC ET 200 CPU
KTP 700 Mobile	Yes
KTP 900 Mobile	Yes

Comfort Panels V11.0

HMI devices	SIMATIC ET 200 CPU
KP400 Comfort	Yes
KTP400 Comfort	Yes
KTP400 Comfort Portrait	Yes
KP700 Comfort	Yes
TP700 Comfort	Yes
TP700 Comfort Portrait	Yes
KP900 Comfort	Yes
TP900 Comfort	Yes
TP900 Comfort Portrait	Yes
KP1200 Comfort	Yes
TP1200 Comfort	Yes
TP1200 Comfort Portrait	Yes
KP1500 Comfort	Yes
TP1500 Comfort	Yes
TP1500 Comfort Portrait	Yes
TP1900 Comfort	Yes
TP1900 Comfort Portrait	Yes
TP2200 Comfort	Yes
TP2200 Comfort Portrait	Yes

Comfort Panels V12.0

HMI devices	SIMATIC ET 200 CPU
KP400 Comfort	Yes
KTP400 Comfort	Yes
KTP400 Comfort Portrait	Yes
KP700 Comfort	Yes
TP700 Comfort	Yes
TP700 Comfort Portrait	Yes
KP900 Comfort	Yes
TP900 Comfort	Yes
TP900 Comfort Portrait	Yes
KP1200 Comfort	Yes
TP1200 Comfort	Yes
TP1200 Comfort Portrait	Yes
KP1500 Comfort	Yes
TP1500 Comfort	Yes
TP1500 Comfort Portrait	Yes
TP1900 Comfort	Yes

HMI devices	SIMATIC ET 200 CPU
TP1900 Comfort Portrait	Yes
TP2200 Comfort	Yes
TP2200 Comfort Portrait	Yes

Comfort Panels V13.0

HMI devices	SIMATIC ET 200 CPU
KP400 Comfort	Yes
KTP400 Comfort	Yes
KTP400 Comfort Portrait	Yes
KP700 Comfort	Yes
TP700 Comfort	Yes
TP700 Comfort Portrait	Yes
KP900 Comfort	Yes
TP900 Comfort	Yes
TP900 Comfort Portrait	Yes
KP1200 Comfort	Yes
TP1200 Comfort	Yes
TP1200 Comfort Portrait	Yes
KP1500 Comfort	Yes
TP1500 Comfort	Yes
TP1500 Comfort Portrait	Yes
TP1900 Comfort	Yes
TP1900 Comfort Portrait	Yes
TP2200 Comfort	Yes
TP2200 Comfort Portrait	Yes

Comfort Panels V13.0.1

HMI devices	SIMATIC ET 200 CPU
KP400 Comfort	Yes
KTP400 Comfort	Yes
KTP400 Comfort Portrait	Yes
KP700 Comfort	Yes
TP700 Comfort	Yes
TP700 Comfort Portrait	Yes
KP900 Comfort	Yes
TP900 Comfort	Yes
TP900 Comfort Portrait	Yes
KP1200 Comfort	Yes
TP1200 Comfort	Yes

HMI devices	SIMATIC ET 200 CPU
TP1200 Comfort Portrait	Yes
KP1500 Comfort	Yes
TP1500 Comfort	Yes
TP1500 Comfort Portrait	Yes
TP1900 Comfort	Yes
TP1900 Comfort Portrait	Yes
TP2200 Comfort	Yes
TP2200 Comfort Portrait	Yes

Runtime V11.0

HMI devices	SIMATIC ET 200 CPU
WinCC RT Advanced	Yes
WinCC RT Professional	Yes

Runtime V12.0

HMI devices	SIMATIC ET 200 CPU
WinCC RT Advanced	Yes
WinCC RT Professional	Yes

Runtime V13.0

HMI devices	SIMATIC ET 200 CPU
WinCC RT Advanced	Yes
WinCC RT Professional	Yes

Runtime V13.0.1

HMI devices	SIMATIC ET 200 CPU
WinCC RT Advanced	Yes
WinCC RT Professional	Yes

1.10.4.2 Valid data types for SIMATIC ET 200 CPU (Basic Panels)

Valid data types for connections with SIMATIC ET 200 CPU

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Length
BOOL	1 bit
SINT	1 byte
INT	2 bytes
DINT	4 bytes
USINT	1 byte
UINT	2 bytes
UDINT	4 bytes
REAL	4 bytes
LREAL	8 bytes
TIME	4 bytes
DATE	2 bytes
TIME_OF_DAY, TOD	4 bytes
STRING	(2+n) bytes, n = 0 to 254
CHAR	1 byte
Array of CHAR	--
BYTE	1 byte
WORD	2 bytes
DWORD	4 bytes
Date_And_Time	8 bytes
DTL	8 bytes
LDT	8 bytes

1.10.5 Creating connections in the "Connections" editor (Basic Panels)

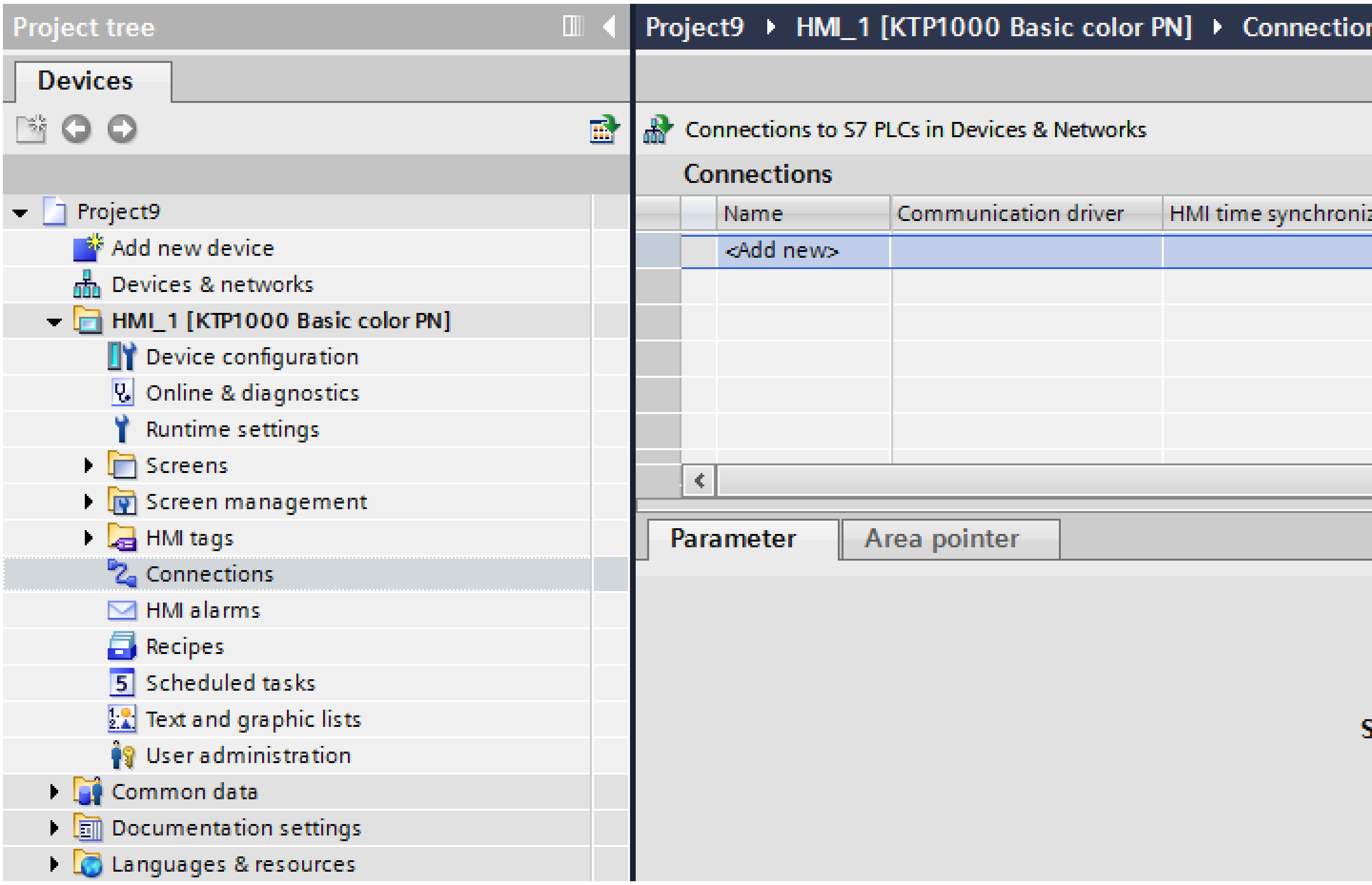
1.10.5.1 Creating a PROFINET connection (Basic Panels)

Requirements

- A project is open.
- An HMI device with a PROFINET interface has been created.

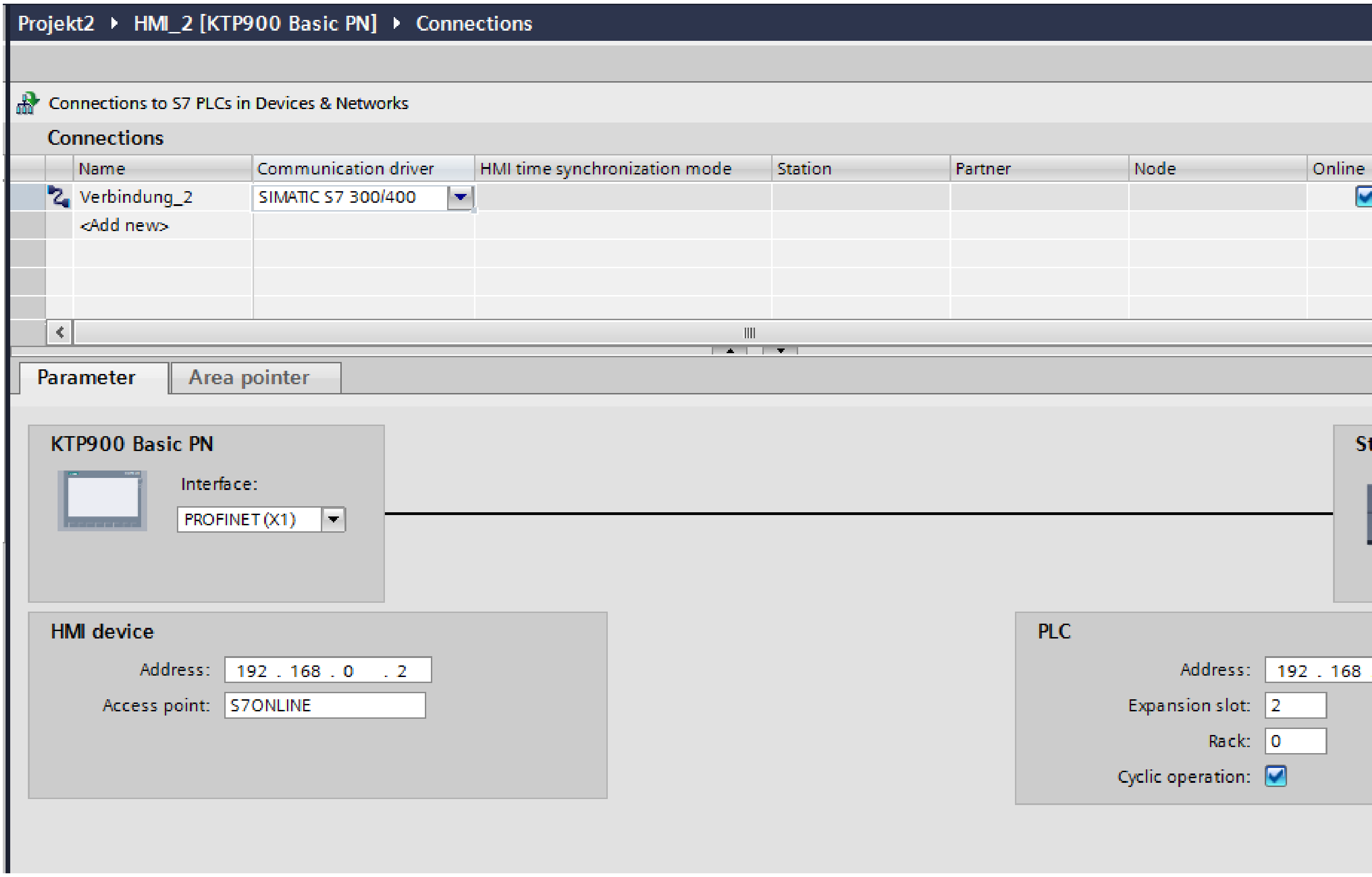
Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. In the "Communication drivers" column, select the "SIMATIC S7 300/400" driver.
4. Click the name of the connection.

5. Select a PROFINET interface of the HMI device in the Inspector window under "Parameters > Interface".



6. Set the IP addresses of the communication partners in the Inspector window:
   – HMI device: "Parameters > HMI device > Address"
   – PLC: "Parameters > PLC > Address"

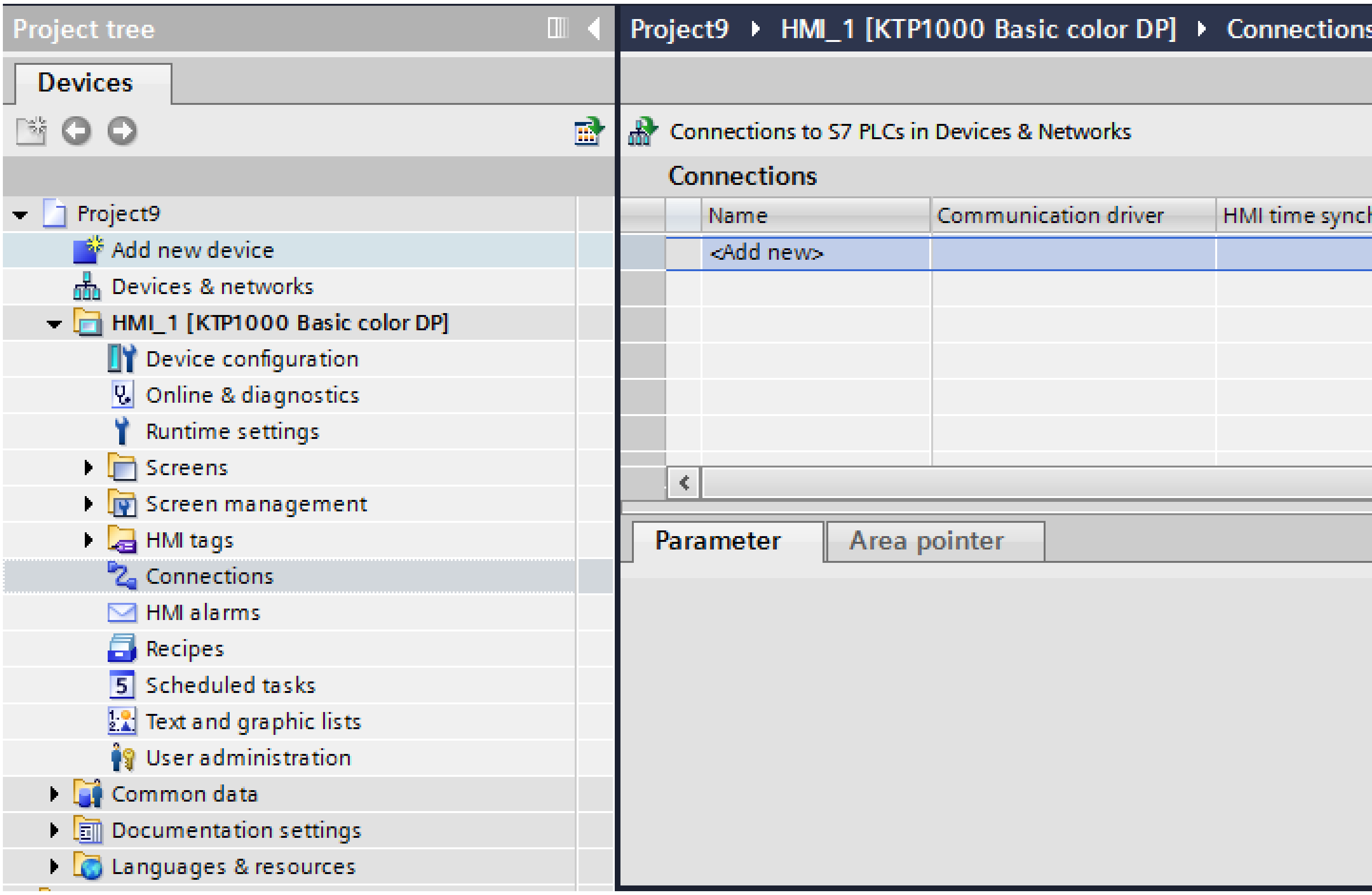
1.10.5.2 Creating a PROFIBUS DP connection (Basic Panels)

Requirements

- A project is open.
- An HMI device with a PROFIBUS interface has been created.

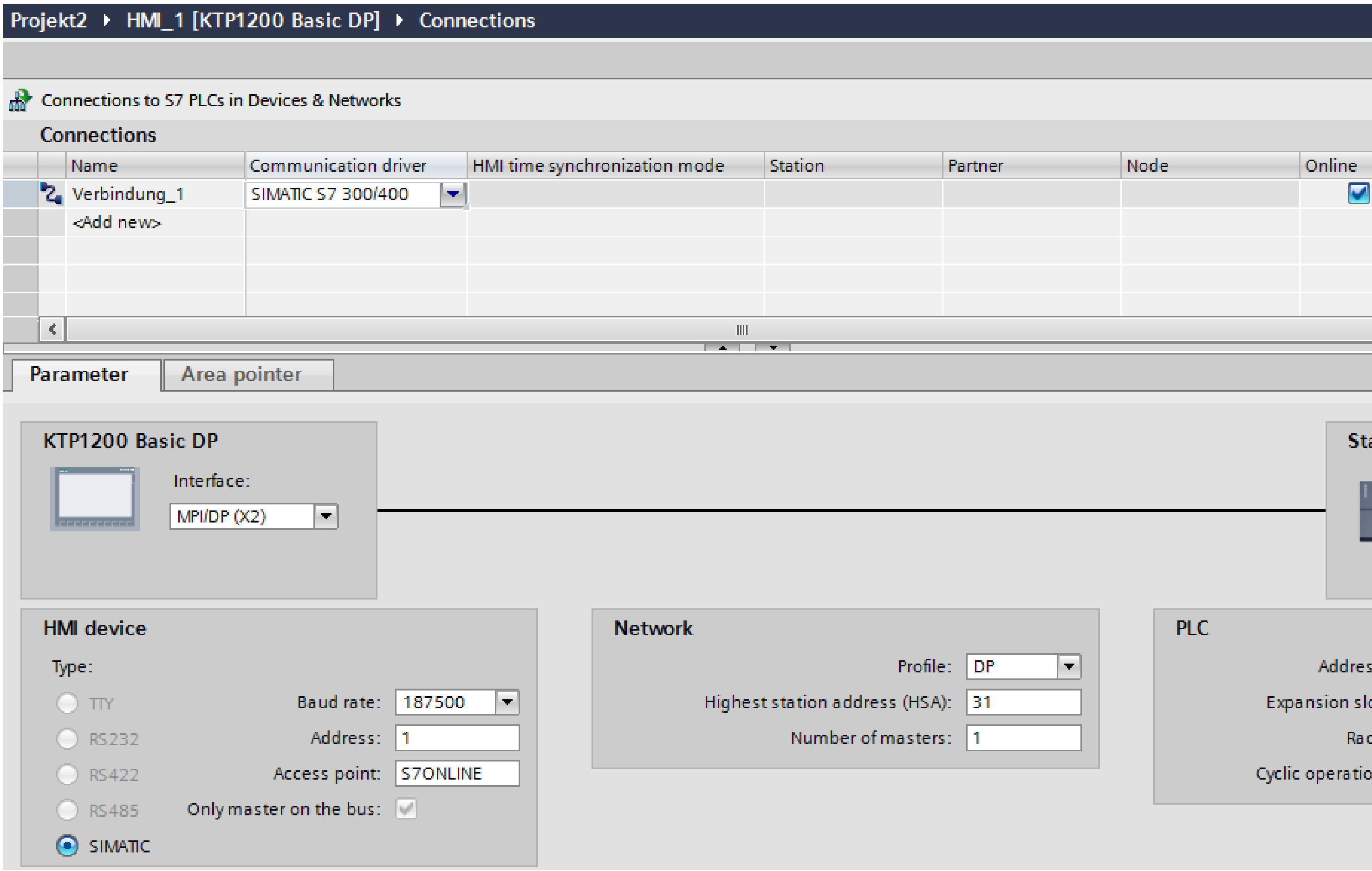
Procedure

1. Open the "Connections" editor of the HMI device.
2. Double-click "<Add>".



3. In the "Communication drivers" column, select the "SIMATIC S7 1200" driver.
4. Click the name of the connection.
5. Select the "MPI/DP" interface in the Inspector window under "Parameters".

6. Select the "DP" profile in the Inspector window under "Parameters > Network".



7. Set the addresses of the communication partners in the Inspector window:
   – HMI device: "Parameters > HMI device > Address"
   – PLC: "Parameters > PLC > Address"

1.10.5.3 Connection parameters (Basic Panels)

PROFIBUS parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the PROFIBUS network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  You set the PROFIBUS address of the HMI device under "Address". The PROFIBUS address must be unique in the PROFIBUS network.
- "Only master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master).
  In S7-200, you must set an HMI device as the master.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the network

Under "Network", you set the parameters for the PROFIBUS network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "DP", "Universal", or "Standard". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual PROFIBUS address. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud in the OP 73 or the OP 77A, the highest station address must be less than or equal 63.

- "Number of masters"
  For "Number of masters", set the number of masters in the PROFIBUS network. This information is necessary to correctly calculate the bus parameters.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the PROFIBUS address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Access password"
  Enter a password in the "Access password" field. This password must match the one you saved to the PLC.

  Note

  You only need a password if you have set "Complete protection" at the PLC.

  No connection is set up to the PLC if the "Complete protection" security level is stored on the PLC and you do not enter a password.

Ethernet parameters (Basic Panels)

Parameters for the HMI device

You set the parameters for the HMI device in the network under "HMI device".. The changes are not transferred automatically to the HMI device. You must change the settings in the Control Panel of the HMI device.

- "Interface"
  If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC.

  Note

  The IP address in the Control Panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

  The IP address already set up in the Control Panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

  The IP address is transferred to the HMI device during project transfer.
  To set up the IP address of the HMI device:
  – Click the HMI device.
  – Open the "Device configuration" editor.
  – Click the Ethernet interface.
  – Assign the IP address in the inspector window under:
    "General > PROFINET interface > Ethernet addresses"
- "Address"
  You assign the IP address of the HMI device in the "Address" area.
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.
- "Access point"
  Specifies the access point for the PG/PC interface that can be used to reach the communication partner.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  Under "Address", set the IP address of the S7 module to which the HMI device is connected.
- "Access password"
  Enter a password in the "Access password" field. This password must match the one you saved to the PLC.

Note

You only need a password if you have set "Complete protection" at the PLC.

No connection is set up to the PLC if the "Complete protection" security level is stored on the PLC and you do not enter a password.

1.11 Communicating with SIMATIC S7 200 (Basic Panels)

1.11.1 Communication with SIMATIC S7 200 (Basic Panels)

Introduction

This section describes the communication between an HMI device and the SIMATIC S7 200 PLC.

You can configure the following communication channels for the SIMATIC S7 200 PLC:

- PROFINET and Ethernet
- PROFIBUS
- MPI
- PPI

HMI connection for communication

You configure connections between the HMI device and a SIMATIC S7 200 in the "Connections" editor of the HMI device.

1.11.2 Creating a connection to SIMATIC S7 200 (Basic Panels)

Introduction

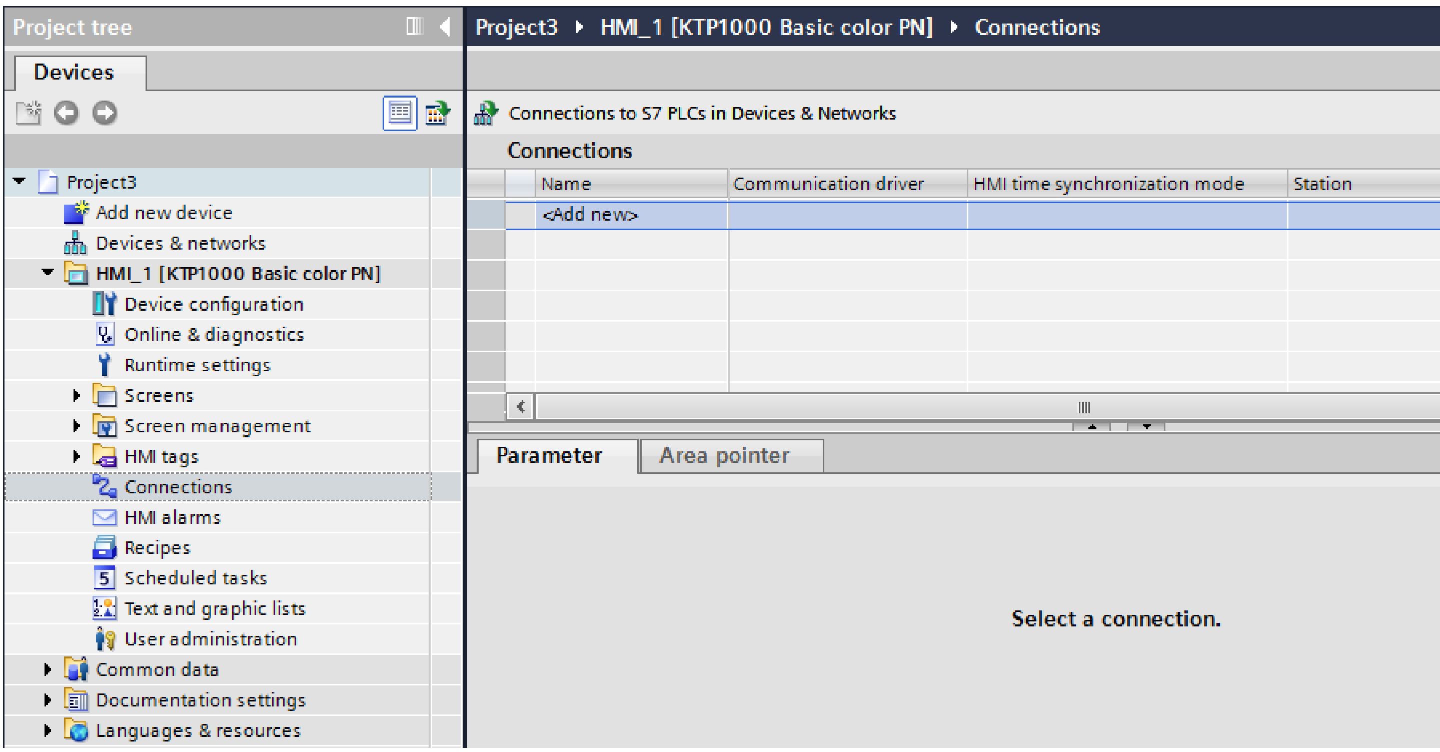
You configure a connection to the SIMATIC S7 200 PLC in the "Connections" editor of the HMI device. The interfaces are named differently depending on the HMI device.

Requirements

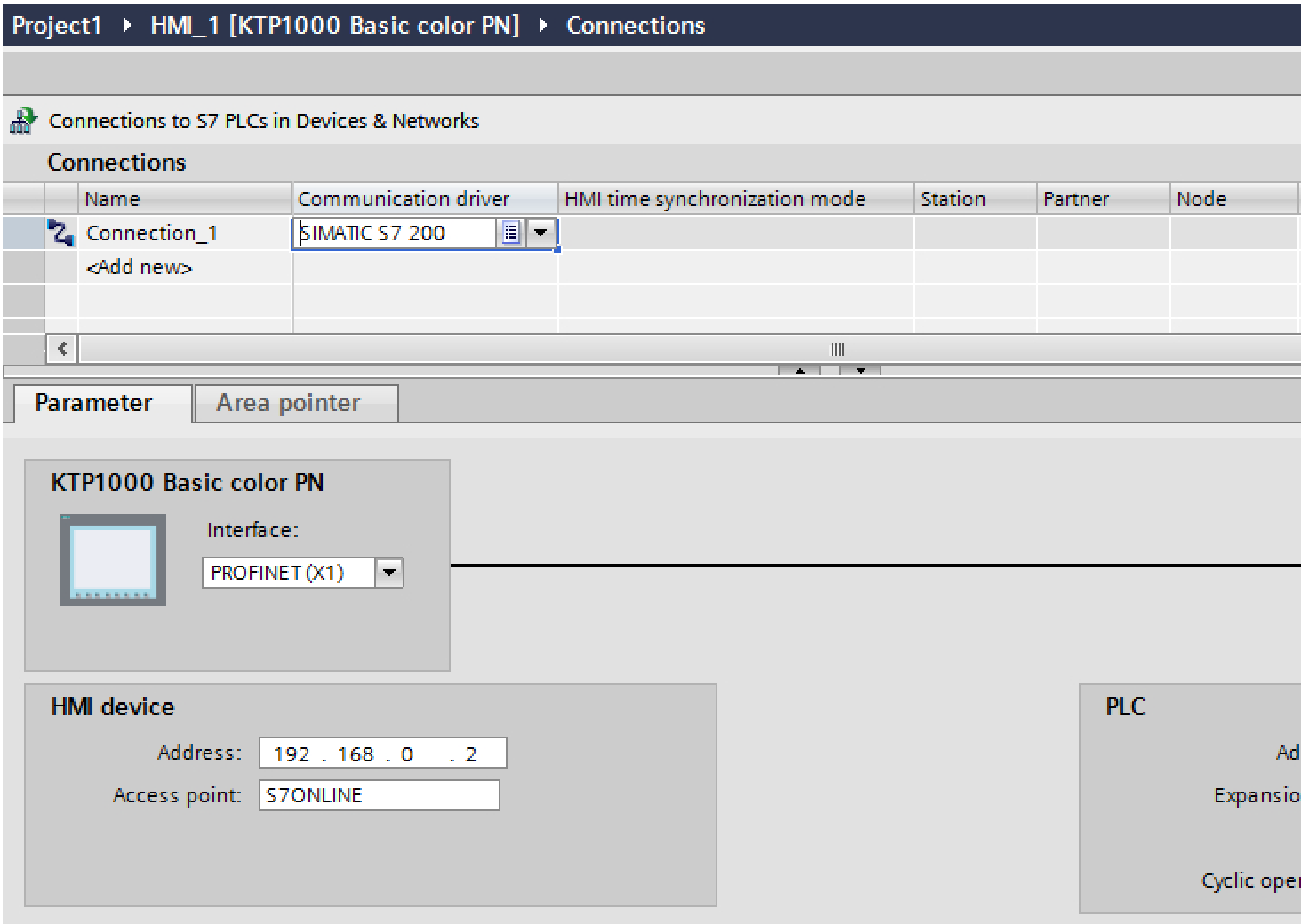
- A project is open.
- An HMI device has been created.

Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.
3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "SIMATIC S7 200" driver.
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".



See the chapter "Parameters for the connection (Page 292)" for additional details.

1.11.3 Parameters for the connection (Basic Panels)

1.11.3.1 Cyclic operation (Basic Panels)

Handling the "Cyclic operation" selection

If "Cyclic operation" is enabled, the HMI device sends a message frame to the CPU at the beginning of communication indicating that certain tags are required on a recurring basis.

The CPU then always transmits the data at the same cyclic interval. This saves the HMI device from having to output new requests for the data.

If cyclic operation is disabled, the HMI device sends a request whenever information is required.

Additional properties:

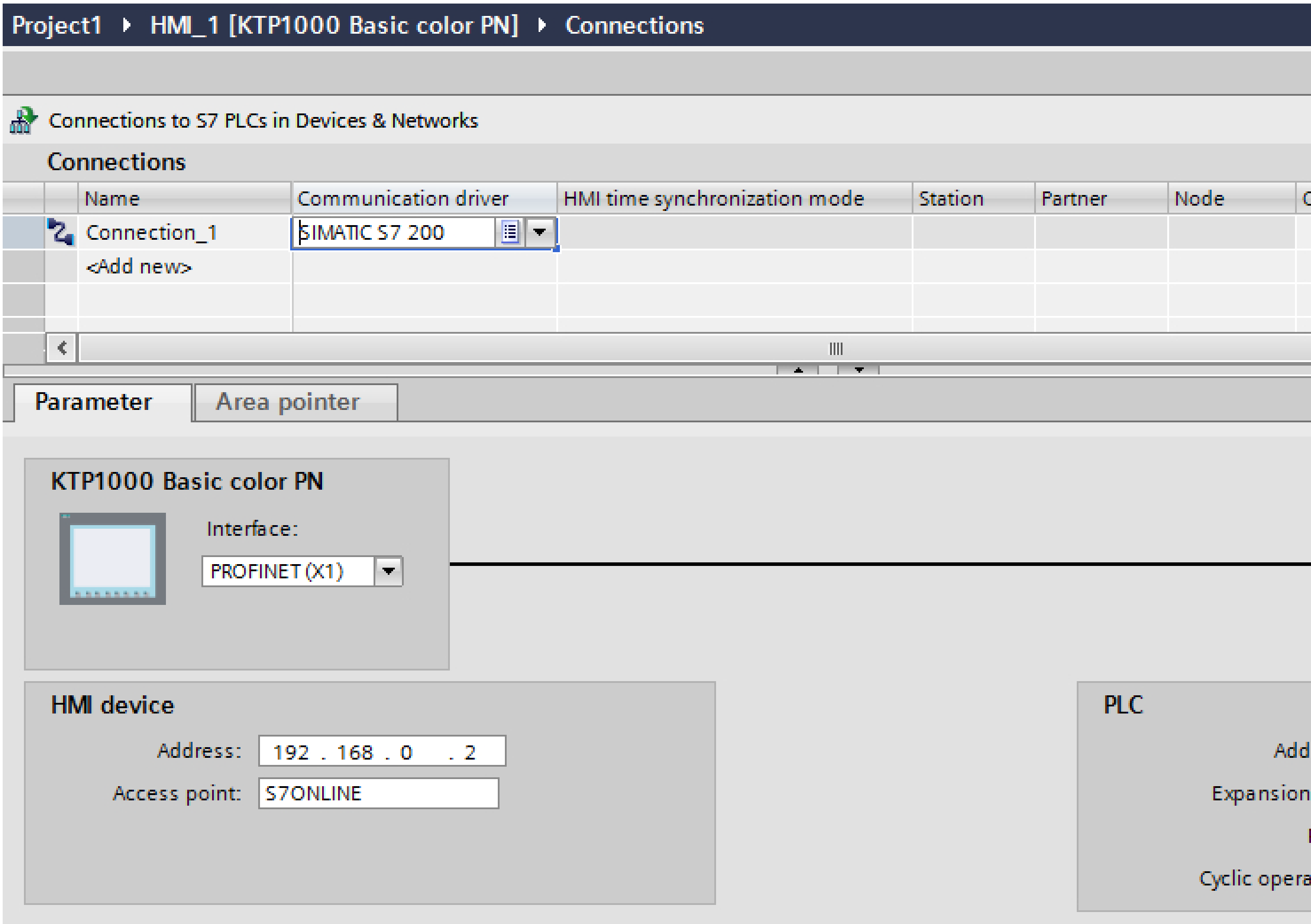
- Cyclic operation reduces data transmission load at the HMI device. The PLC resources are used to relieve load on the HMI device.
- The PLC only supports a certain number of cyclic services. The HMI device handles the operation if the PLC cannot provide any further resources for cyclic services.
- The HMI device generates the cycle if the PLC does not support the cyclic mode.
- Screen tags are not integrated in cyclic operation.
- Cyclic mode is only set up at the restart of Runtime.
- The HMI device transfers several jobs to the PLC if cyclic mode is enabled, depending on the PLC.
- The HMI device only transfers one job to the PLC if cyclic mode is disabled.

1.11.3.2 Parameters for the connection (SIMATIC S7 200) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device" and "PLC" areas are available for assigning parameters according to the interface used.


Project1 ▸ HMI_1 [KTP1000 Basic color PN] ▸ Connections
Connections to S7 PLCs in Devices & Networks
Connections
Name
Communication driver
HMI time synchronization mode
Station
Partner
Node
Connection_1
SIMATIC S7 200
<Add new>
Parameter
Area pointer
KTP1000 Basic color PN
Interface:
PROFINET (X1)
HMI device
Address:
192 . 168 . 0 . 2
Access point:
S7ONLINE
PLC
Expansion
Cyclic opera

1.11.3.3 Ethernet parameters (Basic Panels)

Parameters for the HMI device

You set the parameters for the HMI device in the network under "HMI device".. The changes are not transferred automatically to the HMI device. You must change the settings in the Control Panel of the HMI device.

- "Interface"
  If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC.

  Note

  The IP address in the Control Panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

  The IP address already set up in the Control Panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

  The IP address is transferred to the HMI device during project transfer.
  To set up the IP address of the HMI device:
  – Click the HMI device.
  – Open the "Device configuration" editor.
  – Click the Ethernet interface.
  – Assign the IP address in the inspector window under:
    "General > PROFINET interface > Ethernet addresses"
- "Address"
  You assign the IP address of the HMI device in the "Address" area.
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  Under "Address", set the IP address of the S7 module to which the HMI device is connected.
- "Expansion slot"
  Defines the number of the expansion slot of the CPU to be addressed.

- "Rack"
  Defines the rack number of the CPU to be addressed.
- "Cyclic operation"

Note

The setting "Cyclic operation" cannot be configured for the SIMATIC S7 1200 PLC.

When cyclic operation is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This increases system performance.
Disable cyclic mode if you are operating several HMI devices in parallel.

1.11.3.4 PROFIBUS parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the PROFIBUS network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

Note

If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  You set the PROFIBUS address of the HMI device under "Address". The PROFIBUS address must be unique in the PROFIBUS network.
- "Sole master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master).
  In S7 200, you must set an HMI device as the master.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the network

Under "Network", you set the parameters for the PROFIBUS network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "DP", "Universal", or "Standard". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual PROFIBUS address. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud in the OP 73 or the OP 77A, the highest station address must be less than or equal 63.

- "Number of masters"
  For "Number of masters", set the number of masters in the PROFIBUS network. This information is necessary to correctly calculate the bus parameters.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the PROFIBUS address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Cyclic operation"

  Note

  The setting "Cyclic operation" cannot be configured for the SIMATIC S7 1200 PLC.

  When cyclic operation is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This increases system performance. Disable cyclic mode if you are operating several HMI devices in parallel. This setting is not required for SIMATIC S7-200 PLCs.

1.11.3.5 MPI parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the MPI network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  For "Address", you set the MPI address of the HMI device. The MPI address must be unique throughout the MPI network.
- "Sole master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master). If you have connected only slaves to the HMI device, you must therefore disable the "Sole master on bus" safety feature.
  In S7 200, you must set an HMI device as the master.

Parameters for the network

Under "Network", you set the parameters for the MPI network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "MPI". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual MPI address. The setting must be identical throughout the network.
- "Number of masters"
  This setting is not required for MPI.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the MPI address of the S7 module (CPU, FM, or CP) to which the HMI device is connected.
- "Cyclic mode"
  When cyclic mode is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This improves system performance. Disable cyclic mode if you are operating several HMI devices in parallel. This setting is not required for SIMATIC S7-200.

1.11.3.6 PPI parameters (Basic Panels)

Parameters for the HMI device

You assign the parameters for the HMI device in the network once under "HMI device". The change applies to each communication partner.

- "Type"
  Specifies the physical connection used.
- "Interface"
  For "Interface", you select the HMI device interface via which the HMI device is connected to the PP network.
- "Baud rate"
  For "Baud rate", you set the transmission speed of the data in the network. The baud rate is determined by the slowest HMI device connected to the network. The setting must be identical throughout the network.

  Note

  If you set a baud rate of 1.5 Mbaud for OP 73 or OP 77A, the highest station address must be less than or equal to 63.

- "Address"
  For "Address", you set the PPI address of the HMI device. The PPI address must be unique throughout the PPI network.
- "Access point"
  For "Access point", you set the access point via which the communication partner is reached.
- "Sole master on bus"
  Disables an additional safety feature against bus faults when the HMI device is connected to the network. A passive station (slave) can only send data if it is requested to do so by an active station (master). If you have connected only slaves to the HMI device, you must therefore disable the "Sole master on bus" safety feature.
  In S7 200, you must set an HMI device as the master.

Parameters for the network

Under "Network", you set the parameters for the network to which the HMI device is linked.

- "Profile"
  For "Profile", you select the network profile that is used in the network. In "Profile", set "PPI". The setting must be identical throughout the network.
- "Highest address"
  For "Highest station address", set the highest station address. The highest station address must be greater than or equal to the highest actual MPI address. The setting must be identical throughout the network.
- "Number of masters"
  Set the number of the master on the network to "1".

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  For "Address", set the PPI address of the S7 module to which the HMI device is connected.
- "Cyclic operation"
  This parameter is not required for communication via PPI.

1.11.4 Data exchange (Basic Panels)

1.11.4.1 Data exchange using area pointers (Basic Panels)

General information on area pointers (Basic Panels)

Introduction

You use an area pointer to access a data area in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations.

The PLC and the HMI device trigger defined interactions based on the evaluation of stored data.

Configuration of area pointers

Before you use an area pointer, you enable it under "Connections > Area pointer". You then assign the area pointer parameters.

You can find more detailed information on configuring area pointers in:

Configuring area pointers (Page 58)

"Screen number" area pointer (Basic Panels)

Function

The HMI device saves information about the screen called on the HMI device to the "Screen number" area pointer.

This allows the transfer of the current screen contents from the HMI device to the PLC. The PLC can trigger specific reactions such as the call of a different screen.

Application

Configure and enable the area pointer in "Communication > Connections" before you put it into use. You can create only **one** instance of the "Screen number" area pointer and only on **one** PLC.

The screen number is always transferred to the PLC when a new screen is activated or when the focus within a screen changes from one screen object to another.

Structure

The area pointer is a data area in the memory of the PLC with a fixed length of 5 words.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1st word	Current screen type															
2nd word	Current screen number															
3rd word	Reserved															
4th word	Current field number															
5th word	Reserved															

- Current screen type
  "1" for root screen or
  "4" for permanent window
- Current screen number
  1 to 32767
- Current field number
  1 to 32767

"Date/time" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the HMI device to the PLC.

The PLC writes control job "41" to the job mailbox.

When it evaluates the control job, the HMI device writes its current date and the time in the data area configured in the "Date/time" area pointer. All definitions are coded in BCD format.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

Note

You cannot use the "Date/Time PLC" area pointer if you have configured the "Date/Time" area pointer.

Note

Symbolic addressing is not possible if you are using the "Date/Time" area pointer.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

The date/time data area has the following structure:

Data word	Most significant byte	Least significant byte	
	7 ... 0	7 ... 0	
n+0	Reserved	Hour (0 to 23)	Time
n+1	Minute (0 to 59)	Second (0 to 59)	
n+2	Reserved	Reserved	
n+3	Reserved	Weekday (1 to 7, 1=Sunday)	Date
n+4	Day (1 to 31)	Month (1 to 12)	
n+5	Year (80 to 99/0 to 29)	Reserved	

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

"Date/time PLC" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the PLC to the HMI device. Use this area pointer if the PLC is the time master.

The PLC loads the data area of the area pointer. All definitions are coded in BCD format.

The HMI device reads the data cyclically within the configured acquisition cycle and synchronizes itself.

Note

Set an acquisition cycle of sufficient length for the Date/time area pointer in order to avoid any negative impact on HMI device performance.
Recommended: Acquisition cycle of 1 minute, if the process allows this.

"Date/Time PLC" is a global area pointer and may be configured only once per project.

Note

You cannot use the "Date/Time" area pointer if you have configured the "Date/Time PLC" area pointer.

The date/time data area has the following structure:

DATE_AND_TIME format (in BCD code)

<table>
<tr><th rowspan="2">Data word</th><th colspan="3">Most significant byte</th><th colspan="3">Least significant byte</th></tr>
<tr><th>7</th><th>. . . . . .</th><th>0</th><th>7</th><th>. . . . . .</th><th>0</th></tr>
<tr><td>n+0</td><td colspan="3">Year (80 to 99/0 to 29)</td><td colspan="3">Month (1 to 12)</td></tr>
<tr><td>n+1</td><td colspan="3">Day (1 to 31)</td><td colspan="3">Hour (0 to 23)</td></tr>
<tr><td>n+2</td><td colspan="3">Minute (0 to 59)</td><td colspan="3">Second (0 to 59)</td></tr>
<tr><td>n+3</td><td colspan="3">Reserved</td><td colspan="2">Reserved</td><td>Weekday
(1 to 7, 1=Sun-day)</td></tr>
<tr><td>n+4 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
<tr><td>n+5 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
</table>

1) The two data words must exist in the data area to ensure that the data format matches WinCC flexible and to avoid reading false information.

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

"Coordination" area pointer (Basic Panels)

Function

The "Coordination" area pointer is used to implement the following functions:

- Detecting the startup of the HMI device in the control program
- Detecting the current operating mode of the HMI device in the control program
- Detecting whether the HMI device is ready to communicate in the control program

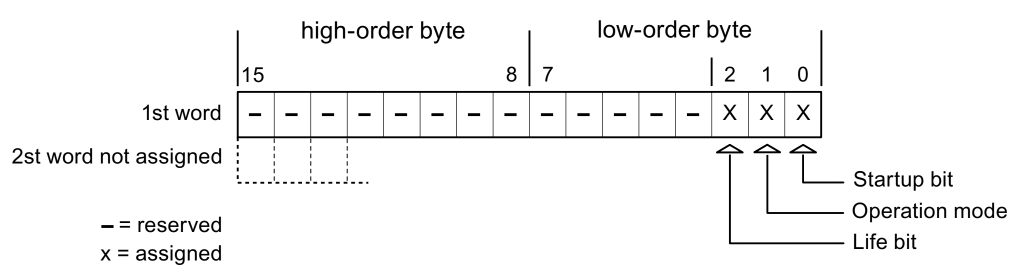
By default, the "Coordination" area pointer has the length of one word and cannot be changed.

Usage

Note

The HMI device always writes the entire coordination area when updating the area pointer. The control program can for this reason not make changes to the coordination area.

Assignment of the bits in the "Coordination" area pointer



Startup bit

The startup bit is set briefly to "0" by the HMI device during startup. It sets the bit permanently to "1" when startup is completed.

Operating mode

The operating mode bit is set to 1 as soon as the user switches the HMI device offline. The status of the operating mode bit is "0" during normal operation of the HMI device. You can determine the current operating mode of the HMI device by reading this bit in the control program.

Life bit

The HMI device inverts the life bit at intervals of approximately one second. You can check whether or not there is still a connection to the HMI device by querying this bit in the control program.

"Project ID" area pointer (Basic Panels)

Function

When Runtime starts, a check can be carried out as to whether the HMI device is connected to the correct PLC. This check is important when operating with several HMI devices.

For this, the HMI device compares a value stored on the PLC with the value specified in configuration. This ensures compatibility of the configuration data with the PLC program. If there is no concordance, a system event is given on the HMI device and Runtime is stopped.

Use

Note

HMI connections cannot be switched "online".

The HMI connection in which the "Project ID" area pointer is used must be switched "online".

To use this area pointer, set up the following during the configuration:

- Define the version of the configuration. Values between 1 and 255 are possible.
  You enter the version in the editor "Runtime settings > General" in the "Identification" area.
- Data address of the value for the version that is stored in the PLC:
  You enter the data address in the editor "Communication > Connections" under "Address".

Connection failure

A connection failure to a device on which the "Project ID" area pointer is configured results in all the other connections of the device being switched to "offline".

This behavior has the following requirements:

- You have configured several connections in a project.
- You are using the "Project ID" area pointer in at least one connection.

Causes which may set connections "offline":

- The PLC is not available.
- The connection has been switched offline in the Engineering System.

"Job mailbox" area pointer (Basic Panels)

Function

The PLC can use the job mailbox to transfer jobs to the HMI device to trigger corresponding actions on the HMI device. These functions include, for example:

- Display screen
- Set date and time

Data structure

The first word of the job mailbox contains the job number. Depending on the job mailbox, up to three parameters can be transferred.

Word	Most significant byte	Least significant byte
n+0	0	Job number
n+1	Parameter 1	
n+2	Parameter 2	
n+3	Parameter 3	

The HMI device evaluates the job mailbox if the first word of this job is not equal to zero. This means that the parameters must be entered in the job mailbox first, followed by the job number.

When the HMI device accepts the job mailbox, the first word is set to 0 again. The execution of the job mailbox is generally not completed at this point in time.

Job mailboxes

All job mailboxes and their parameters are listed below. The "No." column contains the job number of the job mailbox. Job mailboxes can only be triggered by the PLC when the HMI device is online.

No.	Function	
14	**Set time (BCD-coded)**	
	Parameter 1	Left byte: - Right byte: hours (0-23)
	Parameter 2	Left byte: minutes (0-59) Right byte: seconds (0-59)
	Parameter 3	-
15	**Setting the date (BCD coded)** 3)	
	Parameter 1	Left byte: - Right byte: Weekday (1-7: Sunday-Saturday)
	Parameter 2	Left byte: day (1-31) Right byte: month (1-12)
	Parameter 3	Left byte: year
23	**User logon**	

No.	Function	
14	**Set time (BCD-coded)**	
	Logs the user on with the name "PLC user" at the HMI device with the group number transferred in Parameter 1. The logon is possible only when the transferred group number exists in the project.	
	Parameter 1	Group number 1 to 255
	Parameter 2, 3	-
24	**User logoff**	
	Logs off the current user. (The function corresponds to the "logoff" system function)	
	Parameter 1, 2, 3	-
40	**Transfer date/time to PLC**	
	(In the S7 format DATE_AND_TIME) An interval of at least 5 seconds must be maintained between two successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
41	**Transfer date/time to PLC**	
	(In OP/MP format) An interval of at least 5 seconds must be maintained between successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
46	**Update tag**	
	Causes the HMI device to read the current value of the tags from the PLC whose update ID matches the value transferred in Parameter 1. (Function corresponds to the "UpdateTag" system function.)	
	Parameter 1	1 - 100
49	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Warnings" class from the alarm buffer.	
	Parameter 1, 2, 3	-
50	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Errors" class from the alarm buffer.	
	Parameter 1, 2, 3	-
51	**Screen selection** [2)]	
	Parameter 1	Screen number
	Parameter 2	-
	Parameter 3	Field number
69	**Reading data record from PLC** [1)]	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	0: Do not overwrite existing data record 1: Overwrite existing data record
70	**Writing data record to PLC** [1)]	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	-

1)	Only devices supporting recipes
2)	OP 73, OP 77A and TP 177A HMI devices also execute the "Screen selection" job mailbox if the on-screen keyboard is active.
3)	The weekday is ignored on HMI device KTP 600 BASIC PN.

"Data record" area pointer (Basic Panels)

"Data record" area pointer (Basic Panels)

Function

When data records are transferred between the HMI device and PLC, both partners access common communications areas on the PLC.

Data transfer types

There are two ways of transferring data records between the HMI device and PLC:

- Transfer without synchronization
- Transfer with synchronization over the data record

Data records are always transferred directly. That is, the tag values are read from an address or written to an address configured for this tag directly, without redirecting the values by means of interim memory.

Initiating the transfer of data records

There are three ways of triggering the transfer:

- Operator input in the recipe view
- Job mailboxes
  The transfer of data records can also be triggered by the PLC.
- Triggering by configured functions

If the transfer of data records is triggered by a configured function or by a job mailbox, the recipe view on the HMI device remains operable. The data records are transferred in the background.

Simultaneous processing of several transfer requests is, however, not possible. In this case, the HMI device rejects the other transfer requests with a system event.

Sequence of a transfer started by the operator in the recipe view (Basic Panels)

Reading from the PLC started by the operator in the recipe view

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe number to be read and the status "Transferring" in the data record and sets the data record number to 0.	Abort with system event.
3	The HMI device reads the values from the PLC and displays them in the recipe view. If the recipes have synchronized tags, the values from the PLC are also written to the tags.	
4	The HMI device sets the status "Transfer completed."	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC started by the operator in the recipe view

Step	Action	
	Check: Status word = 0?	
1	Yes	No
	The HMI device enters the recipe and data record number to be written and the status "Transferring" in the data mailbox.	Abort with system event.
2	The HMI device writes the current values to the PLC. If the recipes have synchronized tags, the changed values are synchronized in the recipe view and tags and then written to the PLC.	
3	The HMI device sets the status "Transfer completed."	
4	If required, the control program can now evaluate the transferred data.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Sequence of the transfer triggered by a job mailbox (Basic Panels)

The transfer of data records between the HMI device and the PLC can be initiated by the HMI device or by the PLC.

The two job mailboxes No. 69 and No. 70 are available for this type of transfer.

No. 69: Read data record from PLC ("PLC → DAT")

Job mailbox no. 69 transfers data records from the PLC to the HMI device. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	69
Word 2	Recipe number (1-999)	
Word 3	Data record number (1-65,535)	
Word 4	Do not overwrite existing data record: 0 Overwrite existing data record: 1	

No. 70: Write data record to PLC ("DAT → PLC")

Job mailbox No. 70 transfers data records from the HMI device to the PLC. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	70
Word 2	Recipe number (1-999)	
Word 3	Data record number (1-65,535)	
Word 4	—	

Sequence when reading from the PLC with job mailbox "PLC → DAT" (no. 69)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data mailbox.	Abort without return message.
3	The HMI device reads the values from the PLC and saves these to the data record defined in the job mailbox.	
4	• If "Overwrite" was selected in the job, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "Do not overwrite" was selected in the job, and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Sequence of writing to the PLC using job mailbox "DAT → PLC" (no. 70)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data mailbox.	Abort without return message.
3	The HMI device fetches the values of the data record specified in the job from the data medium and writes the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Sequence of the transfer when triggered by a configured function (Basic Panels)

Reading from the PLC using a configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system event.
3	The HMI device reads the values from the PLC and stores them in the data record specified in the function.	
4	• If "Yes" was selected for the "Overwrite" function, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "No" was selected for the "Overwrite" function and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data mailbox.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC by means of configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data mailbox.	Abort with system event.
3	The HMI device fetches the values of the data record specified in the function from the data medium and transfers the values to the PLC.	

Step	Action	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Possible causes of error when transferring data records (Basic Panels)

Possible causes of faults

The section below shows possible causes of errors which lead to a data record transfer being terminated with errors:

- Tag address not set up on the PLC
- Overwriting data records not possible
- Recipe number does not exist
- Data record number does not exist

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Reaction to an aborted transfer due to errors

If the transfer of data records is aborted due to errors, the HMI device reacts as follows:

- Triggering by the operator in the recipe view
  Information in the status bar of the recipe view and output of system events
- Triggered by function
  Output of system events
- Triggering by job mailbox
  No return message on the HMI device

You can nonetheless evaluate the status of the transfer by querying the status word in the data mailbox.

Transfer without synchronization (Basic Panels)

If you select asynchronous transfer of data records between the HMI device and PLC, there is no coordination over the common data areas. It is therefore unnecessary to set up a data area during configuration.

Asynchronous data record transfer can be a useful alternative, for example, when:

- The system is capable of excluding the risk of uncontrolled overwriting of data by the communication peer.
- The PLC does not require information about the recipe number and data record number.
- The transfer of data records is triggered by the operator of the HMI device.

Reading values

When a read job is triggered, the values are read from the PLC addresses and transferred to the HMI device.

- Triggering by the operator in the recipe view:
  The values are downloaded to the HMI device. You can then process, edit, or save these values, for example.
- Triggering by a function or job mailbox:
  The values are saved immediately to the data volume.

Writing values

When a write job is triggered, the values are written to the PLC addresses.

- Triggering by the operator in the recipe view:
  The current values are written to the PLC.
- Triggering by a function or job mailbox:
  The current values are written to the PLC from the data medium.

Transfer with synchronization (Basic Panels)

If you select synchronous transfer, both communication partners set status bits in the common data area. You can use this mechanism to prevent uncontrolled overwriting of data in either direction in your control program.

Application

Synchronous data record transfer can be a useful solution, for example, when:

- The PLC is the "active partner" in the transfer of data records.
- The PLC evaluates the information about the recipe number and data record number.
- The transfer of data records is triggered by means of a Job mailbox.

Requirements

In order to synchronize transfer of data records between the HMI device and the PLC, the following requirements must be met during configuration:

- An area pointer has been set up: "Communication > Connections" editor in "Area pointer".
- The PLC with which the HMI device synchronizes transfer of data records is specified in the recipe:
  "Recipes" editor in the Inspector window the "Coordinated transfer of data records" option under "General > Synchronization > Settings".

Structure of the data area

The data area has a fixed length of 5 words. Structure of the data area:

	15		0
1. Word	Current recipe number (1 - 999)		
2. Word	Current data record number (0 - 65535)		
3. Word	Reserved		
4. Word	Status (0, 2, 4, 12)		
5. Word	Reserved		

- Status
  The status word (word 4) can adopt the following values:

Value		Meaning
Decimal	Binary	
0	0000 0000	Transfer permitted, data record free
2	0000 0010	Transferring
4	0000 0100	Transfer completed without error
12	0000 1100	Transfer completed with error

1.11.4.2 Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

1.11.4.3 Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configure alarms

Several steps are needed to configure alarms, such as operational messages, error alarms, and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a SIMATIC communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
SIMATIC S7 PLCs	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

How the bit positions are counted

For connections with a SIMATIC communication driver, the following counting method applies:

How the bit positions are counted	Byte 0								Byte 1							
	Most significant byte								Least significant byte							
In SIMATIC S7 PLCs	7							0	7							0
In WinCC you configure:	15							8	7							0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

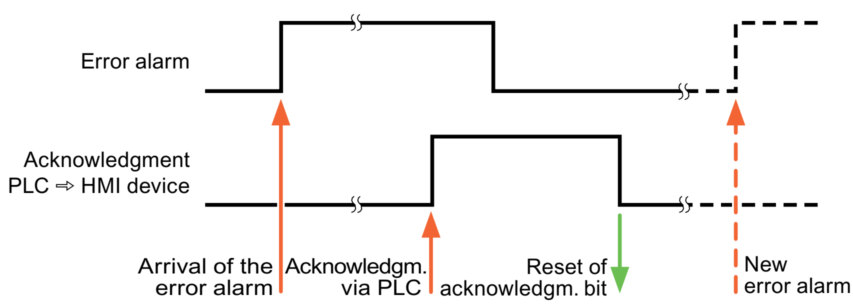
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment

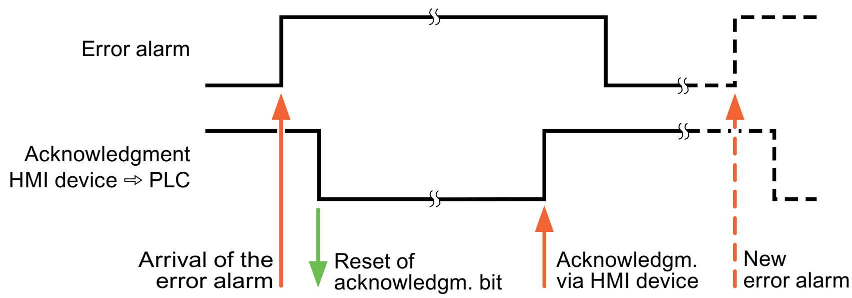
tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.11.5 Performance features of communication (Basic Panels)

1.11.5.1 Device dependency S7 200 (Basic Panels)

Communication with the SIMATIC S7-200 controller

If you use devices from an earlier version of the TIA Portal with TIA Portal V13, it may not be possible to configure integrated connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC S7-200
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes

HMI devices	SIMATIC S7-200
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V12.0

HMI devices	SIMATIC S7-200
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	Yes
KTP1000 Basic PN	Yes
TP1500 Basic PN	Yes

Basic Panels V13.0

HMI devices	SIMATIC S7-200
KTP400 Basic PN	Yes
KTP700 Basic PN	Yes
KTP700 Basic DP	Yes
KTP900 Basic PN	Yes
KTP1200 Basic PN	Yes
KTP1200 Basic DP	Yes

1.11.5.2 Permitted data types for SIMATIC S7 200 (Basic Panels)

Permitted data types for connections with SIMATIC S7 200

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Length
Bool	1 bit
Byte	1 byte
Char	1 byte
Word	2 bytes
Int	2 bytes

Data type	Length
DWord	4 bytes
DInt	4 bytes
Real	4 bytes
StringChar	--
Timer	2 bytes
Array	--

Note

Disconnection with a PPI network

If you are using arrays in the configuration, an array size of approximately 1000 bytes may cause an interruption of the connection.

Use smaller arrays in your configuration.

1.12 Communicating with SIMATIC LOGO! (Basic Panels)

1.12.1 Communication with SIMATIC LOGO! (Basic Panels)

Introduction

This section describes the communication between an HMI device and the SIMATIC LOGO! controller.

You can configure the following communication channels for the SIMATIC LOGO! controller:

- PROFINET
- Ethernet

HMI connection for communication

You configure connections between the HMI device and SIMATIC LOGO! in the "Connections" editor of the HMI device.

Data exchange

Data exchange with the SIMATIC LOGO! control system is possible by means of tags.

Data cannot be exchanged using area pointers.

1.12.2 Creating a connection to SIMATIC LOGO! (Basic Panels)

Introduction

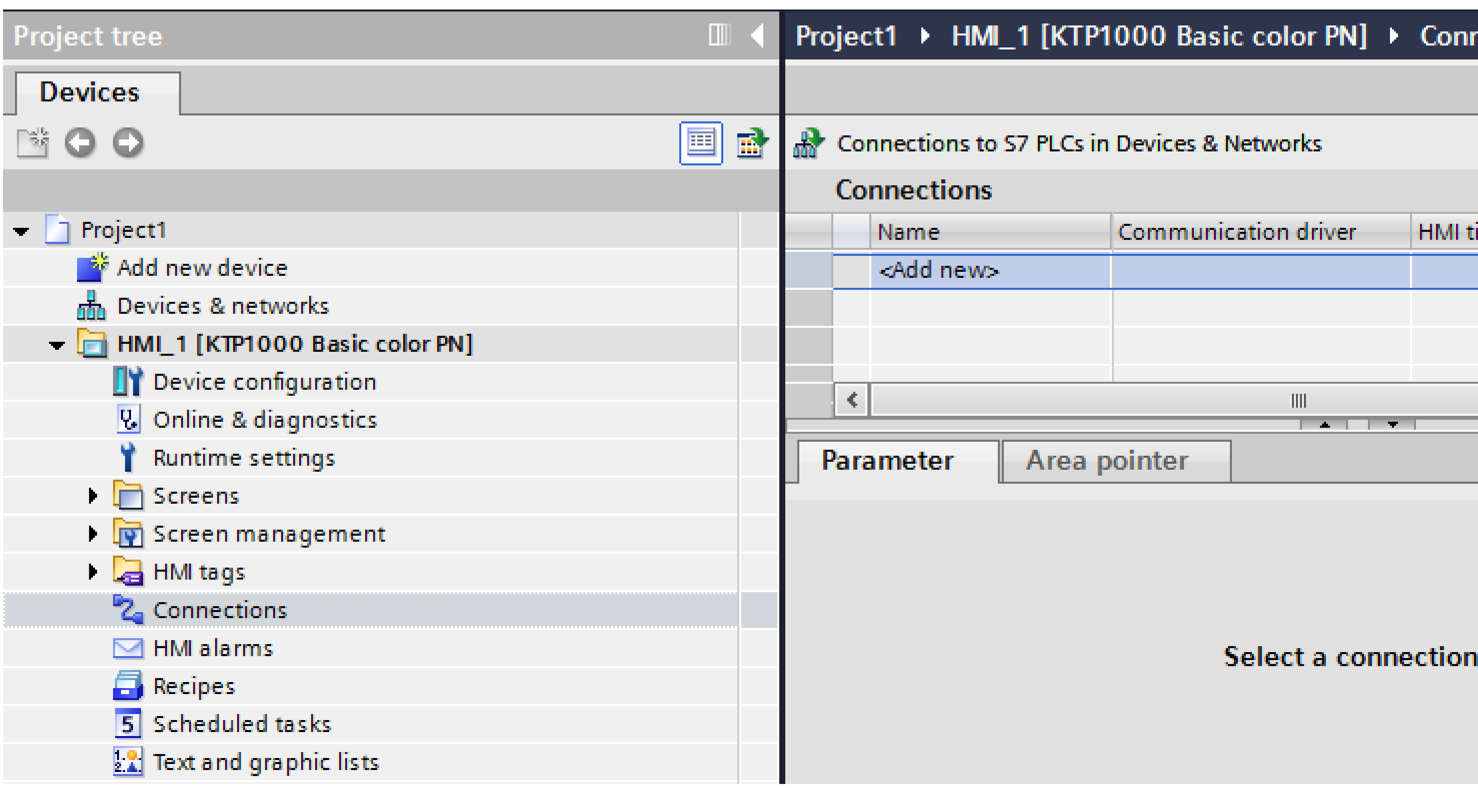
You configure a connection to the SIMATIC LOGO! controller in the "Connections" editor of the HMI device. The interfaces are named differently depending on the HMI device.

Requirements

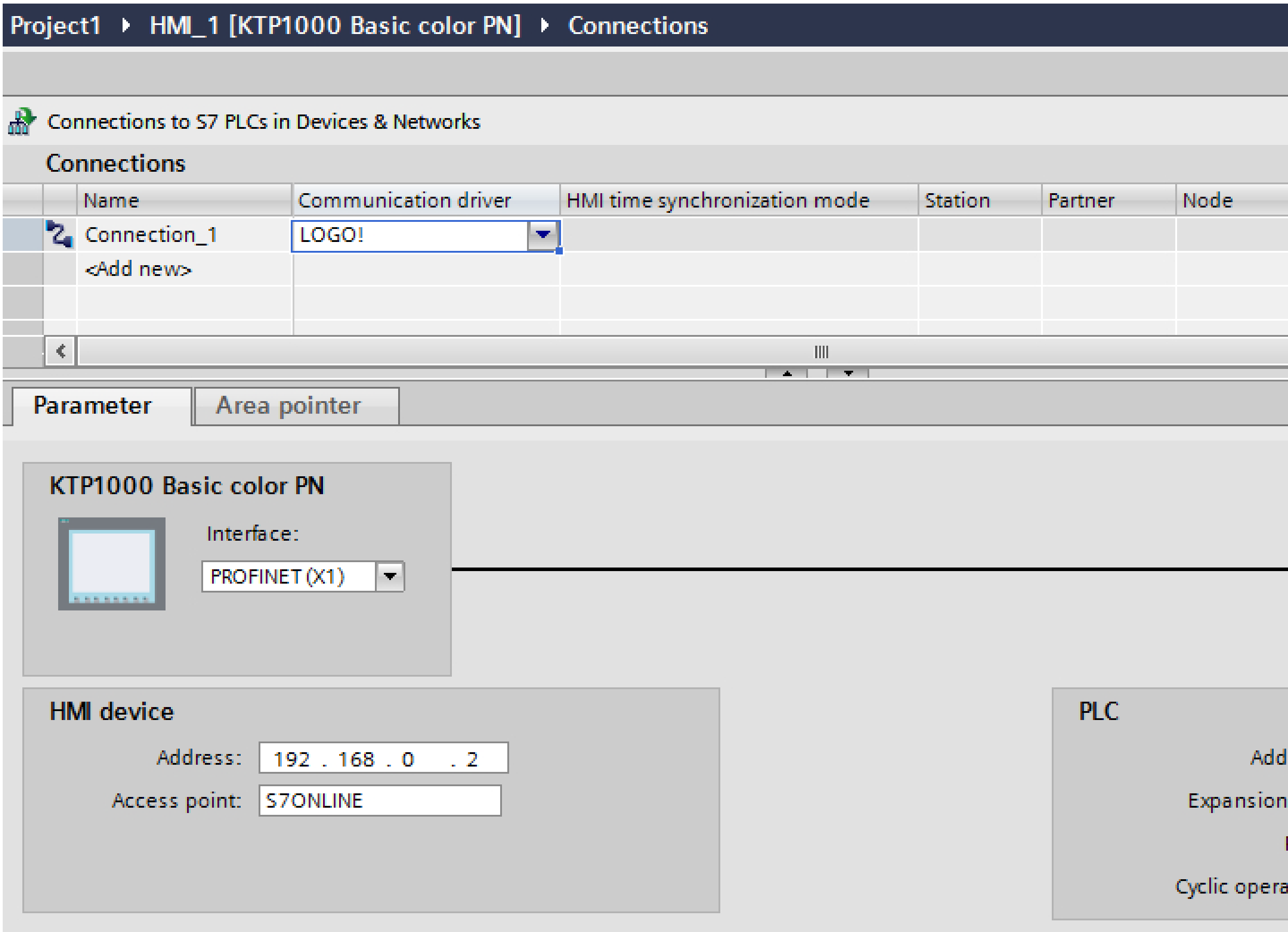
- A project is open.
- An HMI device has been created.

Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.
3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "SIMATIC LOGO!" driver.
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".



See the chapter "AUTOHOTSPOT" for additional details.

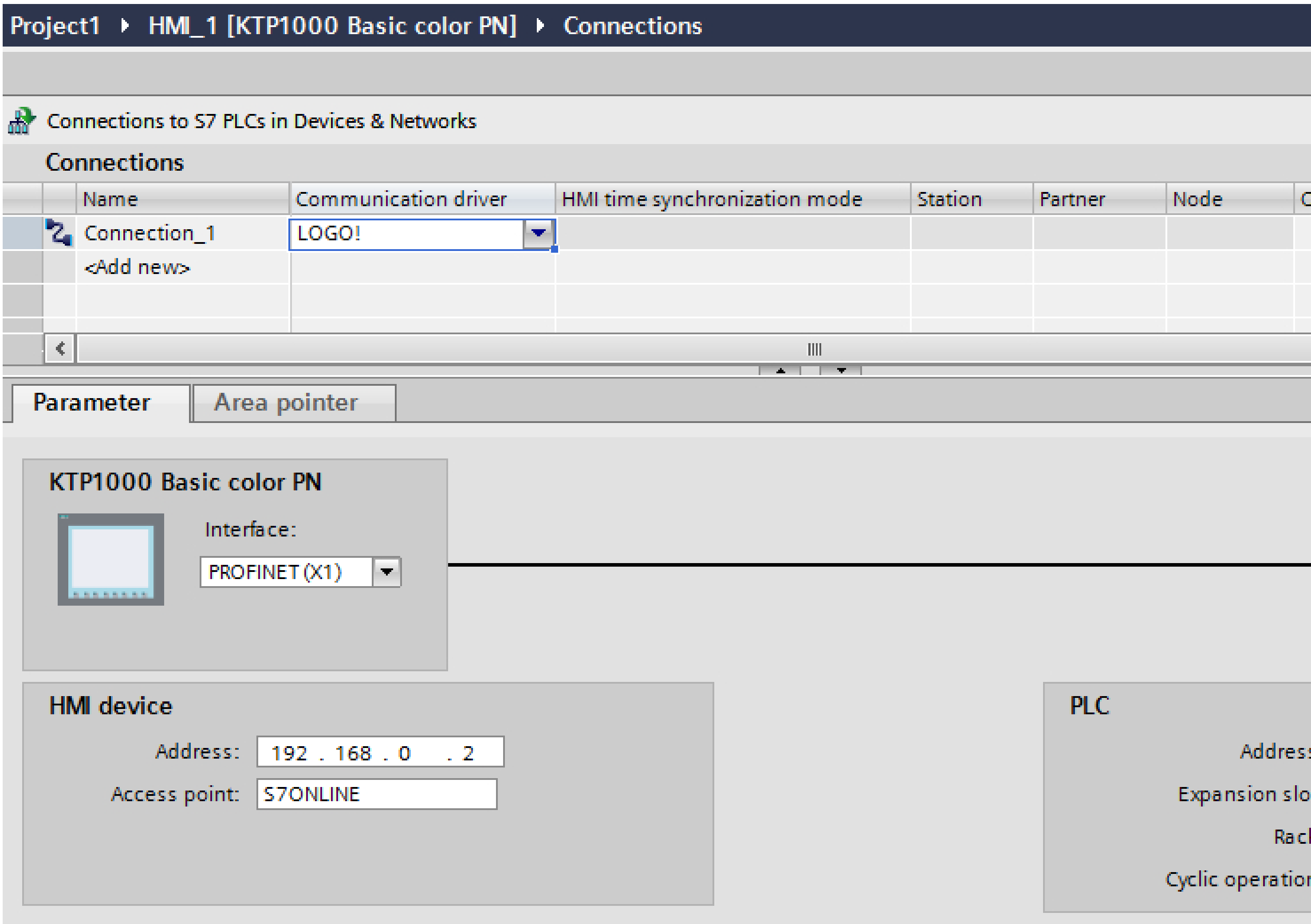
1.12.3 Connection parameters (Basic Panels)

1.12.3.1 Connection parameters (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device" and "PLC" areas are available for assigning parameters according to the interface used.



1.12.3.2 Ethernet parameters (Basic Panels)

Parameters for the HMI device

You set the parameters for the HMI device in the network under "HMI device".. The changes are not transferred automatically to the HMI device. You must change the settings in the Control Panel of the HMI device.

- "Interface"
  If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC.

  Note

  The IP address in the Control Panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

  The IP address already set up in the Control Panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

  The IP address is transferred to the HMI device during project transfer.
  To set up the IP address of the HMI device:
  - Click the HMI device.
  - Open the "Device configuration" editor.
  - Click the Ethernet interface.
  - Assign the IP address in the inspector window under:
    "General > PROFINET interface > Ethernet addresses"
- "Address"
  You assign the IP address of the HMI device in the "Address" area.
  When you transfer the WinCC project to the HMI device, this IP address is set up directly in the HMI device.
- "Access point"
  The access point defines a logical device name through which the communication partner can be reached.

Parameters for the PLC

Under "PLC", you address the S7 module with which the HMI device will exchange data. Assign a name for the connection for each communication partner.

- "Address"
  Under "Address", set the IP address of the S7 module to which the HMI device is connected.
- "Expansion slot"
  Defines the number of the expansion slot of the CPU to be addressed.

- "Rack"
  Defines the rack number of the CPU to be addressed.
- "Cyclic operation"

Note

The setting "Cyclic operation" cannot be configured for the SIMATIC S7 1200 PLC.

When cyclic operation is enabled, the PLC optimizes the data exchange between the HMI device and the PLC. This increases system performance.
Disable cyclic mode if you are operating several HMI devices in parallel.

1.12.3.3 Cyclic operation (Basic Panels)

Handling the "Cyclic operation" selection

If "Cyclic operation" is enabled, the HMI device sends a message frame to the CPU at the beginning of communication indicating that certain tags are required on a recurring basis.

The CPU then always transmits the data at the same cyclic interval. This saves the HMI device from having to output new requests for the data.

If cyclic operation is disabled, the HMI device sends a request whenever information is required.

Additional properties:

- Cyclic operation reduces data transmission load at the HMI device. The PLC resources are used to relieve load on the HMI device.
- The PLC only supports a certain number of cyclic services. The HMI device handles the operation if the PLC cannot provide any further resources for cyclic services.
- The HMI device generates the cycle if the PLC does not support the cyclic mode.
- Screen tags are not integrated in cyclic operation.
- Cyclic mode is only set up at the restart of Runtime.
- The HMI device transfers several jobs to the PLC if cyclic mode is enabled, depending on the PLC.
- The HMI device only transfers one job to the PLC if cyclic mode is disabled.

1.12.4 Data exchange (Basic Panels)

1.12.4.1 Trends (Basic Panels)

General information on trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out either time- or bit-triggered, depending on the configuration.

For additional information see:

AUTOHOTSPOT

Note

The value is read out time-triggered for Basic Panels.

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration. Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

Bit-triggered trends

Through a trigger bit set in the trend request tag, the HMI device either reads in a trend value or an entire trend buffer. This setting is defined in the configuration. Bit-triggered trends are normally used to represent fast changing values. One example might be the injection pressure in the production of plastic parts.

To trigger bit-triggered trends, appropriate external tags must be created in the "HMI tags" editor and connected to trend areas during configuration. The HMI device and PLC then communicate with each other via these trend areas.

The following areas are available for trends:

- Trend request area
- Trend transfer area 1
- Trend transfer area 2 (required only with switch buffers)

Trend request and trend transfer (Basic Panels)

Trend request area

The HMI device sets corresponding bits in the trend request area when you open a screen which contains one or more trends on the HMI device. After closing the screen, the HMI device resets the relevant bits in the trend request area.

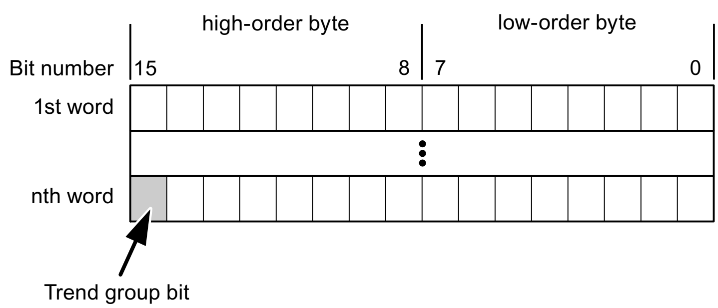
The trend request area can be used for evaluation purposes in the PLC to determine which trend is currently being displayed on the HMI device. Trends can also be triggered without evaluation of the trend request area.

Trend transfer area 1

This area is used to trigger trends. You must set the bit assigned to the trend in the trend transfer area and set the trend group bit in your control program. The trend group bit is the last bit in the trend transfer area.

The HMI device detects the trigger. The HMI device reads either a value or the entire buffer from the PLC. It then resets the trend bit and the trend group bit.

The following picture shows the structure of a trend transfer area.



The trend transfer area must not be modified by the PLC program until the trend group bit has been reset.

Trend transfer area 2

Trend transfer area 2 is required for trends configured with a switch buffer. The trend transfer areas 1 and 2 have a similar structure.

Switch buffer

The switch buffer is a second buffer for the same trend that can be set up during configuration.

The PLC writes to Buffer 2 while the HMI device reads values from Buffer 1, and writes to Buffer 1 when the HMI device is reading Buffer 2. This prevents the PLC from overwriting trend values while the trend is being read by the HMI device.

Permitted data types for trends (Basic Panels)

For SIMATIC S7

You assign one bit to each trend during configuration. Tags and array tags of the "Word" or "Int" data type are permitted.

1.12.4.2 Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configure alarms

Several steps are needed to configure alarms, such as operational messages, error alarms, and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a SIMATIC communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
SIMATIC S7 PLCs	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

How the bit positions are counted

For connections with a SIMATIC communication driver, the following counting method applies:

How the bit positions are counted	Byte 0								Byte 1							
	Most significant byte								Least significant byte							
In SIMATIC S7 PLCs	7							0	7							0
In WinCC you configure:	15							8	7							0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

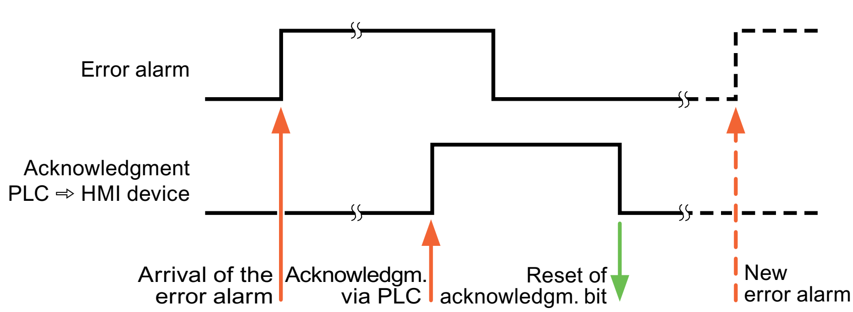
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

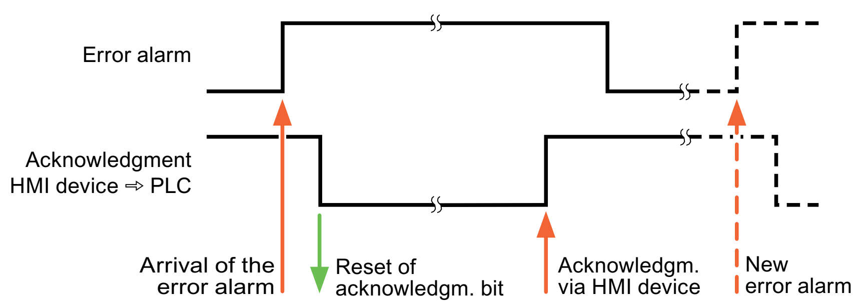
If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.12.5 Performance features of communication (Basic Panels)

1.12.5.1 SIMATIC LOGO! device dependency (Basic Panels)

Communication with the SIMATIC LOGO! controller

If you use devices from an earlier version of the TIA Portal with TIA Portal V13, it may not be possible to configure integrated connections to certain HMI devices.

Basic Panels V11.0

HMI devices	SIMATIC LOGO!
KP300 Basic	No
KP400 Basic	No
KTP400 Basic PN	No
KTP600 Basic DP	No
KTP600 Basic PN	No
KTP1000 Basic DP	No
KTP1000 Basic PN	No
TP1500 Basic PN	No

Basic Panels V12.0

HMI devices	SIMATIC LOGO!
KP300 Basic	Yes
KP400 Basic	Yes
KTP400 Basic PN	Yes
KTP600 Basic DP	Yes
KTP600 Basic PN	Yes
KTP1000 Basic DP	No
KTP1000 Basic PN	No
TP1500 Basic PN	Yes

Basic Panels V13.0

HMI devices	SIMATIC LOGO!
KTP400 Basic PN	Yes
KTP700 Basic PN	Yes
KTP700 Basic DP	Yes
KTP900 Basic PN	Yes
KTP1200 Basic PN	No
KTP1200 Basic DP	No

1.12.5.2 Valid data types for SIMATIC LOGO! (Basic Panels)

Valid data types for connections with SIMATIC LOGO!

Data type	Length
Bool	1 bit
Byte	1 byte
Int	2 bytes
DInt	4 bytes
Word	2 bytes
DWord	4 bytes
Array	--

1.13 Communication with other PLCs (Basic Panels)

1.13.1 Communication with other PLCs (Basic Panels)

Introduction

Communication with other PLCs is communication with PLCs that are not in the SIMATIC family.

These PLCs have proprietary protocols for data exchange. The protocols are configured as communication drivers in WinCC.

Communication drivers

The following communication drivers are supported in WinCC and are already installed:

- Allen-Bradley
  - Allen-Bradley EtherNet/IP
  - Allen-Bradley DF1
- Mitsubishi
  - Mitsubishi MC TCP/IP
  - Mitsubishi FX
- Modicon Modbus
  - Modicon Modbus TCP/IP
  - Modicon Modbus RTU
- Omron
  - Omron Host Link

Communication drivers in WinCC RT Professional

The following communication drivers are supported for RT Professional:

- Allen-Bradley
  - Allen-Bradley EtherNet/IP
- Mitsubishi
  - Mitsubishi MC TCP/IP
- Modicon Modbus
  - Modicon Modbus TCP

Connections between HMI devices and other PLCs

You configure the connections between HMI devices and other PLCs in the "Connections" editor of the HMI device. These connections are non-integrated connections.

1.13.2 Distinctive features when configuring (Basic Panels)

Distinctive features for data exchange

Distinctive features apply when configuring connections to other PLCs, compared to configuring integrated connections.

Note the following distinctive features when configuring:

- Addressing of tags
- Permitted data types
- Distinctive features when configuring area pointers
- Distinctive features when configuring alarms
- Distinctive features when configuring trends

For more detailed information on distinctive features when configuring, refer to Section "Data exchange" of the respective communication driver.

1.13.3 Communication drivers (Basic Panels)

1.13.3.1 Allen-Bradley (Basic Panels)

Allen-Bradley communication drivers (Basic Panels)

Introduction

This section describes the communication between an HMI device and PLCs that use Allen-Bradley communication drivers.

The following communication drivers are supported:

- Allen-Bradley EtherNet/IP
- Allen-Bradley DF1

Data exchange

Data is exchanged by means of tags or area pointers.

- Tags
  The PLC and the HMI device use process values for data exchange. You create tags in the configuration that point to addresses in the PLC. The HMI device reads the value from the defined address, and then displays it. The operator may also enter values on the HMI device, which are then written to the address in the PLC.
- Area pointers
  Area pointers are used to exchange specific data and are only set up when these data are used.

Allen-Bradley EtherNet/IP (Basic Panels)

Configuring a connection via Allen-Bradley EtherNet/IP (Basic Panels)

Introduction

You configure a connection to a PLC with an Allen-Bradley EtherNet/IP communication driver in the "Connections" editor of the HMI device.

The Ethernet interfaces are named differently depending on the HMI device.

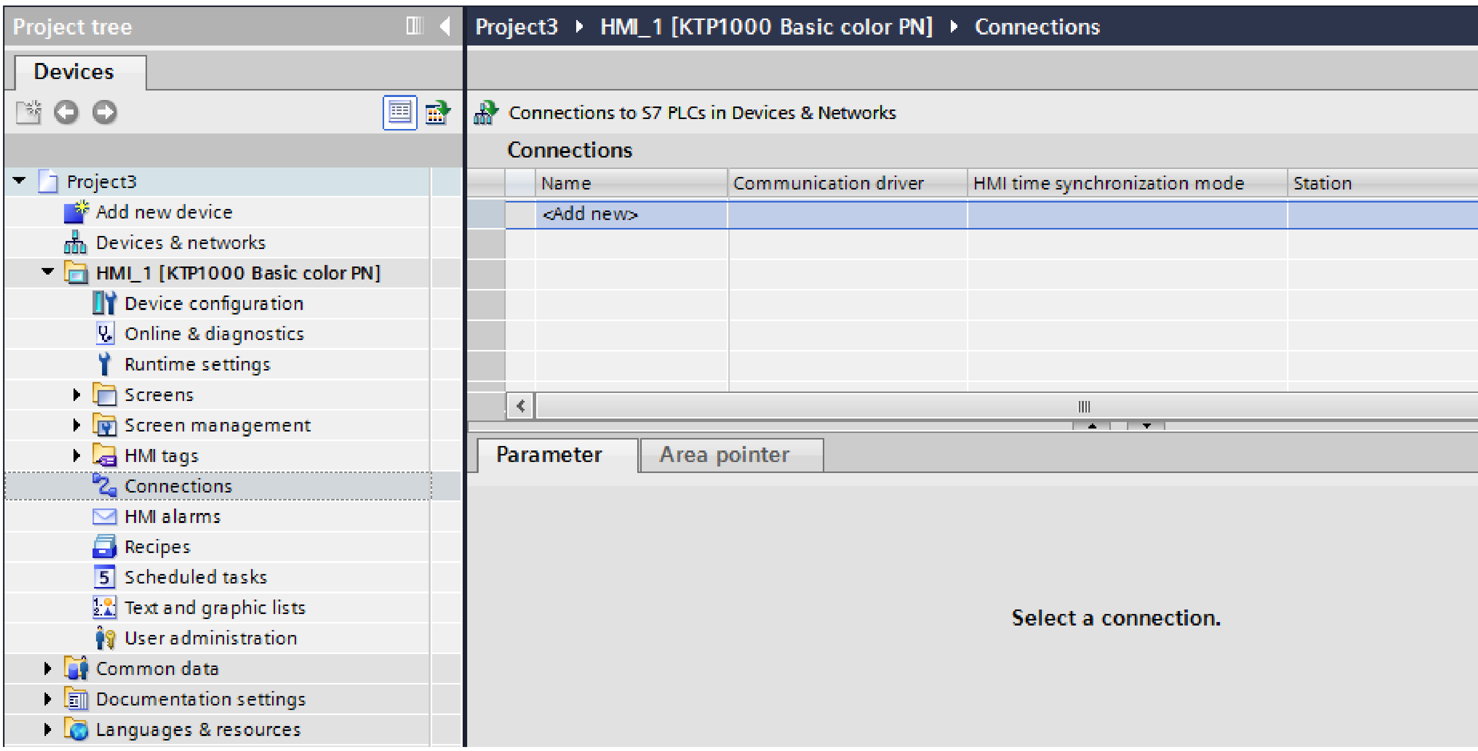
Example: PROFINET interface corresponds to the Ethernet interface

Requirements

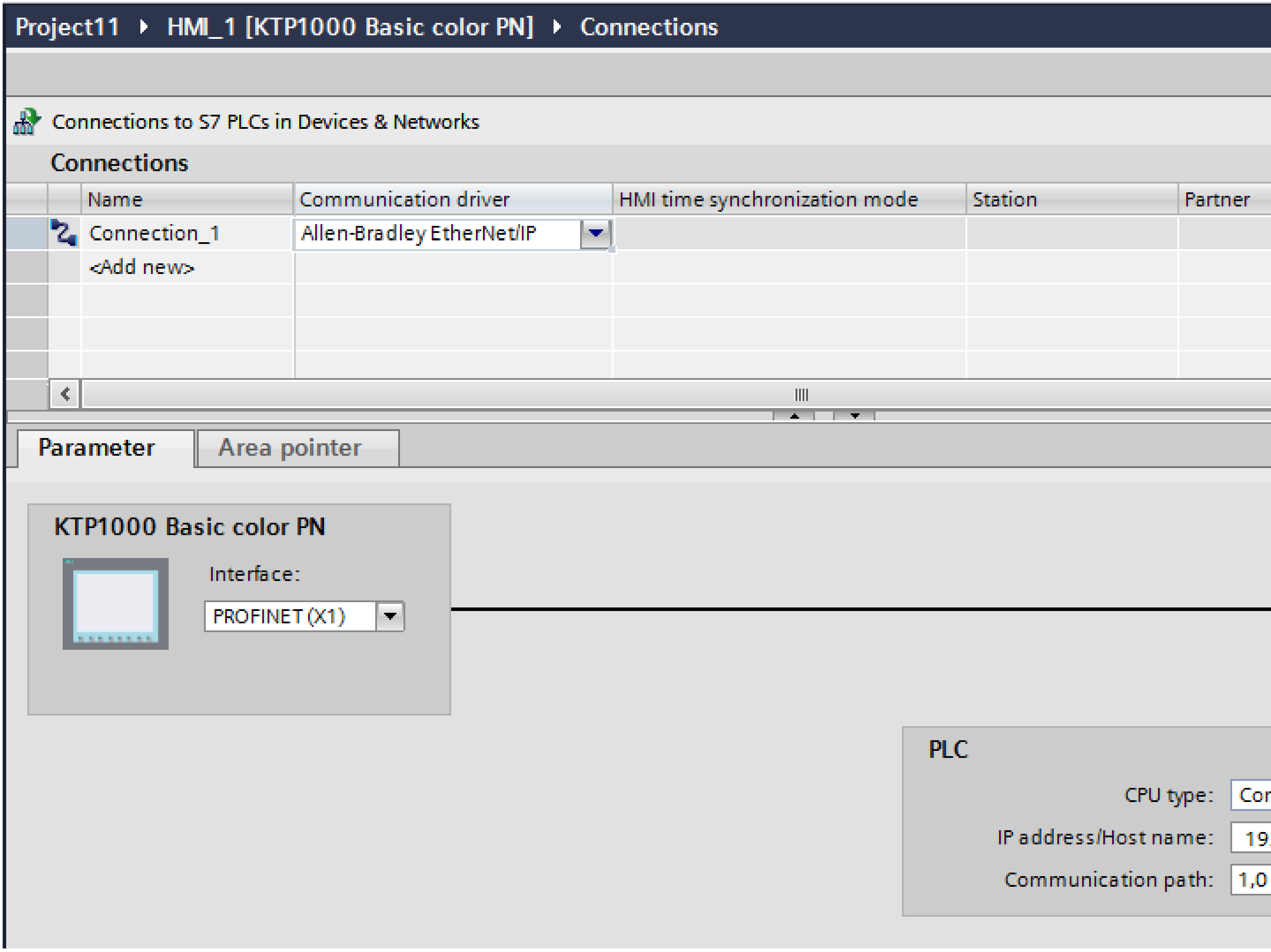
- A project is open.
- An HMI device has been created.

Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.
3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "Allen-Bradley EtherNet/IP" driver.



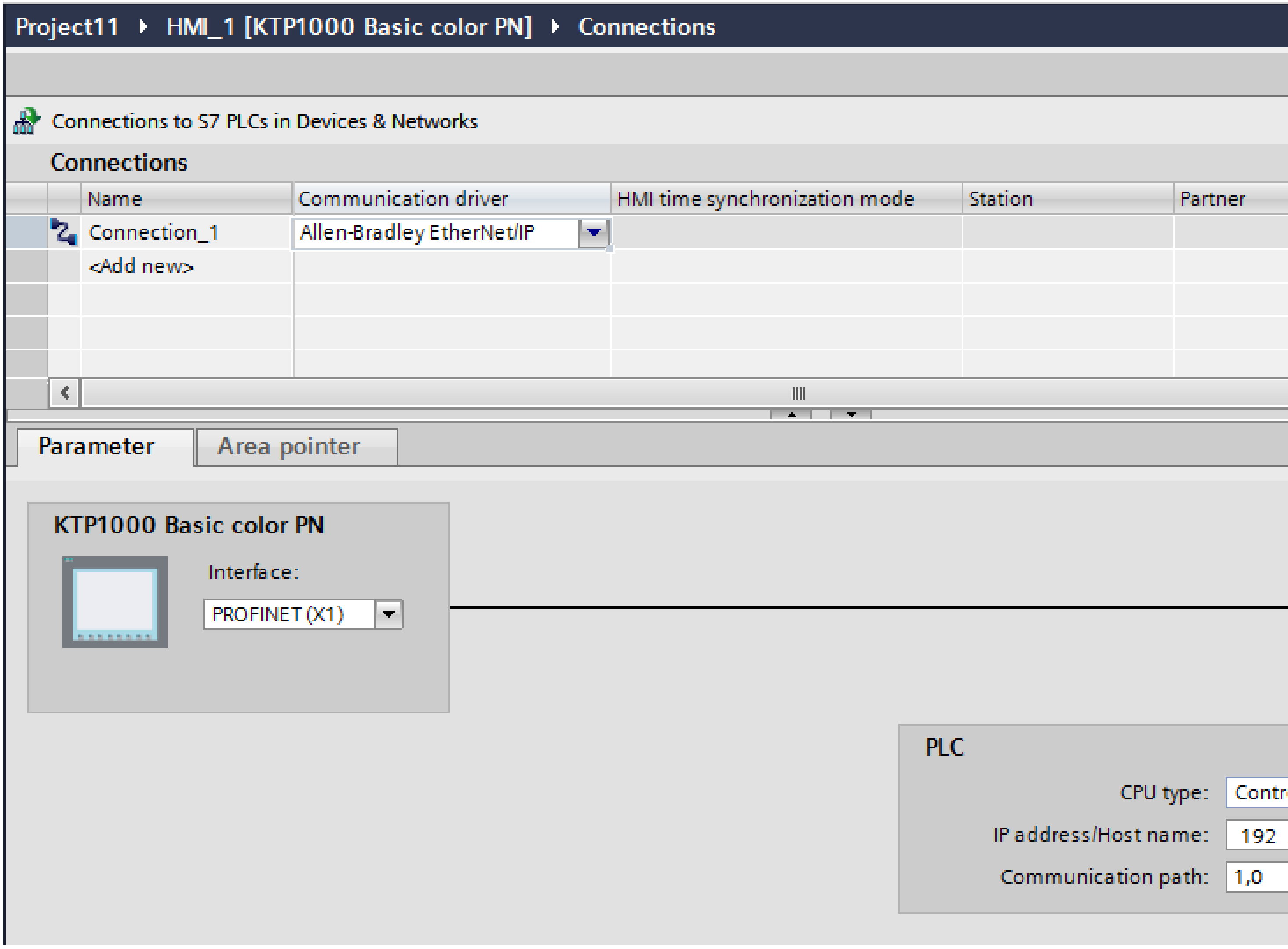
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".

Parameters for the connection (Allen-Bradley EtherNet/IP) (Basic Panels)

Parameters to be set

To assign the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "PLC" area is available for assigning parameters according to the interface used.



Parameters for the HMI device

You can select only one interface for the HMI device in the Inspector window under "Parameters". Depending on the HMI device, there are several interfaces available.

If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC. The IP address is transferred to the HMI device upon subsequent loading.

Note

The IP address in the control panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

The IP address already set up in the control panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

To set up the IP address of the HMI device:

1. Click on the HMI device.
2. Open the "Device configuration" editor.
3. Click the Ethernet interface.
4. Assign the IP address in the inspector window under:
   "General > PROFINET interface > Ethernet addresses"

Parameters for the PLC

- CPU type
  For "CPU type", set the CPU type of the PLC used.
- IP address
  Set the IP address or host name of the Ethernet/IP module of the PLC. Only the IP address can be used on a Basic Panel.
- Communication path
  Set the CIP path from the Ethernet module to the PLC. This establishes a logical connection between the Ethernet module and PLC, even if both devices are located in different CIP networks.
  For additional information see: AUTOHOTSPOT

Connecting HMI device to PLC (Basic Panels)

Connections via Allen-Bradley EtherNet/IP (Basic Panels)

Connection

The HMI device can be connected to the Allen-Bradley PLC using the following components:

- Existing Ethernet network that also contains the PLCs
- Cross-over Ethernet cable connected directly to the Ethernet interface of the CPU or the communication module

The connection of the HMI device to an Allen-Bradley PLC is limited primarily to the physical connection of the HMI device. Special blocks for the connection are not required in the PLC.

Communication types (Basic Panels)

Approved communication types with Allen-Bradley EtherNet/IP

The following communication types are system-tested and approved:

- Point-to-point connection to the approved PLCs
- Multipoint connection from a HMI device (Allen-Bradley Ethernet/IP-Client) with up to 4 PLCs with the respectively approved PLCs. CPU types can be mixed.

Connection

Connection with the following PLCs is approved with Allen-Bradley EtherNet/IP:

- CPU type: "ControlLogix, Compact Logix"
  - ControlLogix
    556x(1756-L6x) with Ethernet module 1756-ENBT
  - Guard Logix-System ControlLogix
    556xS(1756-L6xS) with Ethernet module 1756-ENBT
  - CompactLogix
  - 533xE(1769-L3xE) with Ethernet interface onboard
  - 532xE(1769-L2xE) with Ethernet interface onboard
  - 534x (1768-L4x) with Ethernet module 1768-ENBT
- CPU type: "SLC, MicroLogix"
  - MicroLogix 1100 (with Ethernet interface onboard)
  - MicroLogix 1400 (with Ethernet interface onboard)
  - SLC 5/05 (with Ethernet interface onboard)

Performance features of communication (Basic Panels)

Permitted data types for Allen-Bradley EtherNet/IP (Basic Panels)

Permitted data types

The table lists the data types that can be used when configuring tags and area pointers.

CPU type: ControlLogix, CompactLogix

Data type	Length
Bool	1 bit
DInt	4 bytes
Int	2 bytes
Real	4 bytes
SInt	1 byte
String	1 to 82 characters
UDInt	4 bytes
UInt	2 bytes
USInt	1 byte

Permitted data types arrays

Address	Permitted data types
Array	SInt, USInt, Int, UInt, DInt, UDInt, Real
Individual bits from the basic data types of the PLC SInt, USInt, Int, UInt, DInt, UDInt	Bool*

* Any changed value of certain defined bits is written back to the PLC. There is no check to determine whether any other bits have changed. The PLC (or other PLCs) may only read access the value.

CPU type: SLC, MicroLogix

Data type	Operand type	Length
ASCII	A	0 to 80 characters
Bool	N, R, C, T, B, S, I, O	1 bit
DInt	N	4 bytes
Int	N, R, C, T, S	2 bytes
Real	N, F	4 bytes
String	ST	1 to 82 characters
UDInt	N	4 bytes
UInt	N, R, C, T, B, I, O	2 bytes

Permitted data types arrays

Address	Permitted data types
Array	Int, UInt, DInt, UDInt, Real

Distinctive features for connections with Allen-Bradley Ethernet/IP

With the communication driver Allen Bradley Ethernet/IP and the CPU type SLC, MicroLogix, you can only use array tags for discrete alarms and trends.

Note

I/O modules with 8 or 16 ports occupy one data word on the PLC.

I/O modules with 24 or 32 ports occupy two data words.

The HMI device does not output an error message if using non-existent bits.

You should always make sure that I/O modules with 8 or 24 ports only occupy the bits that are actually assigned to a port.

Supported CPU types for Allen-Bradley EtherNet/IP (Basic Panels)

CPU types

The following CPU types are supported for configuring the Allen-Bradley EtherNet/IP communication driver.

- CompactLogix
  - 1769-L2xE with Ethernet interface onboard
  - 1769-L3xE with Ethernet interface onboard
  - 1768-L4x with Ethernet module 1768-ENBT
- ControlLogix
  - 1756-L6x with Ethernet module 1756-ENBT
- GuardLogix
  - 1756-L61S with Ethernet module 1756-ENBT
  - 1756-L62S with Ethernet module 1756-ENBT
  - 1756-L63S with Ethernet module 1756-ENBT
- MicroLogix
  - MicroLogix 1100 / 1400
- SLC50x
  - SLC5/05

Addressing in the C.Logix CPU type (Basic Panels)

Addressing (Basic Panels)

Addressing

A tag is uniquely referenced in WinCC by means of an address in the PLC. The address must correspond with the tag name in the PLC. The tag address is defined by a string with a length of up to 128 characters.

Using characters for addressing

Valid characters for tag addressing:

- Letters (a to z, A to Z)
- Numbers (0 to 9)
- Underscore (_)

The tag address consists of tag name and other character strings used to specify the tag in the PLC.

Tag name properties:

- The tag name may begin but not end with an underscore character.
- Strings with successive underscore and space characters are invalid.
- The address may not exceed a length of 128 characters.

Note

The characters reserved for tag addressing may not be used in program/tag names or at any other address instance.

The reserved characters are listed below:

Reserved character	Function
.	Element delimiter
:	Definition of a program tag
,	Delimiter for addressing multi-dimensional arrays
/	Reserved for bit addressing.
[ ]	Addressing of array elements or arrays

PLC and program tags

The Allen-Bradley EtherNet/IP communication driver supports addressing of PLC tags (global project tags) and/or program tags (global program tags).

A program tag is declared based on the program name in the PLC and actual tag name which are delimited by colon. PLC tags are simply addressed by their name.

Note

Addressing errors

Addressing errors occur when the tag name and data type are inconsistent.

Note that the tag name defined in the address field in WinCC must match the tag name in the PLC. Make sure that the data types of tags in WinCC match the data types in the PLC.

Note

Module-specific tags, e.g. for data on input and output modules, cannot be addressed directly. Instead, use an alias tag in the PLC.

Example: Local:3:O. Data cannot be addressed in WinCC.

If the alias "MyOut" is defined for Local:3:O in the PLC, you can address with WinCC via MyOut.Data.

Addressing syntax (Basic Panels)

Notation of addresses

The tables below define the notation for the individual addressing options for Allen-Bradley EtherNet/IP.

Table 1-2 Access to arrays, basic data types and structure elements

Data types	Type	Address
Basic data types	PLC tag	Tag name
	Program tag	Programname:tagname
Arrays	PLC tag	Array tag
	Program tag	Program name: array tag
Bits	PLC tag	Tagname/bitnumber
	Program tag	Programname:tagname/bitnumber
Structure elements	PLC tag	Structure tag. Structure element
	Program tag	Program name: structure tag. structure element

Note

Bit addressing with the data types Bool, Real and String is not permitted and will cause an addressing fault.

Description of the syntax

Syntax description:

```
(Programname:)tagname([x(,y)(,z)]){.tagname([x(,y)(,z)])}(/
bitnumber)
```

- The "( )" defines an optional, single instance of an expression.
- The "{ }" defines an optional expression with multiple single instances.

The address string length may not exceed 128 characters.

Addressing types (Basic Panels)

Arrays

An array is a data structure that includes a number of data of the same type. WinCC only supports one-dimensional arrays.

In the address column of the tag editor, enter the array name possibly by specifying a start element. The length is defined in the Array Elements input box of the tag editor. If array limits in the PLC are exceeded (due to faulty indexing), addressing errors result.

These arrays must be declared in the PLC as controller or program tags.

Two- or three-dimensional arrays in the PLC can only be addressed in WinCC if these can be mapped area-wise onto one-dimensional arrays .

Note

During all read accesses and all write accesses, all array elements of a tag are always read or written, respectively. The contents of an array tag which is interconnected with a PLC are always transferred whenever there is a change. The HMI device and the PLC cannot concurrently write data to the same array tag for this reason. Instead of writing data only to a single element, the program writes the entire array to the PLC.

Array elements

Elements of one-dimensional, two-dimensional and three-dimensional arrays in the PLC are indexed by setting an index and the corresponding notation in the tag editor. Array addressing starts at element "0", with arrays of all basic types being valid for element addressing. Read/write operations are only carried out at the addressed element, and not for the entire array.

Bits and bit tags

Bit access is allowed to all basic data types with the exception of Bool, Real and String. Bit addressing is also allowed at array/structure elements. The Bool data type is set in WinCC when bits and bit tags in the basic data types are addressed.

Single-digit bit numbers are addressed with "/x" or "/0x" (x = bit number). Bit numbers are defined by up to two digits.

Note

With the "Bool" data type in the data types SInt, Int and DInt, after changing the specified bit the complete tag is then written in the PLC again. In the meantime, no check is made as to whether other bits in the tag have since changed. Therefore, the PLC may have only read access to the specified tag.

Structures

User-defined data types are created by means of structures. These structures group tags of different data types. Structures may consist of basic types, arrays and of other structures. In WinCC, only structure elements are addressed and not entire structures.

Structure elements

Structure elements are addressed by means of the name of the structure and of the required structure element. This addressing is separated by point. In addition to basic data types, the structure elements may represent arrays or other structures. Only one-dimensional arrays may be used as a structure element.

Note

The nesting depth of structures is only limited by the maximum length of 128 characters for the address.

Address multiplexing (Basic Panels)

Address multiplexing

Address multiplexing is possible with the CompactLogix, ControlLogix CPU type.

Address multiplexing requires two tags:

- "Tag_1" of data type "String"; contains a logical address such as "HMI:Robot5.Block5" as value.
  The value may change to a second valid address, for example, "HMI:Robot4.Block3".
- "Tag_2" is a tag in which the "Allen-Bradley EtherNet/IP" communication driver is set up as a connection.
  Enter a valid name of an HMI_tag in square brackets as the address.
  - e.g.: "[Tag_1]"
  - The tag must be of the String data type.
  - The square brackets indicate address multiplexing.
  - The address is derived from the actual value in "Tag_1".

  Note

  You can only multiplex entire Allen-Bradley EtherNet/IP addresses. Multiplexing of address elements is not possible. "HMI:Robot[Tag_1].Block5" is an invalid address.

  You can optionally click the arrow right icon in the "Address" column. Replace the "Constant" with the "Multiplex" entry by clicking the arrow on the left edge of the next address dialog box. Now the tag selection list only returns tags of data type "String".

  You can also configure a function triggered by a "change of value" event for multiplexed tags.

Examples for addressing (Basic Panels)

Example of a table for addressing

The table below defines the basic variants for addressing PLC tags. Other addressing variants are possible by means of combination.

Type	Type	Address
General	PLC tag	Tag name
	Program tag	Program:tagname
Array	Access to an element of a 2-dimensional array	Arraytag[Dim1,Dim2]
	Element of structure array (1-dimensional)	Arraytag[Dim1].structureelement
	Bit in element basic type array (2-dimensional)	Arraytag[Dim1,Dim2]/Bit
Structure	Array in structure	Structuretag.arraytag
	Bit in the element of an array in the substructure	Structuretag.structure2.arraytag [element]/bit

Note

Program tags are addressed by leading the address with the program name derived from the PLC with colon delimiter.

Example: Programname:arraytag[Dim1,Dim2]

Access to array elements

Type	Address
PLC tag	Arraytag[Dim1]
	Arraytag[Dim1,Dim2]
	Arraytag[Dim1,Dim2,Dim3]
Program tag	Programname:arraytag[Dim1]
	Programname:arraytag[Dim1,Dim2]
	Programname:arraytag[Dim1,Dim2,Dim3]

Examples: Communication path (Basic Panels)

Example 1:

Connection with a PLC in the same Allen-Bradley rack.

1,0

Number	Meaning
1	Stands for a backplane connection.
0	Stands for a CPU slot number.

Example 2:

Connection with a PLC in remote Allen-Bradley racks. Two Allen-Bradley racks are networked on Ethernet.

1,2,2,190.130.3.101,1,5

Number	Meaning
1	Backplane connection
2	Stands for the CPU slot number of the second Ethernet module.
2	Stands for an Ethernet connection.
190.130.3.101	IP address of a remote AB rack on the network – in particular the third Ethernet module
1	Backplane connection
5	Slot number of the CPU

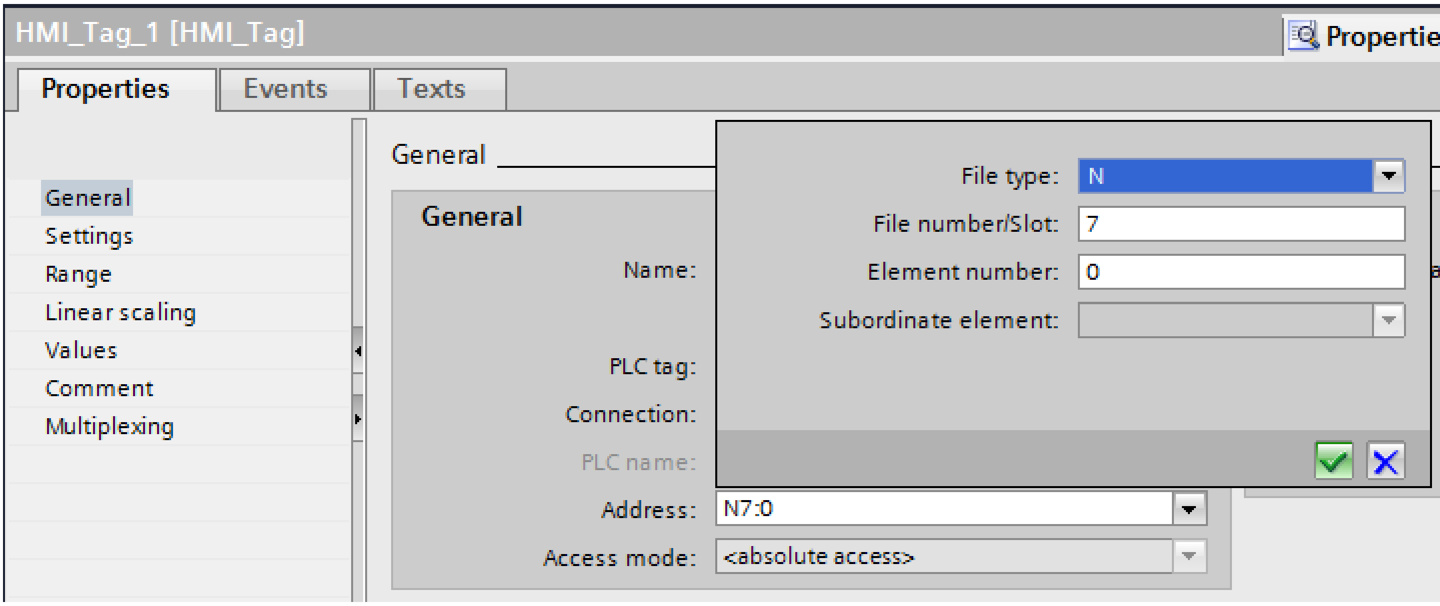
Addressing in the SLC, MicroLogix CPU type (Basic Panels)

Addressing

The addressing in the SLC, MicroLogix CPU type is entered in the following order:

- Operand type
- File number
- Element number

- Child element
- Bit number



The address then appears in the following format without spaces:

- File type file number : Element number . Child element
- e.g. T10:2.ACC

Operand type

You have the following options under operand type:

- I
- O
- S
- B
- C
- T
- R
- F
- N
- ST
- A

File number

Select the number between two limits under file number:

- Low limit
- High limit

The limit values depend on the selected operand type.

Child element

You can select a child element when you have selected one of the following operand types:

- R
- C
- T

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Note

If running the CompactLogix PLC with firmware earlier than version 18, you will possibly have to restart the HMI device following the transfer of the PLC program.

You could also terminate the connection before transferring the PLC program and set up the connection again after having completed the transfer of the PLC program.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance. Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.
- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Allen-Bradley DF1 (Basic Panels)

Configuring a connection via Allen-Bradley DF1 (Basic Panels)

Introduction

You configure a connection to a PLC with an Allen-Bradley DF1 communication driver in the "Connections" editor of the HMI device.

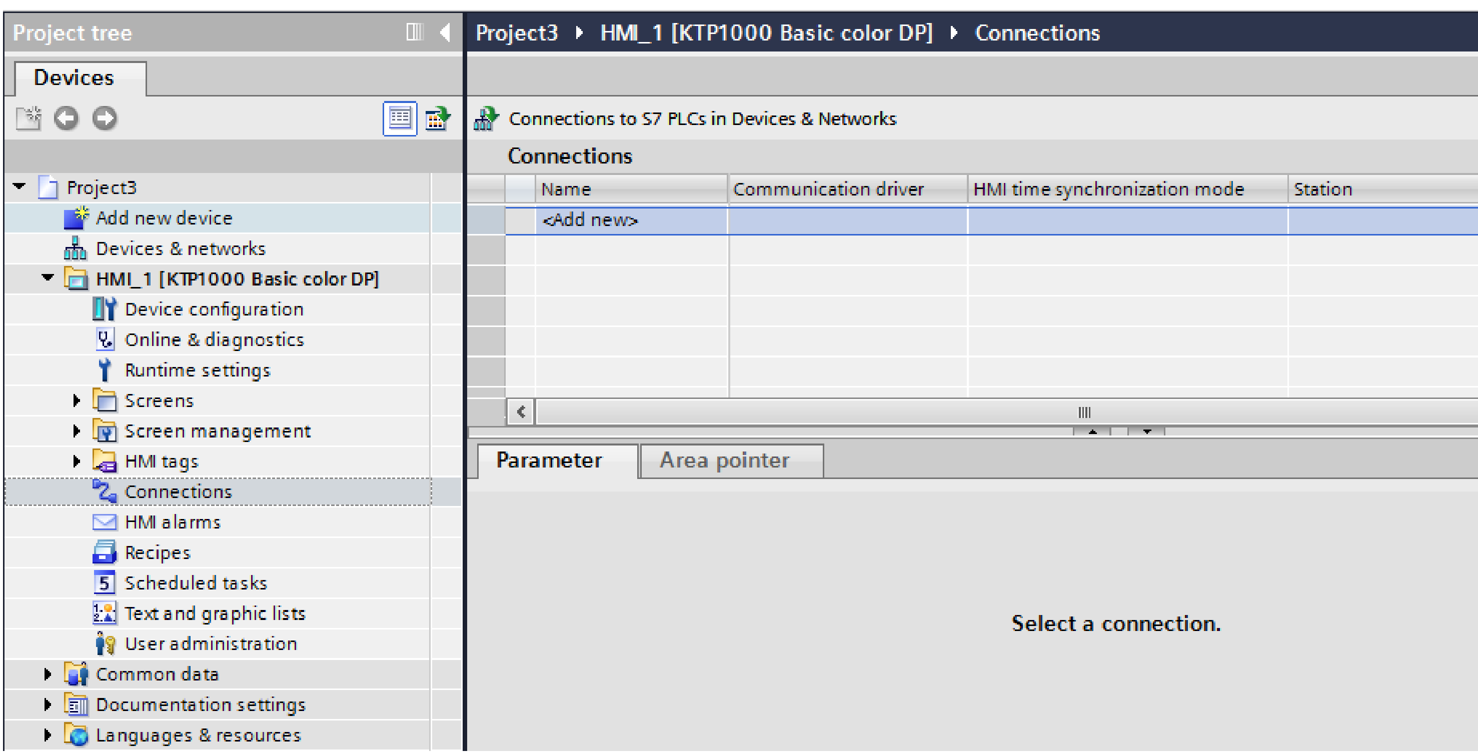
The interfaces are named differently depending on the HMI device.

Requirements

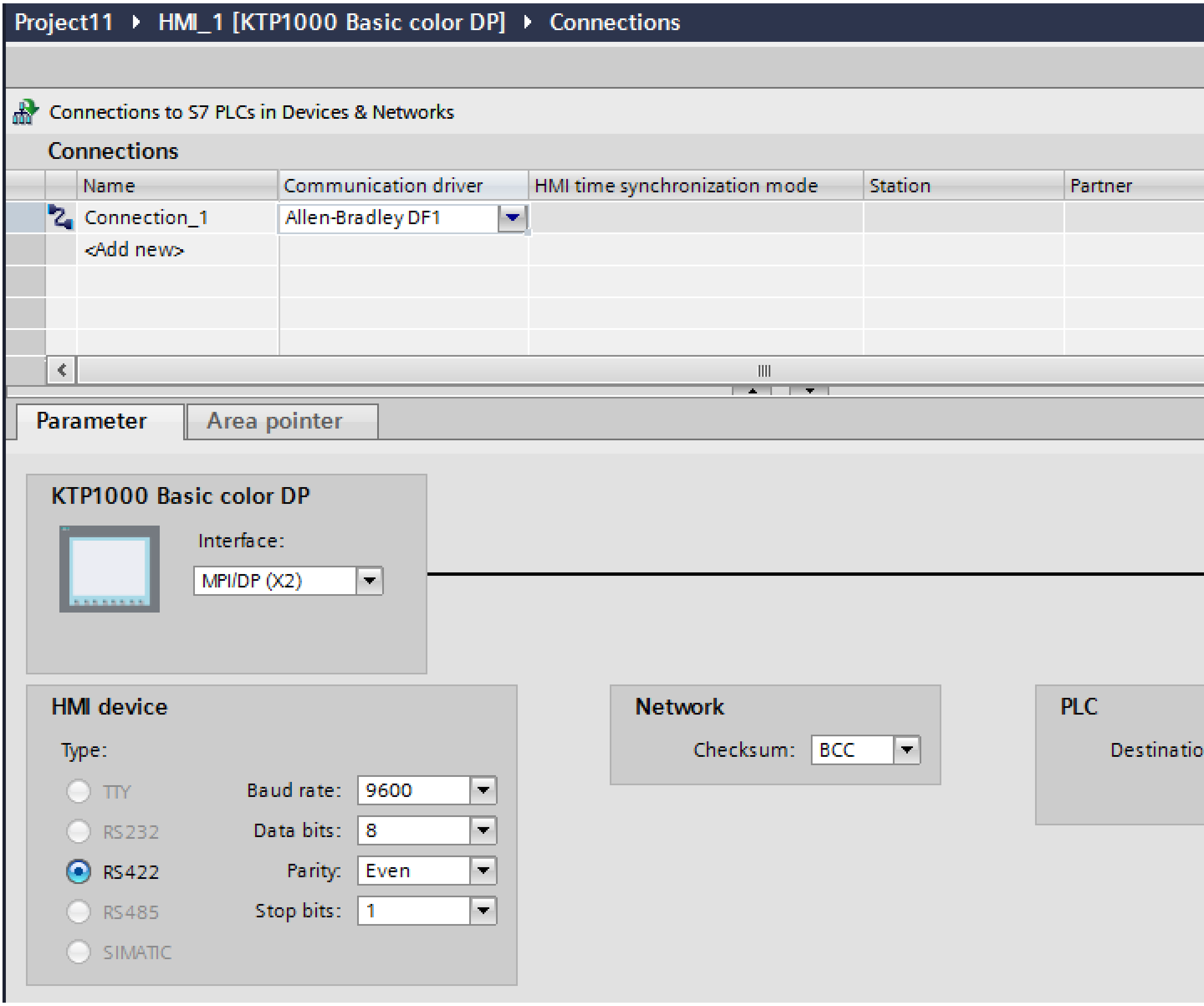
- A project is open.
- An HMI device has been created.

Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.
3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "Allen-Bradley DF1" driver.



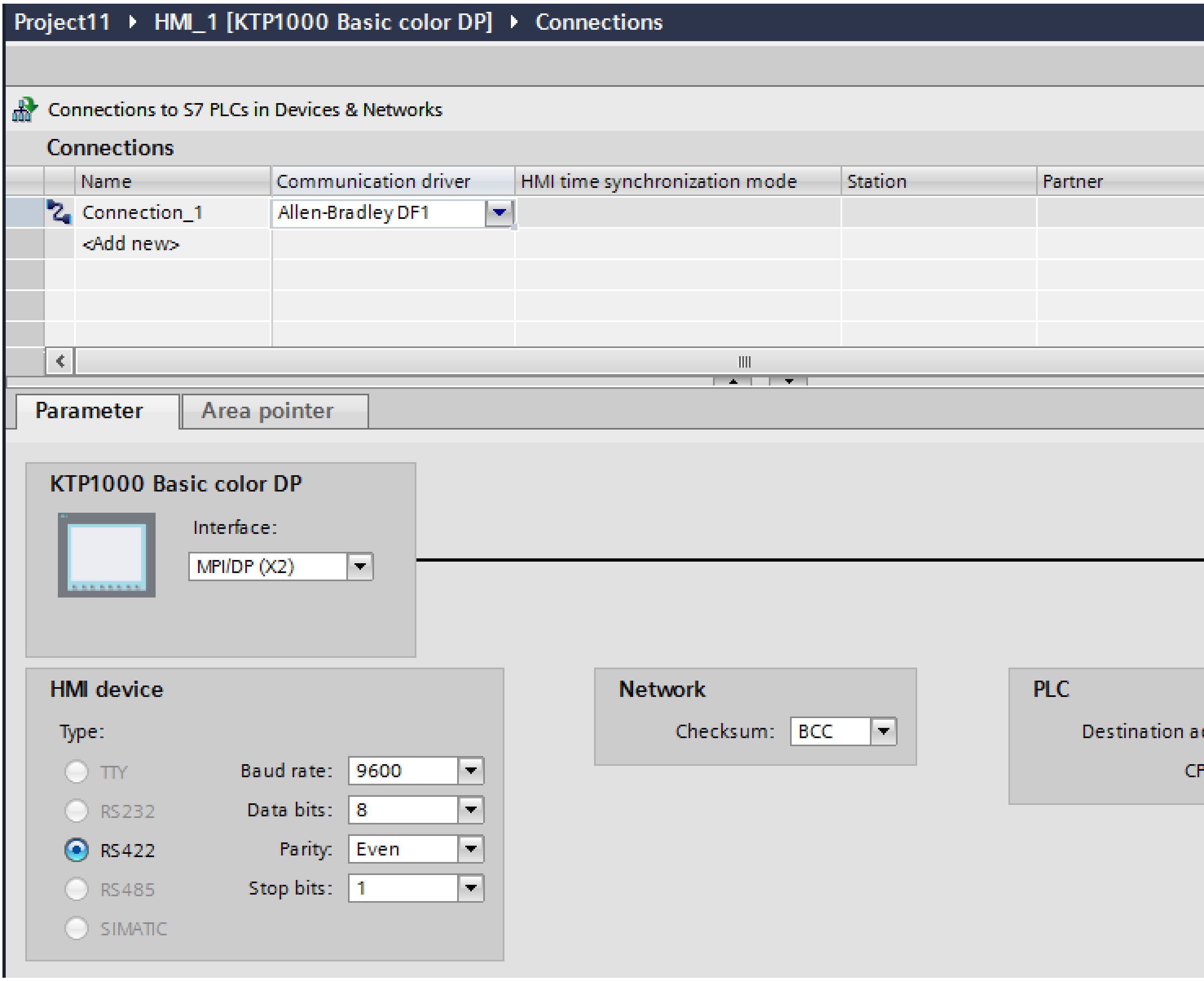
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".

Parameters for the connection (Allen-Bradley DF1) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device", "Network", and "PLC" areas are available for assigning parameters according to the interface used.



Parameters for the HMI device

- Interface
  Under "Interface", you select the interface of the HMI device to which the PLC is connected. For additional information, refer to the device manual for the HMI device.
- Type
  Specifies the physical connection used.

Note

If you are using the IF1B interface, you must also reconnect the RS-485 receive data and the RTS signal to the rear of the HMI devices via 4 DIP switches.

- Baud rate
  For "Baud rate", select the transmission speed between the HMI device and PLC.
- Data bits
  For "Data bits", you can choose between "7 bits" and "8 bits".
- Parity
  For "Parity", you can choose from "None", "Even", and "Odd".
- Stop bits
  For "Stop bits", you can choose between 1 and 2 bits.

Parameters for the network

- Checksum
  For "Checksum", choose the method for determining the error code: "BCC" or "CRC".

Parameters for the PLC

- Destination address
  For "Destination address", choose the PLC address. If there is a point-to-point DF1 connection, you set the address "0".
- CPU type
  For "CPU type", set the CPU type of the PLC used.

Note

Assign the DF1 FULL-DUPLEX driver in the CPU as follows: "NO HANDSHAKING" for "Control Line" and "AUTO-DETECT" for "Embedded Responses".

Connecting HMI device to PLC (Basic Panels)

Connections via Allen-Bradley DF1 (Basic Panels)

Connection

The connection is established when you have matched the parameters of the PLC and the HMI device. Special blocks for the connection are not required in the PLC.

Note

Rockwell offers a variety of communication adapters for integrating "DF1 devices" for the DH485, DH, and DH+ networks. Of these connections, the direct connection and the connection via KF2 and KF3 module are approved. None of the other connections have been system-tested by SIEMENS AG and are therefore not approved.

Communication partners for Allen-Bradley DF1 (Basic Panels)

Connectable PLCs

The communication drivers listed below support Allen-Bradley PLCs: :

PLC	DF1 (point-to-point) RS-232	DF1 (point-to-point) RS-422	DF1 (multipoint) over KF2 module to DH+ LAN RS-232/RS-422	DF1 (multipoint) over KF3 module to DH485 LAN RS-232
SLC500	–	–	–	X
SLC501	–	–	–	X
SLC502	–	–	–	X
SLC503	X	–	–	X
SLC504	X	–	X	X
SLC505	X	–	–	X
MicroLogix	X	–	–	X
PLC-5 [1)]	X	X	X	–

[1)] Processors released for PLC-5: PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60, and PLC-5/80.

Communication types (Basic Panels)

PLCs with Allen-Bradley DF1 communication driver

The communication between the HMI device and the following Allen Bradley PLCs is described in this section:

- SLC500
- SLC501
- SLC502
- SLC503
- SLC504
- SLC505
- PLC5
- MicroLogix

In these PLCs the connection is made by the PLC-internal protocols Allen Bradley DF1, Allen Bradley DH485 and Allen Bradley DH+.

The Allen-Bradley DF1 communication driver is used here, the protocol of which is converted into one of the other two PLC-internal protocols in multipoint communication with the communication modules KF2 (Allen Bradley DH+) and KF3(Allen Bradley DH485).

Enabled types of communication with Allen-Bradley DF1

The following communication types are system-tested and enabled:

- HMI (Allen Bradley DF1)
  Point-to-point connection
- HMI (Allen Bradley DF1)
  Via KF2 module to Allen Bradley DH+ (communication with up to 4 PLCs)
- HMI (Allen Bradley DF1)
  Via KF3 module to Allen Bradley DH485 (communication with up to 4 PLCs)

Connectable PLCs

The Allen Bradley DF1 communication driver is available for the following Allen-Bradley PLCs:

PLC	DF1 (point-to-point) RS 232	DF1 (point-to-point) RS 422	DF1 (multipoint) via KF2 module to DH+ LAN RS 232/RS 422	DF1 (multipoint) via KF3 module to DH485 LAN RS 232 [2)]
SLC500	–	–	–	X
SLC501	–	–	–	X
SLC502	–	–	–	X
SLC503	X [2)]	–	–	X

PLC	DF1 (point-to-point) RS 232	DF1 (point-to-point) RS 422	DF1 (multipoint) via KF2 module to DH+ LAN RS 232/RS 422	DF1 (multipoint) via KF3 module to DH485 LAN RS 232 [2)]
SLC504	X [2)]	–	X	X
SLC505	X [2)]	–	–	X
MicroLogix	X [2)]	–	–	X
PLC-5 [1)]	X	X	X	–

1) Only the following processors are approved for PLC-5: PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60 und PLC-5/80.

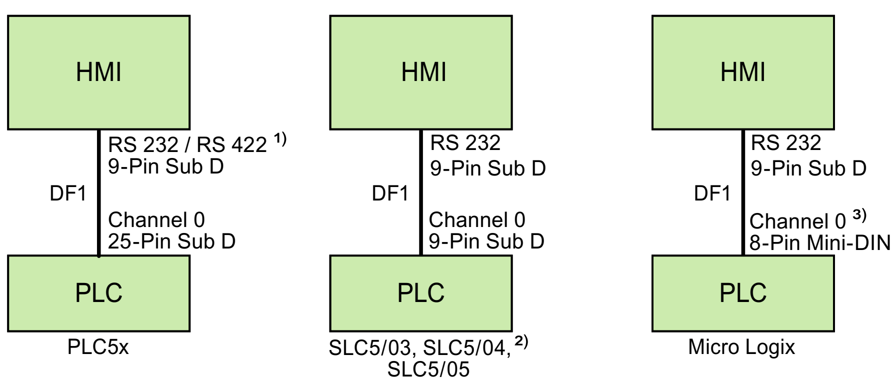
2) For HMI devices which only have an RS 422/485 interface and the communication partner is an RS 232 interface, the RS 422/232 converter is tested and approved.

Order number: 6AV6 671-8XE00-0AX0

DF1 protocol with multi-point connection (Basic Panels)

Point-to-point connection with DF1 protocol

Only point-to-point connections can be established with the DF1 protocol.



1) Only RS 232 is possible for Panel PC and PC.

2) A point-to-point connection to the SLC500, SLC501, and SLC502 PLCs via DF1 is not possible.

3) For MicroLogix ML1500 LRP, Channel 1 (9-pin Sub D) is also possible.

Connecting cable

HMI panel interface used	For connection to PLC5x	For connection to SLC5/03, SLC5/04, SLC5/05	For connection to MicroLogix
RS 232 9-pin	Allen-Bradley cable 1784-CP10	Allen-Bradley cable 1747-CP3	Allen-Bradley cable 1761-CBL-PM02
RS 422 9-pin	Connecting cable 9-pin Sub D RS 422	—	—

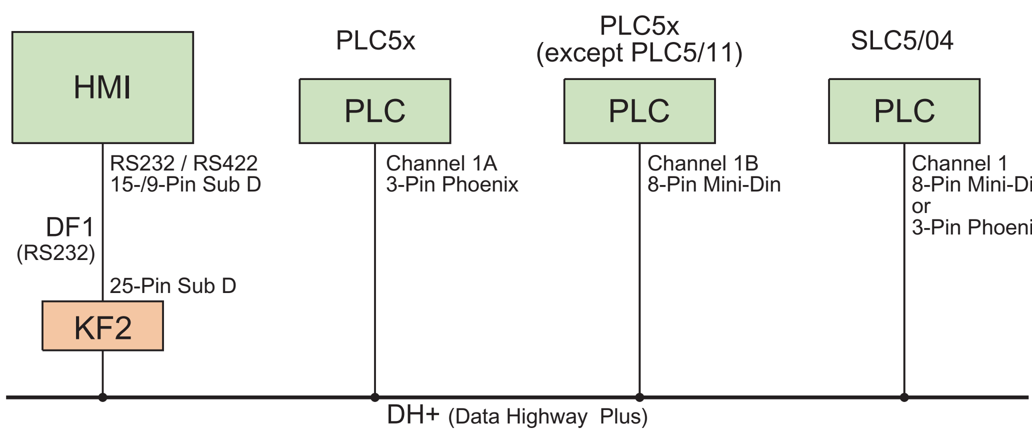
Refer to the relevant device manual to determine which HMI device interface is to be used.

The cable pin assignments can be found in Section "Connecting cables for Allen-Bradley".

DF1 protocol with multi-point connection via KF2 module (Basic Panels)

DF1 protocol with multi-point connection via KF2 module to DH+ LAN

The use of a KF2 protocol interface module enables a connection to be made to PLCs in the DH+ LAN (Data Highway Plus Local Area Network).



Connecting cable

HMI panel interface used	For connection to KF2 interface module
RS 232 9-pin	Allen-Bradley cable 1784-CP10 and 25-pin socket/socket adapter
RS 422 9-pin	9-pin Sub D RS 422 connecting cable and 25-pin female/female adapter

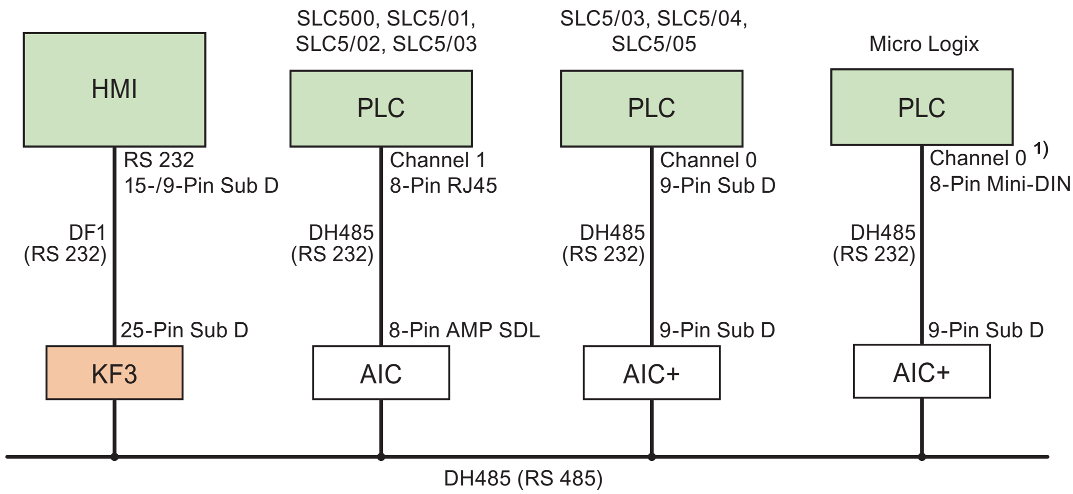
Refer to the Allen-Bradley documentation for the cable connection from the PLCs to the DH+ data bus.

Refer to the relevant device manual to determine which HMI device interface is to be used.

The cable pin assignments can be found in Section "Connecting cables for Allen-Bradley".

DF1 protocol with multi-point connection via KF3 module (Basic Panels)

DF1 protocol with multi-point connection via KF3 module to DH485 LAN



1) For MicroLogix ML1500 LRP, Channel 1 (9-pin Sub D) is also possible.

Connecting cable

HMI panel interface used	For connection to KF3 interface module
RS 232 9-pin	Allen-Bradley cable 1784-CP10 and 25-pin socket/socket adapter

Refer to the relevant device manual to determine which HMI device interface is to be used.

The cable pin assignments can be found in Section "Connecting cables for Allen-Bradley".

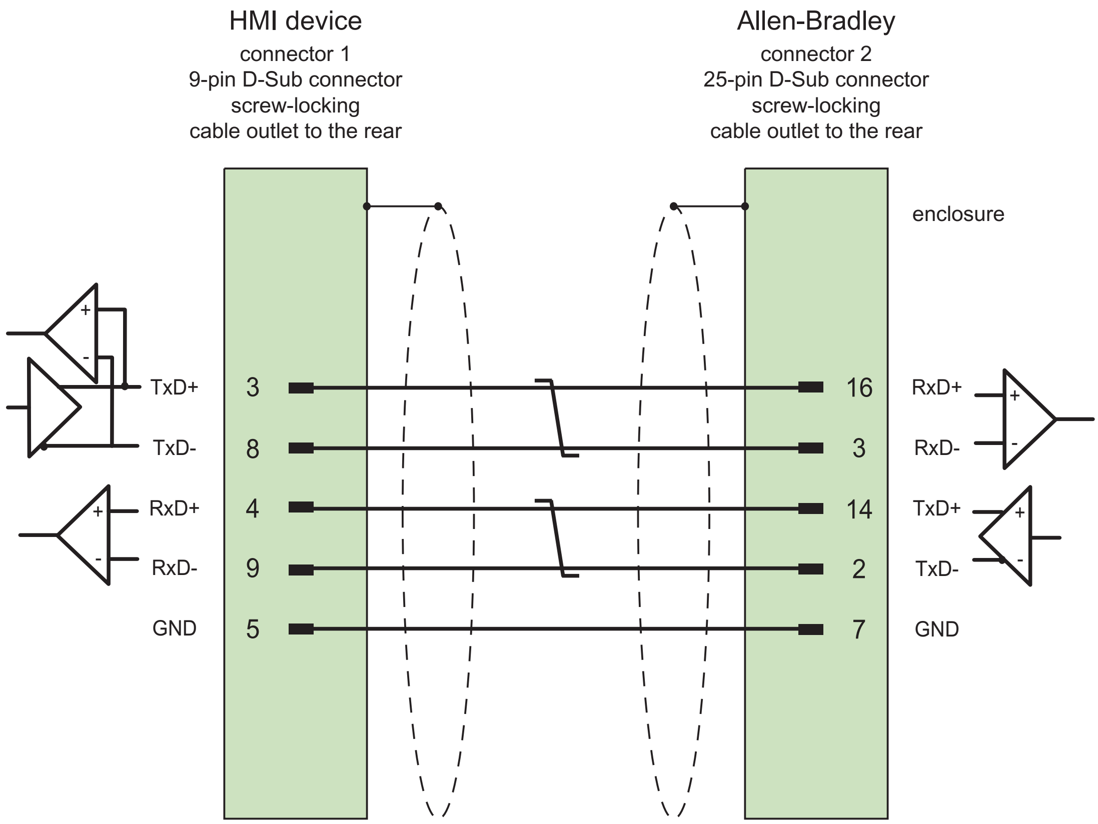
Connecting cables for Allen-Bradley DF1 (Basic Panels)

Connecting cable 9-pin Sub D RS 422 for Allen-Bradley (Basic Panels)

Connecting cable 9-pin Sub D RS 422

For interconnecting the HMI device (RS 422, 9-pin sub D) - PLC5x, KF2, KF3

You require an additional 25-pin, female / female adapter (gender changer) for interconnections with KF2 and KF3.



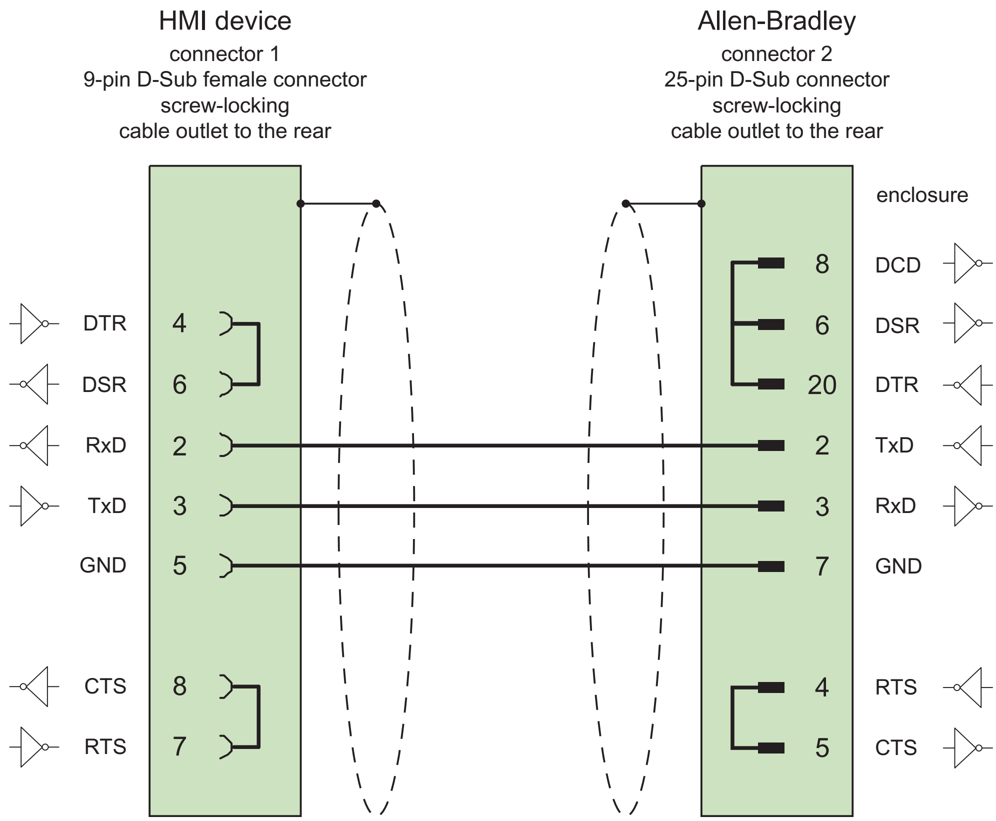
Shield with large-area contact to housing at both ends, interconnected shield contacts
Cable: 3 x 2 x 0.14 mm², shielded,
max. length 60 m

Connecting cable 1784-CP10, RS 232, for Allen-Bradley (Basic Panels)

Allen-Bradley cable 1784-CP10

For interconnecting the HMI device (RS 232, 9-pin sub D) - PLC5x, KF2, KF3

You require an additional 25-pin, female / female adapter (gender changer) for interconnections with KF2 and KF3.



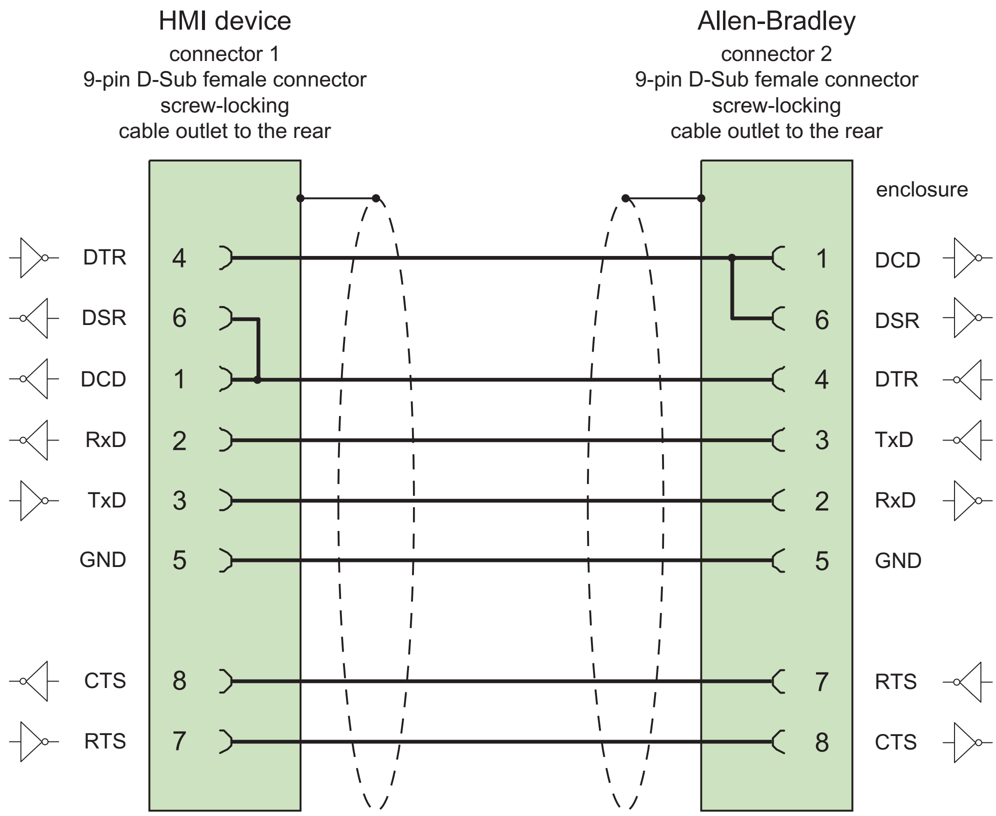
Screen connected with housing over large area on both sides

max. length 15 m

Connecting cable 1747-CP3, RS-232, for Allen-Bradley (Basic Panels)

Allen-Bradley cable 1747-CP3

For interconnecting the HMI device (RS 232, 9-pin sub D) - SLC503, SLC504, SLC505 (Channel 0), AIC+



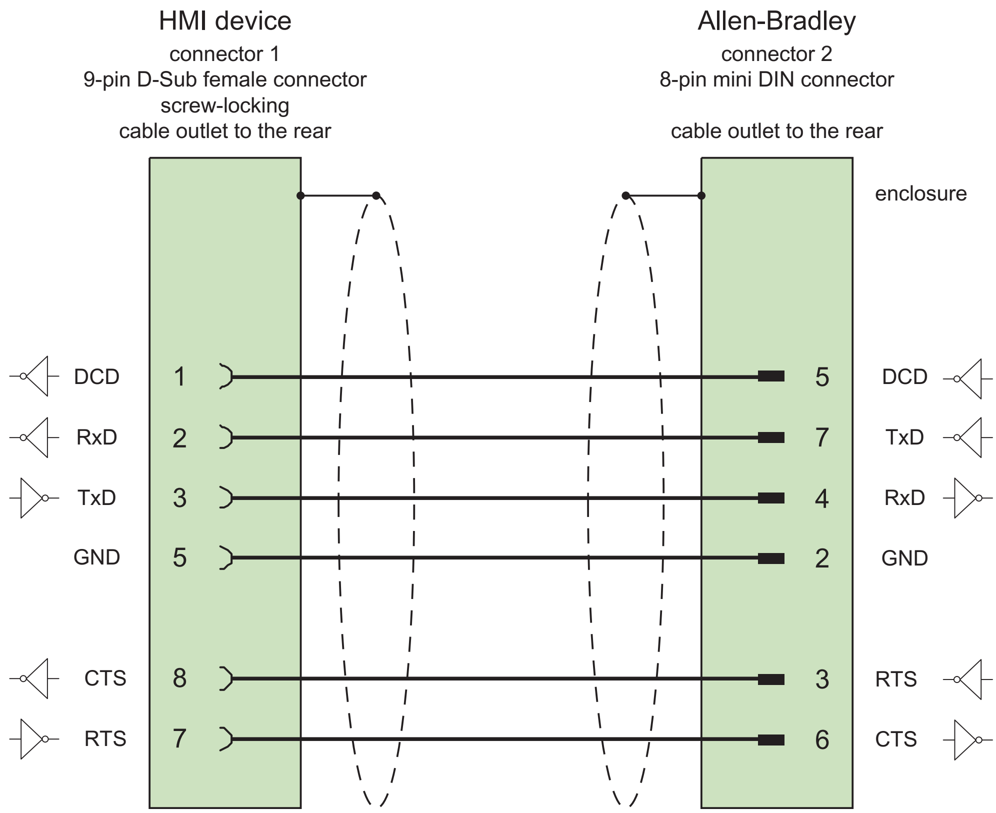
Screen connected with housing over large area on both sides

max. length 3 m

Connecting cable 1761-CBL-PM02, RS-232, for Allen-Bradley (Basic Panels)

Allen-Bradley cable 1761-CBL-PM02

For interconnecting the HMI device (RS 232, 9-pin sub D) - Micro Logix, AIC+



Screen connected with housing over large area on both sides

max. length 15 m

Performance features of communication (Basic Panels)

Permitted data types for Allen-Bradley DF1 (Basic Panels)

Permitted data types for Allen-Bradley DF1

The table lists the user data types that can be used when configuring tags and area pointers.

Data type	Operand type	Length
ASCII	A [1)]	1 to 80 characters
Bool	N, R, C, T, B, S, I, O	1 bit
Int	N, R, C, T, S	2 bytes
DInt	N	4 bytes
UInt	N, R, C, T, B, I, O	2 bytes
UDInt	N	4 bytes
Real	N, F [1)]	4 bytes

1) Selectable depending on the selected CPU type.

Abbreviations

In WinCC, formats of the data types are abbreviated as follows:

- UNSIGNED INT = UInt
- UNSIGNED LONG = UDInt
- SIGNED INT = Int
- SIGNED LONG = DInt

Distinctive features for connections with Allen-Bradley DF1

With Allen Bradley DF1, array tags may only be used for discrete alarms and trends.

Note

I/O modules with 8 or 16 ports occupy one data word on the PLC.

I/O modules with 24 or 32 ports occupy two data words.

The HMI device does not output an error message if using non-existent bits.

You should always make sure that I/O modules with 8 or 24 ports only occupy the bits that are actually assigned to a port.

Supported CPU types for Allen-Bradley DF1 (Basic Panels)

CPU types

The following CPU types are supported for configuring the Allen-Bradley DF1 communication driver.

- SLC
  - SLC500
  - SLC501
  - SLC502
  - SLC503
  - SLC504
  - SLC505
- MicroLogix
  - MicroLogix 1x00
  - MicroLogix 1100 / 1400
- PLC 5
  - PLC-5/11
  - PLC-5/20
  - PLC-5/40
  - PLC-5/60
  - PLC-5/80

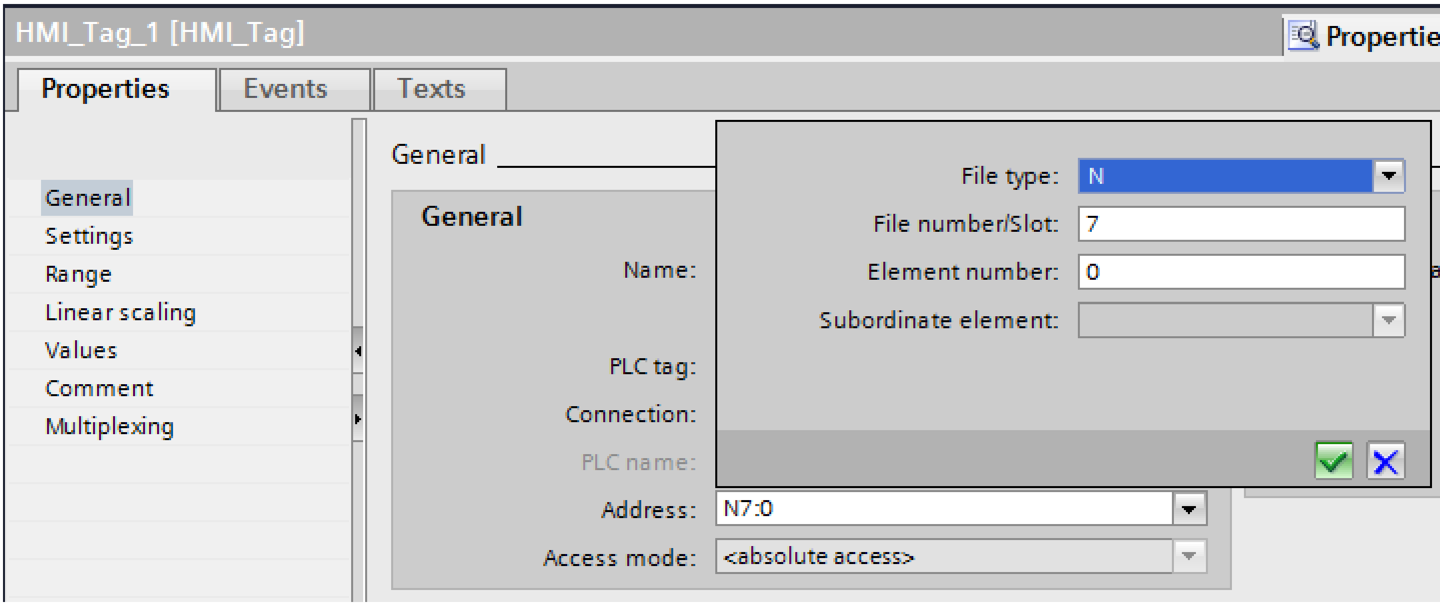
Addressing (Basic Panels)

Addressing

The addressing is entered in the following order in the Allen-Bradley DF1 communication driver:

- Operand type
- File number
- Element number

- Child element
- Bit number



The address then appears in the following format without spaces:

- File type file number : Element number . Child element
- e.g. T8:2.ACC

Operand type

You have the following options under operand type:

- I
- O
- S
- B
- T
- C
- R
- N
- A
- D only for PLC5 CPU type

File number

Select the number between two limits under file number:

- Low limit
- High limit

The limit values depend on the selected file type.

Child element

You can select a child element when you have selected one of the following data types:

- R
- C
- T

Address areas for Allen Bradley DF1 (Basic Panels)

MicroLogix

Address areas	Data types					
	Bool	Int	UInt	DInt	UDInt	Real
N	N7:0/0 - N255:255/15	N7:0 - N255:255	N7:0 - N255:255	N7:0 - N255:254	N7:0 - N255:254	N7:0 - N255:254
F	--	--	--	--	F8:0 - F255:255	--
R	R6:0.EN - R255:255.ER - R255:255.DN - R255:255.FD - R255:255.IN - R255:255.EU - R255:255.EM - R255:255.UL	R6:0.LEN - R255:255.POS	R6:0.LEN - R255:255.POS	--	--	--
C	C5:0.CU - C255:255.CD - C255:255.DN - C255:255.OV - C255:255.UN	C5:0.PRE - C255:255.ACC	C5:0.PRE - C255:255.ACC	--	--	--
T	T4:0.DN - T255:255.TT - T255:255.EN	T4:0.PRE - T255:255.ACC	T4:0.PRE - T255:255.ACC	--	--	--
B	B3:0/0 - B255:255/15	--	B3:0 - B255:255	--	--	--
S	S2:0/0 - S2:65/15	S2:0 - S2:65	--	--	--	--

Address areas	Data types					
	Bool	Int	UInt	DInt	UDInt	Real
I	I0:0/0 - I38:255/15	--	I0:0 - I38:255	--	--	--
O	O0:0/0 - O38:255/15	--	O0:0/0 - O38:255	--	--	--

SLC500

Address areas	Data types					
	Bool	Int	UInt	DInt	UDInt	Real
N	N7:0/0 - N255:255/15	N7:0 - N255:255	N7:0 - N255:255	N7:0 - N255:254	N7:0 - N255:254	N7:0 - N255:254
R	R6:0.EN - R255:255.ER - R255:255.DN - R255:255.FD - R255:255.IN - R255:255.EU - R255:255.EM - R255:255.UL	R6:0.LEN - R255:255.POS	R6:0.LEN - R255:255.POS	--	--	--
C	C5:0.CU - C255:255.CD - C255:255.DN - C255:255.OV - C255:255.UN	C5:0.PRE - C255:255.ACC	C5:0.PRE - C255:255.ACC	--	--	--
T	T4:0.DN - T255:255.TT - T255:255.EN	T4:0.PRE - T255:255.ACC	T4:0.PRE - T255:255.ACC	--	--	--
B	B3:0/0 - B255:255/15	--	B3:0 - B255:255	--	--	--
S	S2:0/0 - S2:15/15	S2:0 - S2:15	--	--	--	--
I	I0:0/0 - I38:255/15	--	I0:0 - I38:255	--	--	--
O	O0:0/0 - O38:255/15	--	O0:0 - O38:255	--	--	--

SLC501/502

Address areas	Data types					
	Bool	Int	UInt	DInt	UDInt	Real
N	N7:0/0 - N255:255/15	N7:0 - N255:255	N7:0 - N255:255	N7:0 - N255:254	N7:0 - N255:254	N7:0 - N255:254
R	R6:0.EN - R255:255.ER - R255:255.DN - R255:255.FD - R255:255.IN - R255:255.EU - R255:255.EM - R255:255.UL	R6:0.LEN - R255:255.POS	R6:0.LEN - R255:255.POS	--	--	--
C	C5:0.CU - C255:255.CD - C255:255.DN - C255:255.OV - C255:255.UN	C5:0.PRE - C255:255.ACC	C5:0.PRE - C255:255.ACC	--	--	--
T	T4:0.DN - T255:255.TT - T255:255.EN	T4:0.PRE - T255:255.ACC	T4:0.PRE - T255:255.ACC	--	--	--
B	B3:0/0 - B255:255/15	--	B3:0 - B255:255	--	--	--
S	S2:0/0 - S2:32/15	S2:0 - S2:32	--	--	--	--
I	I0:0/0 - I38:255/15	--	I0:0 - I38:255	--	--	--
O	O0:0/0 - O38:255/15	--	O0:0 - O38:255	--	--	--

PLC5

Address areas	Data types						
	Bool	Int	UInt	DInt	UDInt	Real	ASCII
N	N3:0/0 - N999:999/15	N3:0 - N999:999	N3:0 - N999:999	N3:0 - N999:999	N3:0 - N999:998	N3:0 - N999:998	--
F	--	--	--	--	--	F3:0 - F999:999	--
A	--	--	--	--	--	--	A3:0 - A999:999

Address areas	Data types						
	Bool	Int	UInt	DInt	UDInt	Real	ASCII
R	R3:0.EN - R999:999.ER - R999:999.DN - R999:999.FD - R999:999.IN - R999:999.EU - R999:999.EM - R999:999.UL	R3:0.LEN - R999:999.POS	R3:0.LEN - R999:999.POS	--	--	--	--
C	C3:0.CU - C999:999.CD - C999:999.DN - C999:999.OV - C999:999.UN	C3:0.PRE - C999:999.ACC	C3:0.PRE - C999:999.ACC	--	--	--	--
T	T3:0.DN - T999:999.TT - T999:999.EN	T3:0.PRE - T999:999.ACC	T3:0.PRE - T999:999.ACC	--	--	--	--
B	B3:0/0 - B999:999/15	--	B3:0 - B999:999	--	--	--	--
S	S2:0/0 - S2:127/15	S2:0 - S2:127	--	--	--	--	--
I	I1:0/0 - I1:277/17	--	I1:0 - I1:277	--	--	--	--
O	O0:0/0 - O0:277/17	--	O0:0 - O0:277	--	--	--	--
D	D3:0/0 - D999:999/15	D3:0 - D999:999	D3:0 - D999:999	--	D3:0 - D999:998	--	--

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance. Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.
- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Data exchange (Basic Panels)

Area pointers for Allen-Bradley (Basic Panels)

Area pointers for connections using an Allen-Bradley communication driver

You use an area pointer to access a data area in the PLC.

For more detailed information on area pointers and their configuration, refer to Section "Data exchange using area pointers".

Distinctive features for connections via Allen-Bradley EtherNet/IP

You can configure the following area pointers

Area pointer	Allen-Bradley EtherNet/IP	Allen-Bradley DF1
Screen number	Yes	Yes
Date/time	Yes	Yes
Date/time PLC	Yes	Yes
Coordination	Yes	Yes
Project ID	Yes	Yes
Job mailbox	Yes	Yes
Data record	Yes	Yes

Restrictions Allen-Bradley Ethernet/IP

The following restrictions apply for configuring area pointers.

CPU type	Data types	File types
ControlLogix,CompactLogix	Int, UInt	--
SLC, MicroLogix	Int, UInt	N, B

Restrictions Allen-Bradley DF1

The following restrictions apply for configuring area pointers.

CPU type	Data types	File types
MicroLogix	--	N, O, I, B
SLC50x	--	N, O, I, B
PLC5	--	N, O, I, B

See also

Data exchange using area pointers (Page 451)

Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configuring alarms for non-integrated connections

Several steps are necessary to configure alarms such as warnings, error messages and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Restrictions

Only tags whose "File type" is "N", "O", "I", "S" and "B" are allowed for use as a "trigger tag" for discrete alarms. These tags are only valid for the data types "Int" and "UInt".

Data types

For connections with an Allen-Bradley communication driver, the following data types are supported:

Communication drivers	PLC	Permitted data types	
		Discrete alarms	Analog alarms
Allen-Bradley DF1	SLC500, SLC501, SLC502, SLC503, SLC504, SLC505, PLC5, MicroLogix	Int, UInt	Int, UInt, Long, ULong, Real
Allen-Bradley EtherNet/IP	ControlLogix, Compact-Logix, SLC, Micrologix	Int, UInt	SInt, USInt, Int, UInt, DInt, UDInt, Real

How the bit positions are counted

For connections with an Allen-Bradley communication driver, the following counting method applies:

How the bit positions are counted	Left byte								Right byte							
In Allen-Bradley PLCs	15							8	7							0
In WinCC you configure:	15							8	7							0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

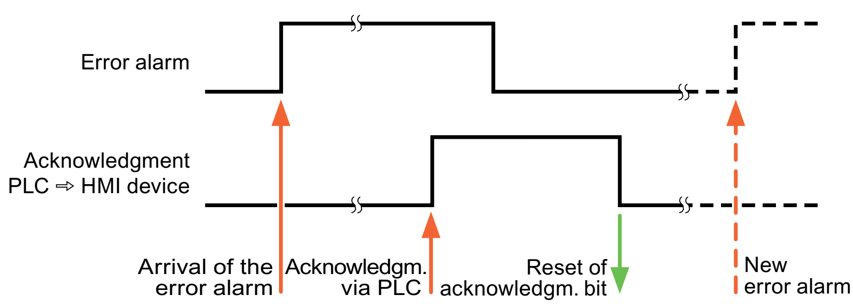
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

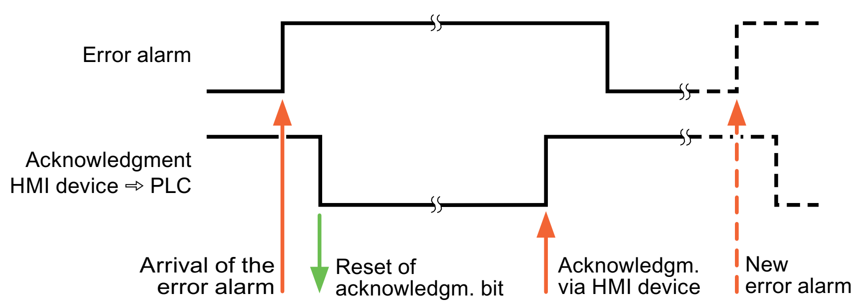
If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.13.3.2 Mitsubishi (Basic Panels)

Mitsubishi communication drivers (Basic Panels)

Introduction

This section describes the communication between an HMI device and PLCs that use Mitsubishi communication drivers.

The following communication drivers are supported:

- Mitsubishi MC TCP/IP
- Mitsubishi FX

Data exchange

Data is exchanged by means of tags or area pointers.

- Tags
  The PLC and the HMI device use process values for data exchange. You create tags in the configuration that point to addresses in the PLC. The HMI device reads the value from the defined address, and then displays it. The operator may also enter values on the HMI device, which are then written to the address in the PLC.
- Area pointers
  Area pointers are used to exchange specific data and are only set up when these data are used.

Mitsubishi MC TCP/IP (Basic Panels)

Configuring a connection via Mitsubishi MC TCP/IP (Basic Panels)

Introduction

You configure a connection to a PLC with a Mitsubishi MC TCPI/IP communication driver in the "Connections" editor of the HMI device.

The Ethernet interfaces are named differently depending on the HMI device.

Example: PROFINET interface corresponds to the Ethernet interface

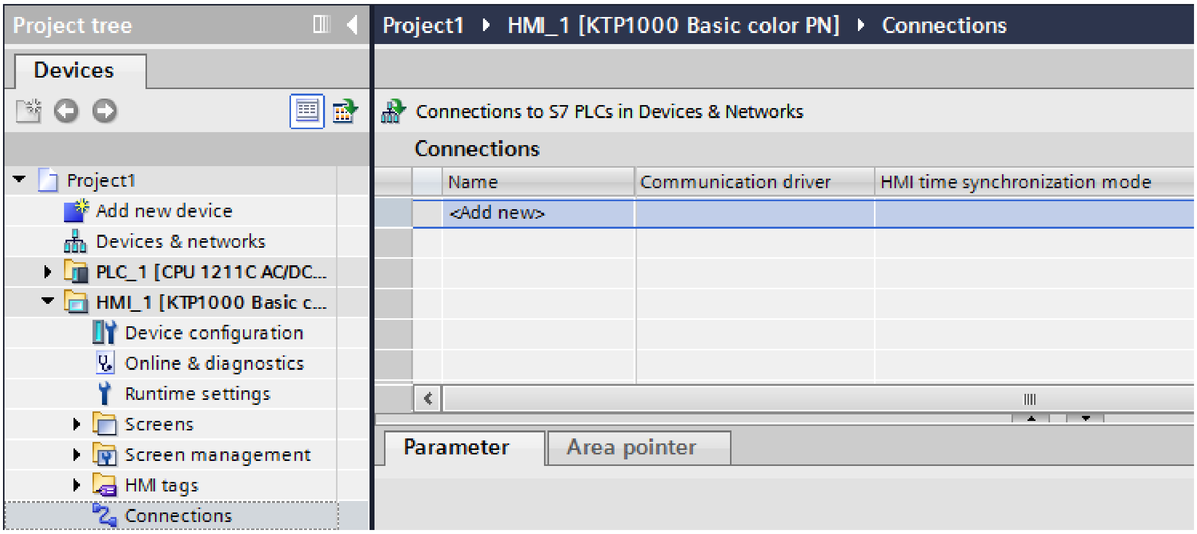
Requirements

- A project is open.
- An HMI device has been created.

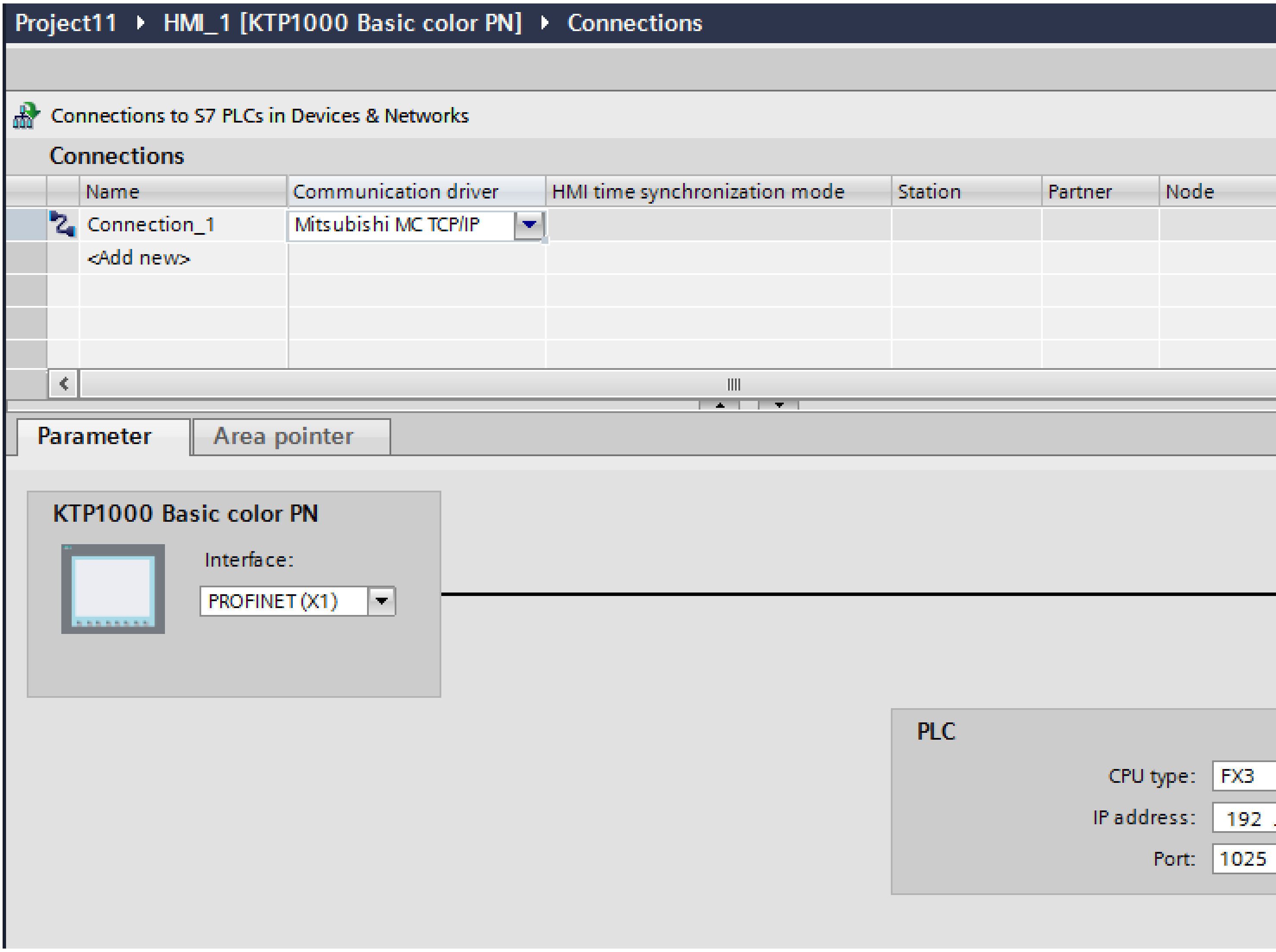
Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.

3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "Mitsubishi MC TCPI/IP" driver.



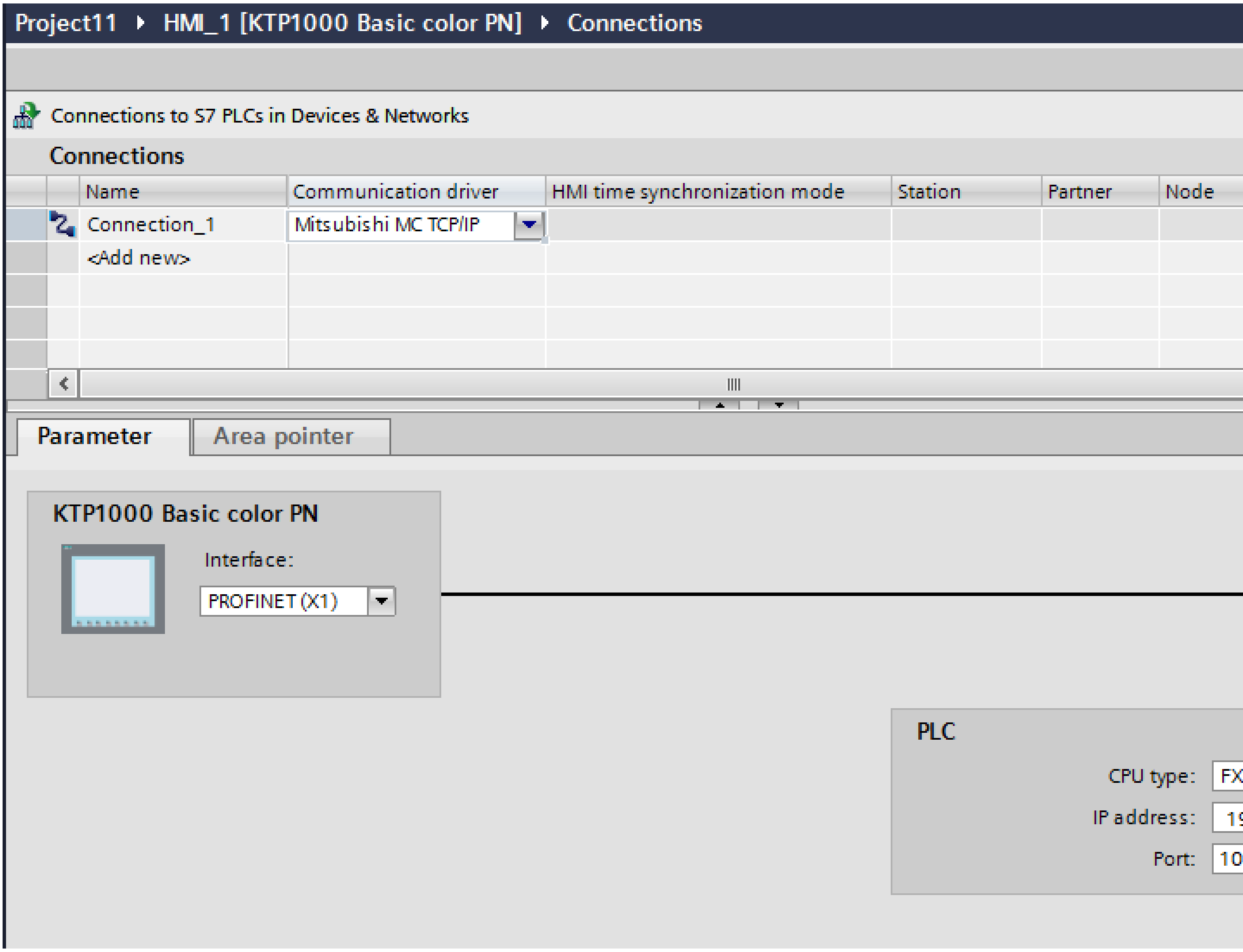
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".

Parameters for the connection (Mitsubishi MC TCP/IP) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device" and "PLC" areas are available for assigning parameters according to the interface used.



Parameters for the HMI device

You can select only one interface for the HMI device in the Inspector window under "Parameters". Depending on the HMI device, there are several interfaces available.

If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC. The IP address is transferred to the HMI device during project transfer.

Note

The IP address in the control panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

The IP address already set up in the control panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

To set up the IP address of the HMI device:

1. Click on the HMI device.
2. Open the "Device configuration" editor.
3. Click the Ethernet interface.
4. Assign the IP address in the inspector window under:
   "General > PROFINET interface > Ethernet addresses"

Parameters for the PLC

- CPU type
  For "CPU type", you set the type of PLC to which the HMI device is connected.
  The following settings are possible:
  –FX3
  –Q
  If you select the FX3 CPU type, the Mitsubishi MC protocol "1E" is used and "3E" for the "Q" CPU type.
  The "Binary code" protocol variant is always used.

  Note

  If the CPU type is changed for a configured connection, tags with the following properties must be revised:

  - Operands that do not exist for the new CPU type, such as "W", "B", "F".
  - Inputs and outputs with different addressing (hexadecimal/octal)
  - Addresses greater than the valid address area of the new CPU type

- IP address
  Set the IP address or host name of the Ethernet/IP module of the PLC. Only the IP address can be used on a Basic Panel.
- Port
  Set the port number of the module of the PLC.

Connecting HMI device to PLC (Basic Panels)

Connections via Mitsubishi MC TCP/IP (Basic Panels)

Connection

The HMI device can be connected to the Mitsubishi PLC using the following components:

- Existing Ethernet network that also contains the PLCs
- Cross-over Ethernet cable connected directly to the Ethernet interface of the CPU or the communication module

The connection of the HMI device to a Mitsubishi PLC is limited primarily to the physical connection of the HMI device. Special blocks for the connection are not required in the PLC.

Connect the HMI device to one or several Q-series and/or FX3 PLCs. Connect the HMI device via the following interfaces:

- Communication interface OnBoard
- Approved communication module suitable for the PLC

> **Note**
>
> **Timeout response with TCP/IP (Ethernet)**
>
> Due to the use of the TCP/IP protocol, the breakdown of a connection is detected at the earliest after approximately one minute. Communication failure cannot be reliably detected if no tags are requested, for example, no output tags in the current screen.
>
> Configure area pointer coordination for each PLC. This setting ensures that a communication failure is recognized after approximately two minutes, even in the aforementioned scenario.

Communication types (Basic Panels)

Approved communication types

- Only applies for Mitsubishi FX(PG protocol):
  The point-to-point connection from a HMI device to an approved Mitsubishi FX–CPU via Mitsubishi FX is system-tested and approved by Siemens AG.
- Only applies for Mitsubishi MC TCP/IP:
  The following communication types are system-tested and approved:
  - Point-to-point connection to the approved PLCs
  - Multipoint connection from a HMI device with up to 4 PLCs with the respectively approved PLCs. CPU types (FX3 and Q) can be mixed.

    > **Note**
    >
    > The HMI device is a client and the PLC must operate as a server.

Connectable PLCs

Connections can be implemented for the following Mitsubishi PLCs:

PLC	Mitsubishi FX (PG protocol)	Mitsubishi MC TCP/IP
MELSEC FX1n, FX2n	Yes	No
MELSEC FX3U, FX3UC, FX3G with communication module FX3U-ENET	No	Yes
MELSEC System Q • Q-series with the communication module QJ71E71-100 • QnUDEH CPU with Ethernet interface onboard	No	Yes

Parameterization of the communications modules (Basic Panels)

FX3 PLCs (Basic Panels)

Procedure

1. Start the FX-Configurator.
2. Select the module.
3. Assign the following settings in the "Operational settings" dialog:
   - Communication data code:
     Binary code
   - Initial timing:
     Always wait for OPEN
   - IP address:
     IP address
   - Send frame setting:
     Ethernet(V2.0)
   - TCP Existence confirmation setting:
     Use the Ping

4. Assign the following settings in the "Open Settings" dialog:
   – Protocol:
     TCP
   – Open system:
     Unpassive
   – Fixed buffer:
     Receive
   – Fixed buffer communication procedure:
     Procedure exist(MC)
   – Pairing open
     Disable
   – Existence confirmation
     No confirm
   – Host station Port No. (DEC)
     Port number

     Note

     The port number chosen in the communication module must match the port number in WinCC. A connection with a port number must be assigned for each connected HMI device.

     You must specify port numbers in decimal values.
5. Confirm the default settings of the other dialog boxes.

The network no. and station no. parameters are not relevant for the connection and can be chosen as required.

Q PLCs (Basic Panels)

Procedure

1. Click "Edit network parameters".
2. Select the network type:
   – Ethernet
     The network number and the group / station number are not evaluated and can be freely assigned

3. Assign the following settings in the "Operational settings" dialog:
   - Communication data code:
     Binary code
   - Initial timing:
     Always wait for OPEN
   - IP address:
     IP address
   - Send frame setting:
     Ethernet(V2.0)
   - Enable write operations during RUN
4. Assign the following settings in the "Open settings" dialog:
   - Protocol:
     TCP
   - Open system:
     Unpassive
   - Pairing open
     Disable
   - Existence confirmation
     No confirm
   - Host station Port No. (HEX)
     Port-Nummer

Note

The port number chosen in the communication module must match the port number in WinCC. A connection with a port number must be assigned for each connected HMI device.

You must specify port numbers in hexadecimal values.

Internal Ethernet port of the Q0xUDEH CPU (Basic Panels)

Procedure

1. Assign the following settings in the "Internal Ethernet Port" dialog:
   – IP address:
     IP address
   – Communication data code:
     Binary code
   – Enable online changes
2. Assign the following settings in the "Open settings" dialog:
   – Protocol:
     TCP
   – Open system:
     MC-Protocol
   – Host station Port No. (HEX)
     Port number

   Note

   The port number chosen in the communication module must match the port number in WinCC. A connection with a port number must be assigned for each connected HMI device.

Performance features of communication (Basic Panels)

Permitted data types for Mitsubishi MC TCPI/IP (Basic Panels)

Permitted data types

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Operand type	Length
4-bit block	M, X, Y, B, F	1 byte
8-bit block	M, X, Y B, F	1 byte
12-bit block	M, X, Y B, F	2 bytes
16-bit block	M, X, Y B, F	2 bytes
20-bit block	M, X, Y B, F	4 bytes
24-bit block	M, X, Y B, F	4 bytes
28-bit block	M, X, Y B, F	4 bytes
32-bit block	M, X, Y B, F	4 bytes
Bool	M, D, X, Y B, F	1-bit
DInt	D, W	4 bytes

Data type	Operand type	Length
DWord	D, C, W	4 bytes
Int	D, W	2 bytes
Real 1)	D, W	4 bytes
String 1)	D	1 to 80 characters
Word	D, T, C, W	2 bytes

1) The "String" and "Real" data types are not available for all CPUs.

2) Operand types B, F and W are only available for CPU type "Q".

Note

Note the following for write accesses:

Tags can only be written if "Enable online changes" or "Enable write operations during RUN" was selected when parameterizing the Mitsubishi communication modules.

For data type "Bool" in operand type "D", the entire word is written back to the PLC following a change to the specified bit. There is no check to determine whether any other bits in the word have changed. As a result, the PLC only has read access to the specified word.

Note

Array elements in I/O fields cannot be used in communication with a Mitsubishi PLC.

Supported CPU types for Mitsubishi MC TCP/IP (Basic Panels)

CPU types

The following CPU types are supported for configuring the Mitsubishi MC TCP/IP communication driver.

- FX3 series
  - FX 3G / FX 3G with communication modul FX3U-ENET
  - FX 3U / FX 3U with communication modul FX3U-ENET
  - FX 3UC / FX 3UC with communication modul FX3U-ENET
- Q series
  - Q-Series with QJ71E71-100 communication module
- iQ series / QnUD
  - QnUDEHCPU with built in ethernet module

Addresses for Mitsubishi MC TCP/IP (Basic Panels)

Address areas for connections via Mitsubishi MC TCP/IP

The address area boundaries differ for the different series; refer to the Mitsubishi Computerlink manuals for this information.

Examples of address area boundaries dependent on the CPU and communication format:

Name	Operand type	Max. address FX3	Max. address Q-Series
Output/Input	Y/X	Octal X/Y 0 - 777	HEX X/Y 0 - 7FF
Bit memory	M	M0 - M3071 and M8000 - M8255	M/L/S 0 - 8191
Data register	D	D0 - 7999 D8000 - D8255	D0 - 8191 D9000 - D9255 becomes SD1000 - SD1255
Counter	C	C0 - 255	C0 - 1023
Timer	T	T0 - 255	T0 - 2047
Link register	W	--	Hex: W0 - FFF
Link flag	B	--	Hex: B0 - FFF
Error flag	F	--	F0 - 2047

Address areas for Mitsubishi MC TCP/IP (Basic Panels)

FX3

Address areas	Data types													
	Bool	Int	Word	DInt	DWord	Real	String	4-bit block	8-bit block	12-bit block	16-bit block	20-bit block	24-bit block	28-bit block	32-bit block
M	M0 - M9999	--	--	--	--	--	--	M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
D	D0.0 - D9999.15	D0 - D9999	D0 - D9999	D0 - D9999	D0 - D9999	D0 - D9999	D0 - D9999	--	--	--	--	--	--	--	--
T	--	--	T0 - T999	--	--	--	--	--	--	--	--	--	--	--	--
C	--	C0 - C999	C0 - C999	C0 - C998	C0 - C998	--	--	--	--	--	--	--	--	--	--
X	X0 - X777	--	--	--	--	--	--	X0 - X774	X0 - X770	X0 - X764	X0 - X760	X0 - X754	X0 - X750	X0 - X744	X0 - X740
Y	Y0 - Y777	--	--	--	--	--	--	Y0 - Y774	Y0 - Y770	Y0 - Y764	Y0 - Y760	Y0 - Y754	Y0 - Y750	Y0 - Y744	Y0 - Y740

Q

Address areas	Data types														
	Bool	Int	Word	DInt	DWord	Real	String	4-bit block	8-bit block	12-bit block	16-bit block	20-bit block	24-bit block	28-bit block	32-bit block
M	M0 - M9999	--	--	--	--	--	--	M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
F	F0 - F9999	--	--	--	--	--	--	F0 - F9996	F0 - F9992	F0 - F9988	F0 - F9984	F0 - F9980	F0 - F9976	F0 - F9972	F0 - F9968
B	B0 - BFFFF	--	--	--	--	--	--	B0 - BFFFC	B0 - BFFF8	B0 - BFFF4	B0 - BFFF0	B0 - BFFEC	B0 - BFFE8	B0 - BFFE4	B0 - BFFE0
D	D0.0 - D65534.15	D0 - D65534	D0 - D65534	D0 - D65533	D0 - D65533	D0 - D65533	D0 - D65534	--	--	--	--	--	--	--	--
T	--	--	T0 - T2047	--	--	--	--	--	--	--	--	--	--	--	--
C		C0 - C2047	C0 - C2047	C0 - C2046	C0 - C2046	--	--	--	--	--	--	--	--	--	--
W	--	W0 - WFFFF	W0 - WFFFF	W0 - WFFFE	W0 - WFFFE	W0 - WFFFE	--	--	--	--	--	--	--	--	--
X	X0 - XFFFF	--	--	--	--	--	--	X0 - XFFFC	X0 - XFFF8	X0 - XFFF4	X0 - XFFF0	X0 - XFFEC	X0 - XFFE8	X0 - XFFE4	X0 - XFFE0
Y	Y0 - YFFFF	--	--	--	--	--	--	Y0 - YFFFC	Y0 - YFFF8	Y0 - YFFF4	Y0 - YFFF0	Y0 - YF-FEC	Y0 - YFFE8	Y0 - YFFE4	Y0 - YFFE0

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance. Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.
- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Mitsubishi FX (Basic Panels)

Configuring a connection via Mitsubishi FX (Basic Panels)

Introduction

You configure a connection to a PLC with a Mitsubishi FX communication driver in the "Connections" editor of the HMI device.

The Mitsubishi FX protocol is also referred to as the Mitsubishi PG protocol.

The interfaces are named differently depending on the HMI device.

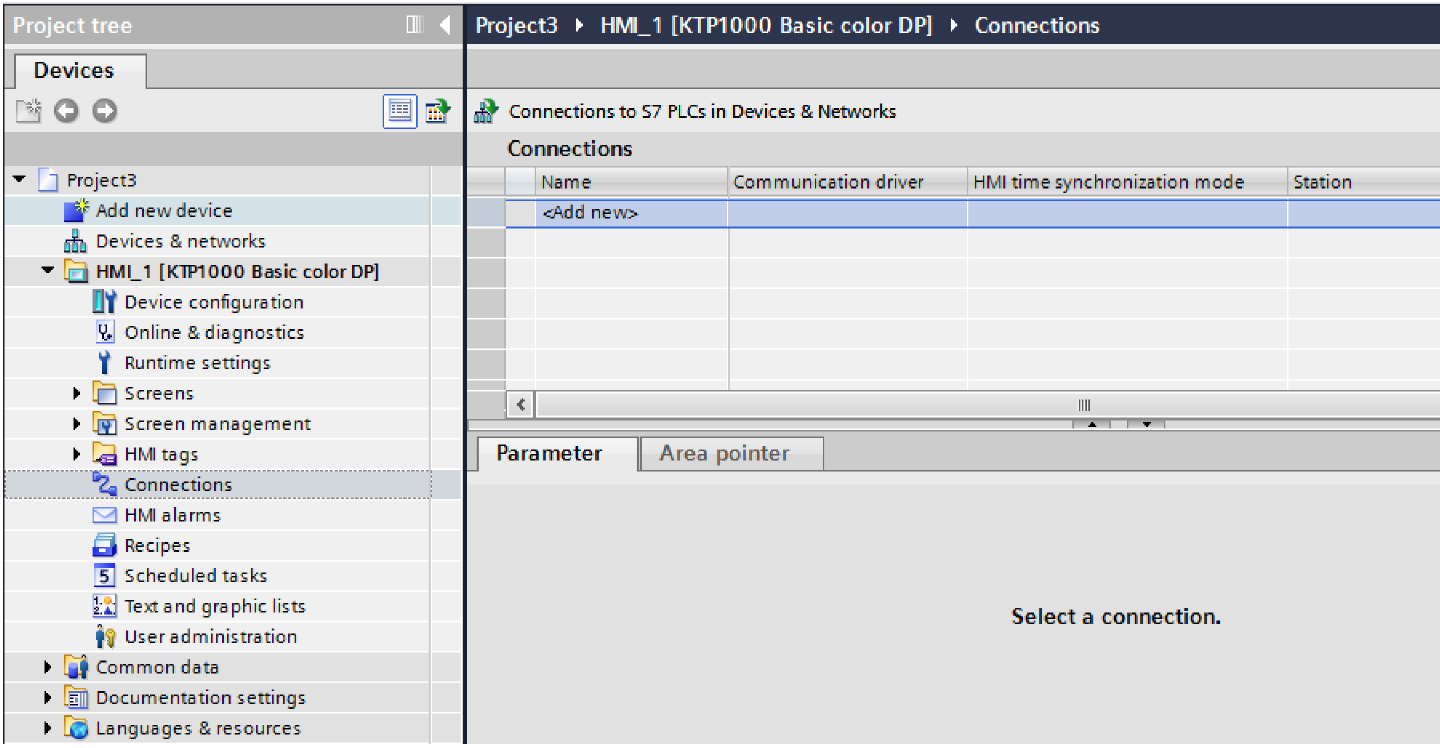
Requirements

- A project is open.
- An HMI device has been created.

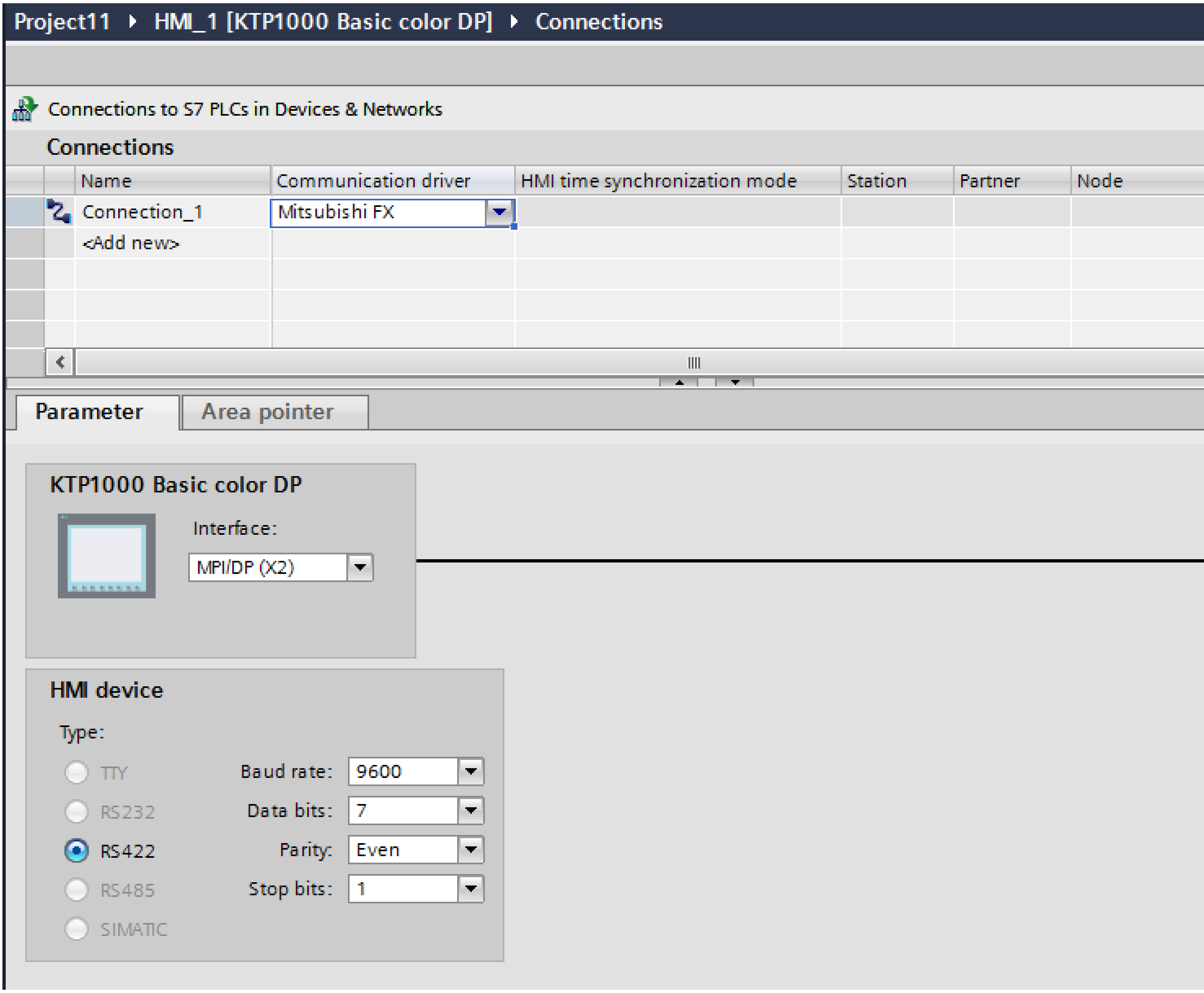
Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.

3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "Mitsubishi FX" driver.



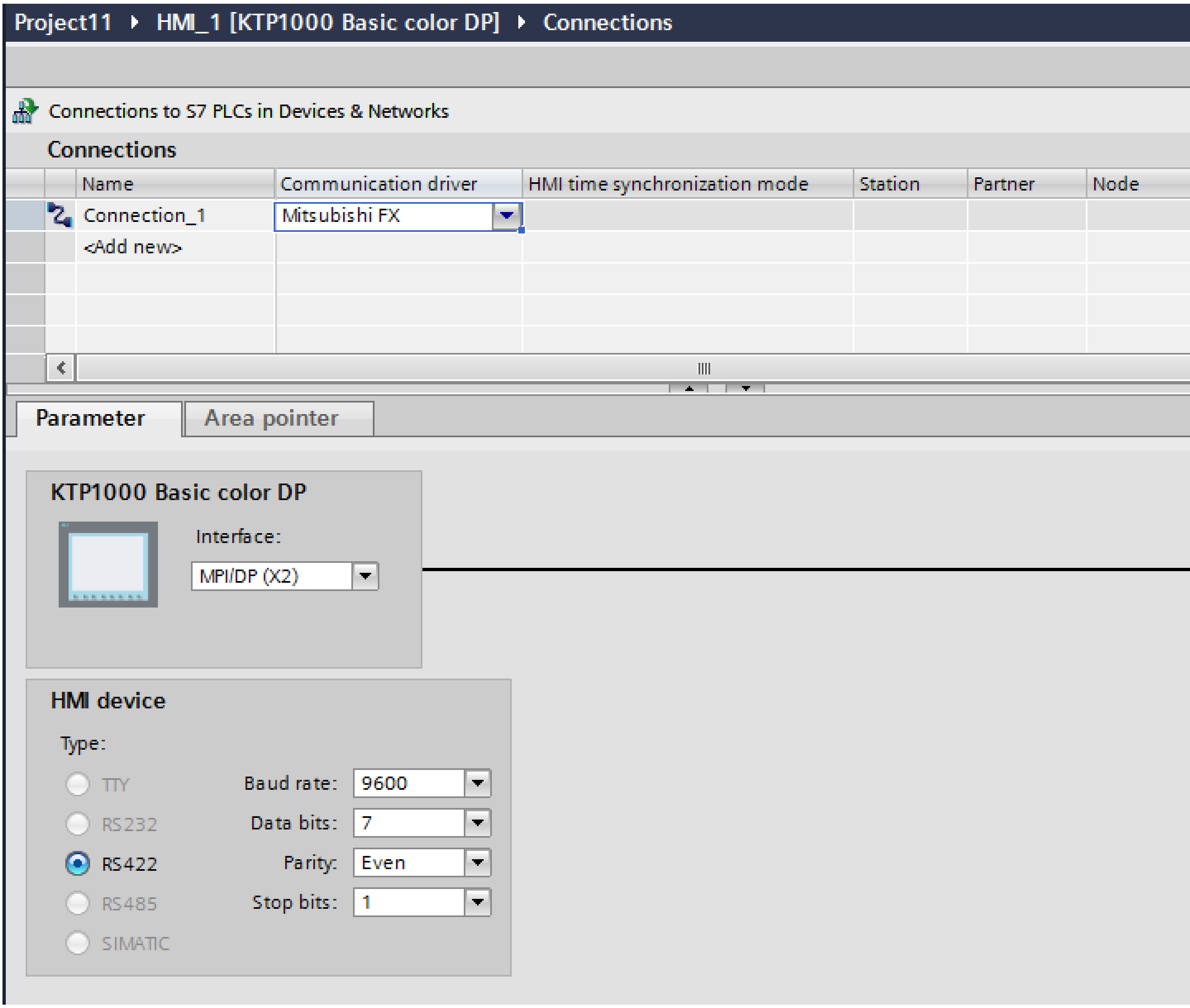
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".

Parameters for the connection (Mitsubishi FX) (Basic Panels)

Parameters to be set

To assign the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device", "Network", and "PLC" areas are available for assigning parameters according to the interface used.



Parameters for the HMI device

You can select an interface for the HMI device in the Inspector window under "Parameters". Depending on the HMI device, there are several interfaces available.

- "Type"
  Specifies the physical connection used.

  Note

  If you use the IF1B interface, you must switch over the RS422 receive data and the RTS signal additionally by 4 DIL-switches on the back of the HMI device.

Parameters for the PLC

- Baud rate: For "Baud rate", select the transmission speed between the HMI device and PLC. Select the baud rate "9600".
- Data bits: Under "Data bits", select "7 bits".
- Parity: Under "Parity", select "Even".
- Stop bits: Under "Stop bits", select "1 bit".

Connecting HMI device to PLC (Basic Panels)

Communication types (Basic Panels)

Approved communication types

- Only applies for Mitsubishi FX(PG protocol):
  The point-to-point connection from a HMI device to an approved Mitsubishi FX–CPU via Mitsubishi FX (PG protocol := protocol for access to the program and memory elements of the FX series PC CPU version V1.21 and after) is system-tested and approved by Siemens AG.
- Only applies for Mitsubishi MC TCP/IP:
  The following communication types are system-tested and approved:
  – Point-to-point connection to the approved PLCs
  – Multipoint connection from a HMI device with up to 4 PLCs with the respectively approved PLCs. CPU types (FX3 and Q) can be mixed.

    Note

    The HMI device is a client and the PLC must operate as a server.

Connectable PLCs

Connections can be implemented for the following Mitsubishi PLCs:

PLC	Mitsubishi FX (PG protocol)	Mitsubishi MC TCP/IP
MELSEC FX1n, FX2n	Yes	No
MELSEC FX3U, FX3UC, FX3G with communication module FX3U-ENET	No	Yes
MELSEC System Q • Q-series with the communication module QJ71E71-100 • QnUDEH CPU with Ethernet interface onboard	No	Yes

Connections via Mitsubishi FX (Basic Panels)

Connection

Connect the HMI device to the programming interface of the CPU (RS 422) (see documentation of the PLC).

The connection between the HMI device and the Mitsubishi PLC is basically restricted to setting the interface parameters. Special blocks for the connection are not required in the PLC.

Connecting cable

The following connecting cables are available to connect the HMI device to the PLC.

Interface to HMI device or adapter	Mitsubishi Electric PLC via FX protocol
	FX1n, Fx2n, Mini DIN, 8–pin
RS 232, 9-pin	Mitsubishi SC-09 [1)]
RS 422, 9-pin	Connecting cable RS422-2P

1) Since the Mitsubishi PLCs communicate via RS 422 as a standard, the Mitsubishi programming cable SC–09 with integrated RS 422/RS 232 adaptor is necessary for connecting a HMI device via RS 232.

Note

Applies only to RS 232:

Cable length is restricted to 0.32 m.

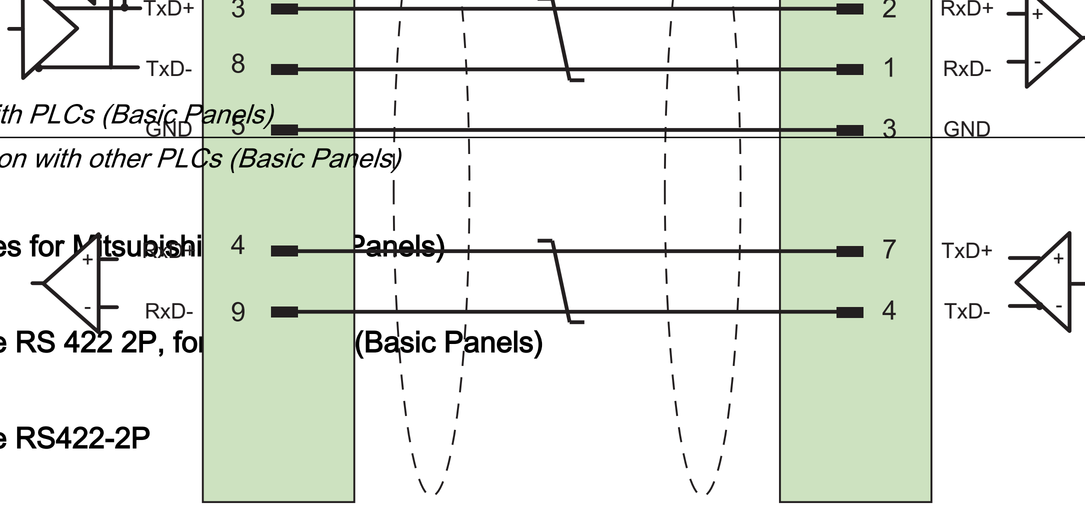
Refer to the relevant device manual to determine which HMI device interface is to be used.

The cable pin assignments can be found in Section "Connecting cables for Mitsubishi FX".

Connecting cables for Mitsubishi (Basic Panels)

Connecting cable RS 422 2P, for Mitsubishi (Basic Panels)

Connecting cable RS422-2P



Shield with large-area contact to housing at both ends
Cable: 3 x 2 x 0.14 mm², shielded,
max. length 500 m

Performance features of communication (Basic Panels)

Permitted data types for Mitsubishi FX (Basic Panels)

Permitted data types

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Operand type	Length
4-bit block	M, X, Y	1 byte
8-bit block	M, X, Y	1 byte
12-bit block	M, X, Y	2 bytes
16-bit block	M, X, Y	2 bytes
20-bit block	M, X, Y	4 bytes
24-bit block	M, X, Y	4 bytes
28-bit block	M, X, Y	4 bytes
32-bit block	M, X, Y	4 bytes
Bool	D, M, X, Y	1-bit
DWord	D, C-32 bit	4 bytes
Real	D	4 bytes
String	D	1 to 50 characters
Word	D, T, C-16 bit	2 bytes

Note

Note the following for write accesses:

For data type "Bool" in operand type "D", the entire word is written back to the PLC following a change to the specified bit. There is no check to determine whether any other bits in the word have changed. As a result, the PLC only has read access to the specified word.

Note

Array elements in I/O fields cannot be used in communication with a Mitsubishi PLC.

Supported CPU types for Mitsubishi FX (Basic Panels)

CPU types

The following CPU types are supported for configuring the Mitsubishi FX communication driver.

- FX1 series
  - FX1n
- FX2 series
  - FX2n

Address areas for Mitsubishi FX (Basic Panels)

FX1n and FX2n

Address areas	Data types												
	Bool	Word	DWord	Real	String	4-bit block	8-bit block	12-bit block	16-bit block	20-bit block	24-bit block	28-bit block	32-bit block
M	M0 - M9999					M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
D	D0.0 - D999.15	D0 - D999	D0 - D998	D0 - D998	D0 - D998								
T		T0 - T255											
C-16-Bit		C-16-Bit 0 - C-16-Bit 199											
C-32-Bit			C-32-Bit 200 - C-32-Bit 255										
X	X0 - X255					X0 - X252	X0 - X248	X0 - X244	X0 - X240	X0 - X236	X0 - X232	X0 - X228	X0 - X224
Y	Y0 - X255					Y0 - Y252	Y0 - Y248	Y0 - Y244	Y0 - Y240	Y0 - Y236	Y0 - Y232	Y0 - Y228	Y0 - Y224

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance. Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.
- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Data exchange (Basic Panels)

Area pointers for Mitsubishi (Basic Panels)

Area pointers for connections via Mitsubishi communication drivers

You use an area pointer to access a data area in the PLC.

For more detailed information on area pointers and their configuration, refer to Section "Data exchange using area pointers".

Special considerations for connections via Mitsubishi communication drivers

You can configure the following area pointers

Area pointers	Mitsubishi MC TCP/IP	Mitsubishi FX
Screen number	Yes	Yes
Date/time	Yes	Yes
Date/time PLC	Yes	Yes
Coordination	Yes	Yes
Project ID	Yes	Yes
Job mailbox	Yes	Yes
Data record	Yes	Yes

Restrictions for Mitsubishi FX and MC TCP/IP

The following restrictions apply for configuring area pointers.

CPU type	Data types	Operand type
FX/FX3	Int, Word	D
Q	Int, Word	D

See also

Data exchange using area pointers (Page 451)

Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configuring alarms for non-integrated connections

Several steps are necessary to configure alarms such as warnings, error messages and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a Mitsubishi communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
FX1n, FX2n, FX3 series, Q-Series, iQ-Series	Word, Int [1]	4-bit block, 8-bit block, 12-bit block, 16-bit block, 20-bit block, 24-bit block, 28-bit block, 32-bit block, Word, DWord, Int [1], DInt [1], Real,
[1] Not for Mitsubishi FX communication driver		

How the bit positions are counted

For connections with a Mitsubishi communication driver, the following counting method applies:

How the bit positions are counted	Left byte								Right byte							
In Mitsubishi PLCs	15							8	7							0
In WinCC you configure:	15							8	7							0

Restrictions on alarms

- Mitsubishi MC TCP/IP
  Only tags of operand type "D" and data types "Word" and "Int" are permitted as trigger tags for discrete alarms. You can use array tags (operand type: "D"; data types: "ARRAY [x..y] of Word" or "ARRAY [x..y] of Int") for discrete alarms.
- Mitsubishi FX
  Only tags of operand type "D" and data type "Word" are permitted as trigger tags for discrete alarms. You can use array tags (operand type "D"; data type "ARRAY [x..y] of Word") for discrete alarms."

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

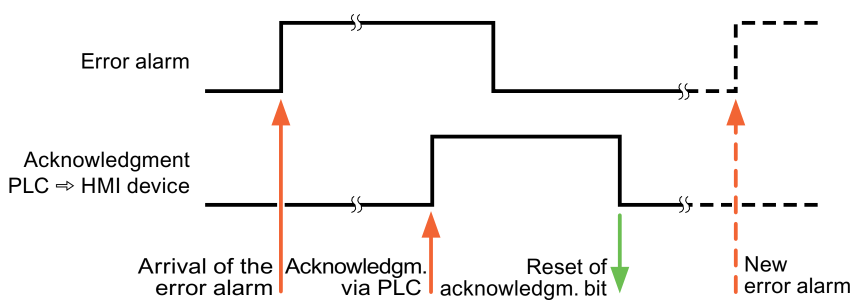
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment

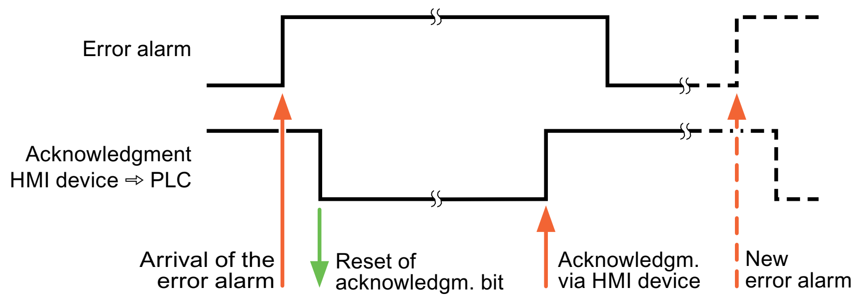
tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.13.3.3 Modicon Modbus (Basic Panels)

Modicon Modbus communication drivers (Basic Panels)

Introduction

This section describes the communication between an HMI device and PLCs that use Modicon Modbus communication drivers.

The following communication drivers are supported:

- Modicon Modbus TCP/IP
- Modicon Modbus RTU

Data exchange

Data is exchanged by means of tags or area pointers.

- Tags
  The PLC and the HMI device use process values for data exchange. You create tags in the configuration that point to addresses in the PLC. The HMI device reads the value from the defined address, and then displays it. The operator may also enter values on the HMI device, which are then written to the address in the PLC.
- Area pointers
  Area pointers are used to exchange specific data and are only set up when these data are used.

Modicon Modbus TCP/IP (Basic Panels)

Configuring a connection via Modicon Modbus TCP/IP (Basic Panels)

Introduction

You configure a connection to one of the PLCs with Modicon Modbus TCP/IP communication driver in the "Connections" editor of the HMI device.

The Ethernet interfaces are named differently depending on the HMI device.

Example: PROFINET interface corresponds to the Ethernet interface

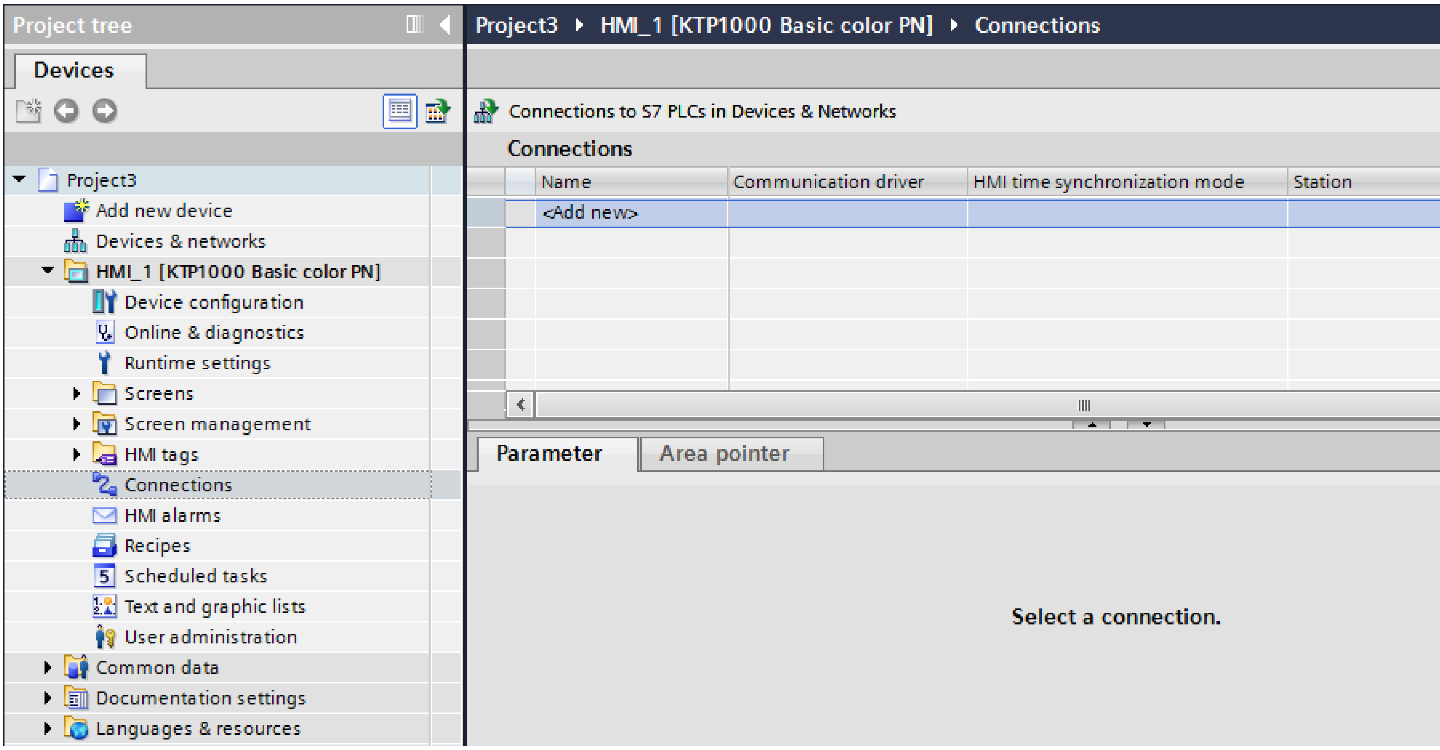
Requirements

- A project is open.
- An HMI device has been created.

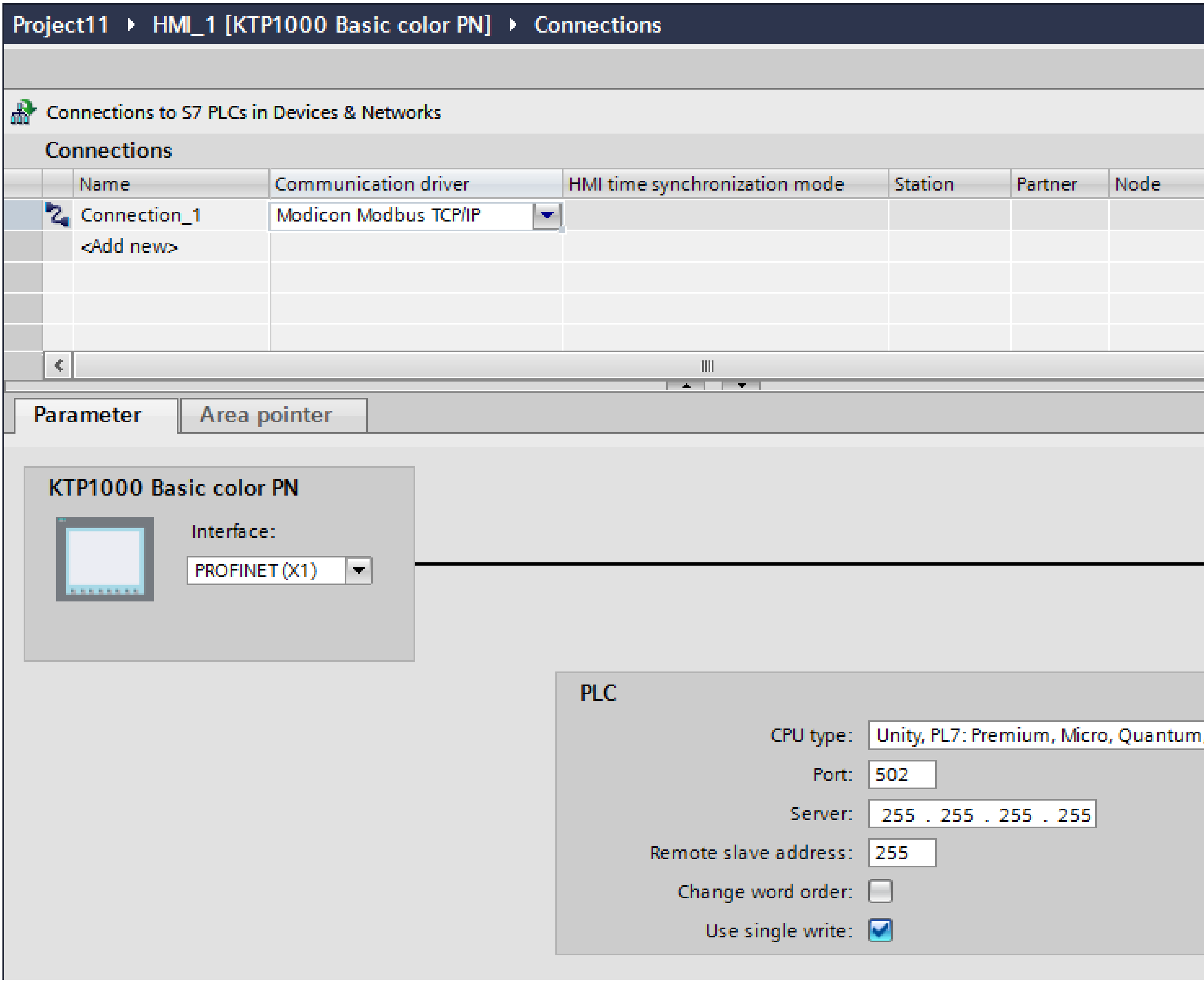
Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.

3. Double-click "<Add>" in the "Connections" editor.



4. Select the "Modicon Modbus TCP" driver in the "Communication driver" column.



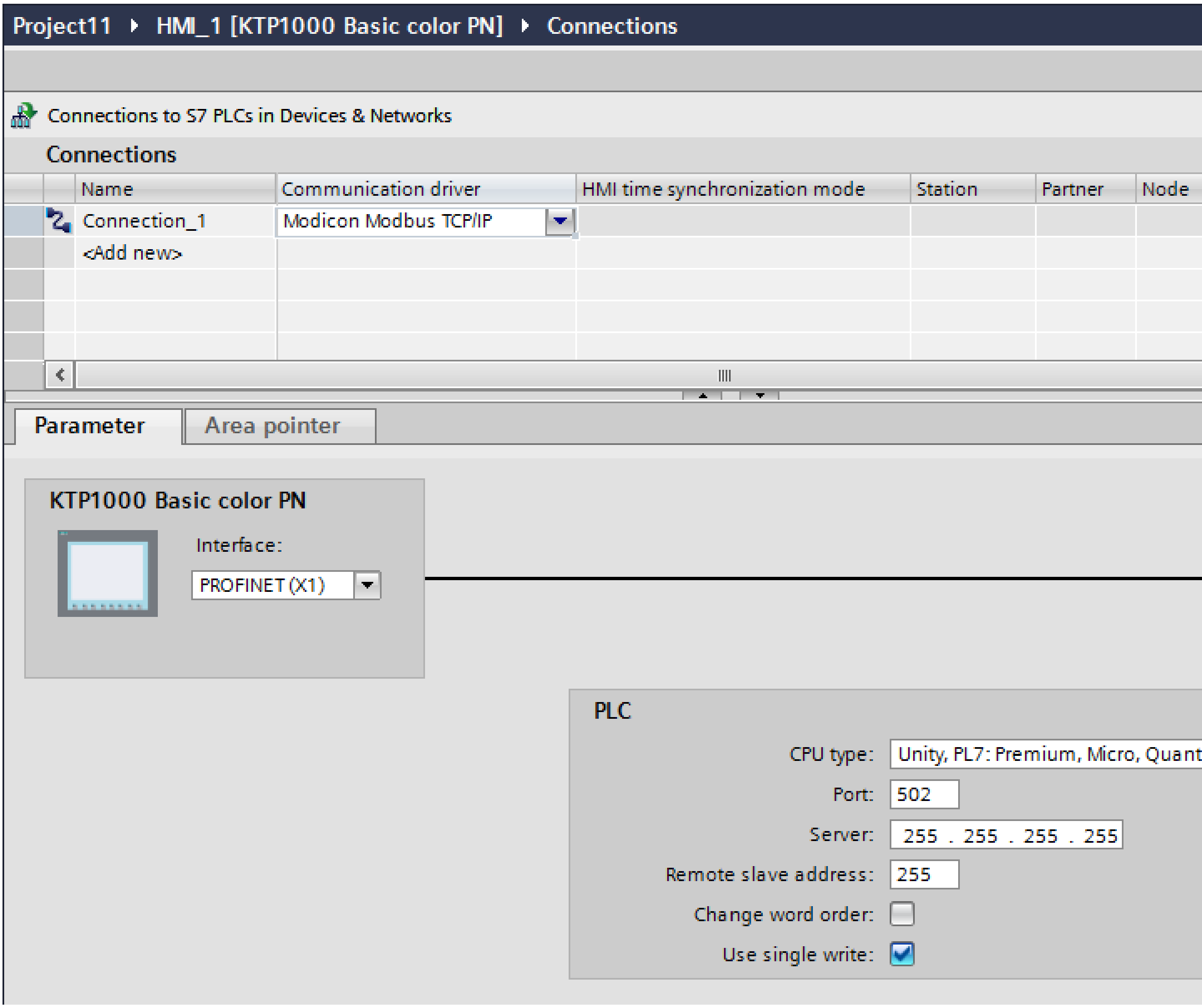
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".

Parameters for the connection (Modicon Modbus TCP/IP) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device" and "PLC" areas are available for assigning parameters according to the interface used.



Parameters for the HMI device

You can select only one interface for the HMI device in the Inspector window under "Parameters". Depending on the HMI device, there are several interfaces available.

If you are directly connected to the HMI device during configuration, you can set up the IP address of the HMI device in WinCC. The IP address is transferred to the HMI device during project transfer.

Note

The IP address in the control panel will be overwritten upon subsequent loading if you have already set up the IP address in the HMI device control panel.

The IP address already set up in the control panel will be retained upon subsequent loading if you activate "Set IP address using a different method".

To set up the IP address of the HMI device:

1. Click the HMI device.
2. Open the "Device configuration" editor.
3. Click the Ethernet interface.
4. Assign the IP address in the inspector window under:
   "General > PROFINET interface > Ethernet addresses"

Parameters for the PLC

- "CPU type"
  For "CPU type", you set the Modicon PLC to which the HMI device is connected.
- "Port"
  For "Port", you set the port that is used for the TCP/IP connection. The port used by the Modicon PLCs is 502.
- "Server"
  You set the IP address or host name of the PLC under "Server". Only the IP address can be used on a Basic Panel.
- "Distributed slave address"
  For "Distributed slave address", set the slave address of the remote PLC, but only if a bridge is used.
  If a bridge is not used, the default value "255" (or "0") must be retained.

- "Change word order"
  The "Change word order " parameter only affects the word order of the 32-bit values display. The setting pertains to the data types Double, Double+/-, and Float. The byte order cannot be changed.
  - "Change word order" not activated
    The most significant byte is sent first.
    For double words, the least significant word is sent before the most significant word. This setting has been system-tested for all approved PLCs.
  - "Change word order" activated
    The most significant byte is sent first.
    For double words, the most significant word is sent before the least significant word.

    Note

    This setting must be used for the SIEMENS SENTRON PAC3200 and PAC4200 multi-function meters and can be used for PLCs of other manufacturers.
- "Use single write"
  If you deselect this function, only function codes 15H and 16H are used for writing into the PLC.
  If this function remains selected, the function codes 05H, 06H, 15H, and 16H are used.

Connecting HMI device to PLC (Basic Panels)

Connections via Modicon Modbus TCP/IP (Basic Panels)

Connection

The HMI device can be connected to the Modicon Modbus PLC using the following components:

- Existing Ethernet network that also contains the PLCs
- Cross-over Ethernet cable connected directly to the Ethernet interface of the CPU or the communication module

The connection of the HMI device to a Modicon Modbus PLC is limited primarily to the physical connection of the HMI device. Special blocks for the connection are not required in the PLC.

Note

Timeout response with TCP/IP (Ethernet)

Due to the use of the TCP/IP protocol, the breakdown of a connection is detected at the earliest after approximately one minute. Communication failure cannot be reliably detected if no tags are requested, for example, no output tags in the current screen.

Configure area pointer coordination for each PLC. This setting ensures that a communication failure is recognized after approximately two minutes, even in the aforementioned scenario.

Communication types (Basic Panels)

Approved communication types

The following communication types are system-tested and approved:

- Point-to-point coupling:
- Multiple point coupling of a HMI device (Modbus TCP/IP Client) with up to 4 PLCs, each with different couplings. CPU types can be mixed.
  The following couplings are possible:
  - Coupling the Ethernet CPU interface of the TSX Unity Quantum.
  - Coupling via the communication modules for Ethernet 140 NOE 771 01 for the TSX Quantum and TSX Unity Quantum series
  - Coupling via the Ethernet interface of the 171 CCC 980 30 CPU adapter of the Momentum series
  - Coupling the Ethernet CPU interface of the TSX Unity Premium.
  - Coupling via the Ethernet TCP/IP connect module TSX ETY 110 for the TSX Premium and TSX Unity Premium series
  - Coupling via the Ethernet TCP/IP connect module TSX ETY 410 for the Micro series
  - Coupling via the Ethernet TCP/IP Modbus Plus Bridge 174 CEV 200 40 to the Modbus Plus interface of the Compact, the TSX Quantum and the TSX Unity Quantum series

Via the TCP/IP Modbus Plus Bridge, 174 CEV 200 40, the PLCs can be accessed at their Remote Slave Address via the Ethernet interface of this bridge.

Note

Integration of the HMI device in a Modbus network via a bridge is not possible. The HMI device is the Modbus master.

Restrictions

The coupling of the HMI device to PLCs of other manufacturers who offer a Modbus TCP/IP interface is not system-tested and thus, not enabled.

However, if another PLC is to be used, observe the following instructions:

- Use the following CPU types, because these operate without address offset and in the usual bit count manner.
  - Unity, PL7: Premium, Micro, Quantum, M340
- The following function codes are used for the respective data areas:

Reading function codes		Address range	
01	ReadCoilStatus	0x / %M	DIGITAL_OUT
02	ReadInputStatus	1x / %I	DIGITAL_IN
03	ReadHoldingRegisters	4x / %MW	USERDATA

Reading function codes		Address range	
04	ReadInputRegisters	3x / %IW	ANALOG_IN
20 (14Hex)	ReadGeneralReference	6x / –	EXTENDEDMEMORY (not for all CPUs)

Writing function codes		Address range	
06 [1)]	PresetSingleRegister	4x / %MW	USERDATA Single
16 (10Hex)	PresetMultipleRegisters	4x / %MW	USERDATA Multiple
05 [1)]	ForceSingleCoil	0x / %M	DIGITAL_OUT with BIT
15 (0FHex)	ForceMultipleCoils	0x / %M	DIGITAL_OUT with 16 BIT GROUP
21 (15Hex)	WriteGeneralReference	6x / –	EXTENDEDMEMORY (not for all CPUs)

1) Select use with "Use single write".

Connectable PLCs

Connections can be implemented for the following Modicon Modbus PLCs:

Modicon Modbus PLC	Supported protocol	
	Modicon Modbus RTU [2)]	Modicon Modbus TCP/IP
TSX Compact	x	x [1)]
TSX Quantum	x	x
Momentum	x	x
Premium	-	x
Micro	-	x
M340 20x0 (without 2010)	-	x

1) Only via Ethernet TCP/IP Modbus Plus Bridge

2) Communication via RS 232 is tested and enabled for the PLC. In the HMIs that only have a RS 422/485 interface, the RS 422/232 converter with the order number 6AV6 671-8XE00-0AX0 was tested and enabled.

Performance features of communication (Basic Panels)

Permissible data types for Modicon Modbus TCP/IP (Basic Panels)

Permitted data types

The table lists the data types that can be used when configuring tags and area pointers.

Note

If you change the Modicon Modbus RTU communication driver to Modicon Modbus TCP/IP, the string in the "String" data type may be different.

Permitted data types for CPU type "Unity, PLC: Premium, Micro, Quantum M340"

Data type	Operand type	Length
+/- Double	%MW	4 bytes
+/- Int	%MW, %IW	2 bytes
16-bit group	%MW, %I	2 bytes
ASCII	%MW	0 to 80 characters
Bit	%MW, %IW, %M, %I	1-bit
Double	%MW	4 bytes
Float	%MW	4 bytes
Int	%MW, %IW	2 bytes

Note

The ranges "%I" and "%IW" are not supported for the following CPU types:

- Premium
- Micro
- M340

Permitted data types for CPU type "Concept, ProWORX: Compact, Quantum, Momentum"

Data type	Operand type	Length
+/- Double	4x, 6x	4 bytes
+/- Int	3x, 4x, 6x	2 bytes
16-bit group	0x, 1x	2 bytes
ASCII	4x, 6x	0 to 80 characters
Bit	0x, 1x, 3x, 4x, 6x	1-bit
Double	4x, 6x	4 bytes

Data type	Operand type	Length
Float	4x, 6x	4 bytes
Int	3x, 4x, 6x	2 bytes

Bit counting method

The usual bit counting method "16 LSB - 1 MSB" in the following CPU types is only used in the "HMI tags" editor with the selected "Bit" data type:

- Concept, ProWORX: Compact, Quantum, Momentum

The following bit location assignment applies:

	Left byte								Right byte							
Counting with tags	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Format for "Signed"

The placeholder "+/-" stands for the data types "Signed Int" and "Signed Double".

Supported CPU types for Modicon Modbus TCP/IP (Basic Panels)

CPU types

The following CPU types are supported for configuring the Modicon Modbus TCP/IP communication driver.

- Compact
- Momentum
- Quantum
  - Concept Quantum
  - Unity Quantum
- Micro
- Premium
- Modicon M340
  - 20x0 (except 2010)

Address areas for Modicon Modbus TCP/IP (Basic Panels)

UnityPl7

Address areas	Data types							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
%1	%I0 - %I65535	%I65535 %I0 - %I65520	--	--	--	--	--	--
%M	%M0 - %M65535	%M65535 %M0 - %M65520	--	--	--	--	--	--
%IW	%IW0.0 - %IW65535.15	--	%IW0 - %IW65535	%IW0 - %IW65535	--	--	--	--
%MW	%MW0.0 - %MW65535.15	--	%MW0 - %MW65535	%MW0 - %MW65535	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65535

ConceptProWORX

Address areas	Data types							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
0x	0x1 - 0x65535	0x1 - 0x65520	--	--	--	--	--	--
1x	1x100001 - 1x165535	1x100001 - 1x165520	--	--	--	--	--	--
3x	3x300001.1 - 3x365535.16	--	3x300001 - 3x365535	3x300001 - 3x365535	--	--	--	--
4x	4x400001.1 - 4x465535.16	--	4x400001 - 4x465535	4x400001 - 4x465535	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465535
6x	6x60000.1:1 - 6x69999.16:10	--	6x60000:1 - 6x69999:10	6x60000:1 - 6x69999:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69999:10

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance. Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.
- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Modicon Modbus RTU (Basic Panels)

Configuring a connection via Modicon Modbus RTU (Basic Panels)

Introduction

You configure a connection to a PLC with a Modicon Modbus RTU communication driver in the "Connections" editor of the HMI device.

The interfaces are named differently depending on the HMI device.

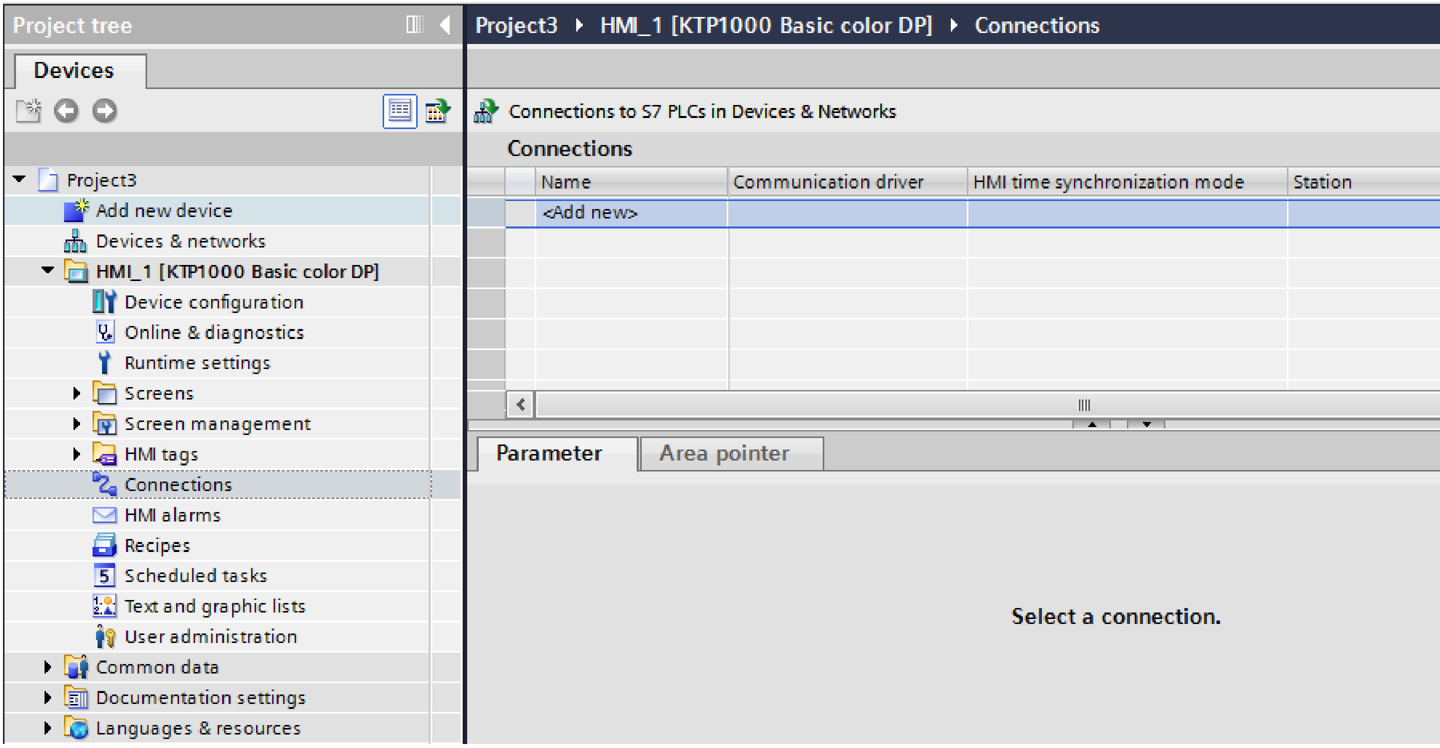
Requirements

- A project is open.
- An HMI device has been created.

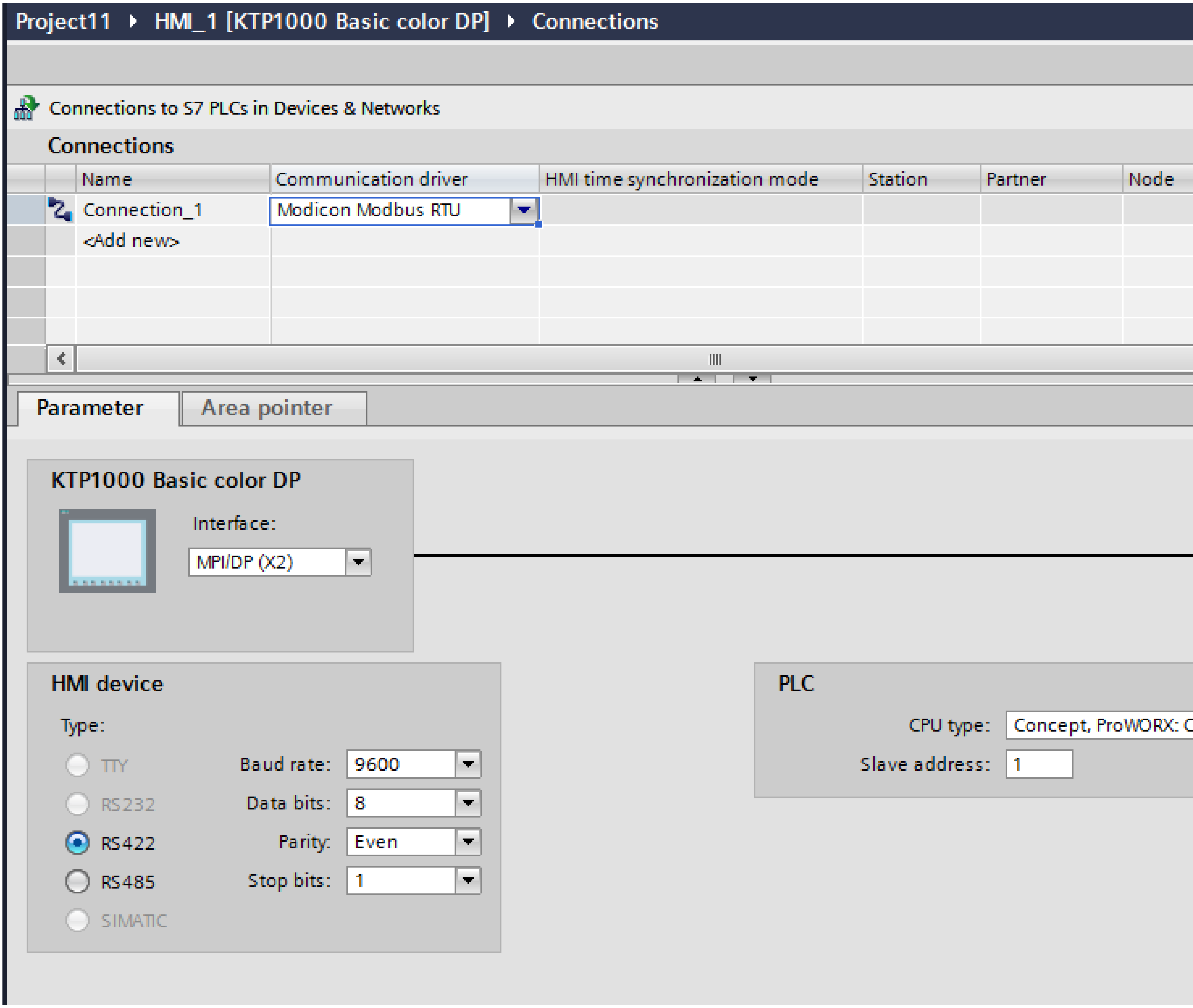
Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.

3. Double-click "<Add>" in the "Connections" editor.



4. In the "Communication drivers" column, select the "Modicon Modbus RTU" driver.



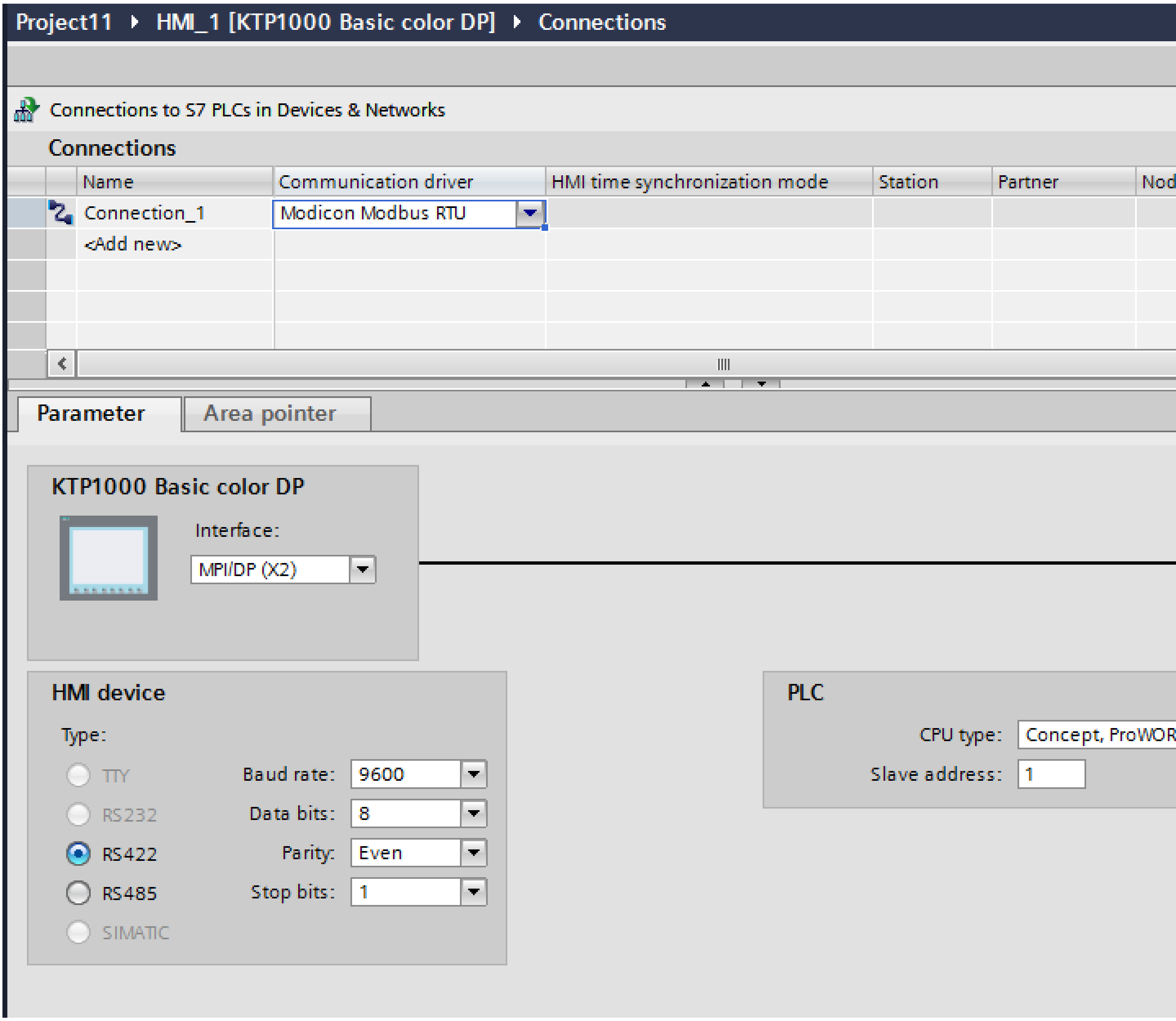
5. Select all necessary connection parameters for the interface in the Inspector window under "Parameters".

Parameters for the connection (Modicon Modbus RTU) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device" and "PLC" areas are available for assigning parameters according to the interface used.



Parameters for the HMI device

You can select an interface for the HMI device in the Inspector window under "Parameters". Depending on the HMI device, there are several interfaces available.

- Type
  Only RS 232 is system-tested.
  No warranty is given for RS 485.

  Note

  RS 422 is only approved in combination with the RS 422-RS 232 converter.

  Order number: 6AV6 671-8XE00-0AX0

  Note

  If you use the IF1B interface, you must switch over the RS422 receive data additionally by 4 DIL-switches on the back of the HMI device.

- Baud rate
  For "Baud rate", select the transmission speed between the HMI device and Modicon PLC.
  A baud rate of 19200 or 9600 can be selected for the communication.
  A baud rate of 4800 can be selected for certain HMI devices.
- Data bits
  For "Data bits", only the value "8" can be selected.
- Parity
  For "Parity", you can choose from "None", "Even", and "Odd".
- Stop bits
  For "Stop bits", you can choose between 1 and 2 bits.

Parameters for the PLC

- CPU type
  For "CPU type", you set the Modicon PLC to which the HMI device is connected.
  You can select the following CPUs:
  – Concept, ProWORX: Compact, Quantum
- Slave address
  Under "Slave address" you set which slave address the CPU has.

Connecting HMI device to PLC (Basic Panels)

Connections via Modicon Modbus RTU (Basic Panels)

Connection

Connect the HMI device to the Modicon Modbus RTU interface of the Modicon Modbus RTU slave.

The connection of the HMI device to Modicon is limited primarily to the physical connection of the HMI device. Special blocks for the connection are not required in the PLC.

Connection cable

The following connecting cables are available to connect the HMI device to Modicon Modbus.

Interface to the HMI device	Modicon PLC		
	directly via Modbus interface (RS232) 9-pin Sub D male connector	Via MB Bridge (RS 232)	directly via Modbus interface (RS232) 8-pin RJ45 connector
RS 232, 9-pin	PP1	PP1	PP2

The cable pin assignments can be found in Section "Connecting cables for Modicon Modbus RTU".

Communication types (Basic Panels)

Approved communication types

The following communication types are system-tested and approved:

- Ppoint-to-point connection only via the RS-232 interface.
- Multipoint connection from a HMI device (Modbus-Master) with up to 4 PLCs: The HMI device must be connected with a Modbus Plus Bridge or a Compact, Momentum CPU or TSX Quantum CPU which is configured as a Modbus Plus Bridge.
- You connect the other PLCs via the Modbus Plus connection on the first PLC. The PLCs can be reached under their address via the bridge functionality of the first PLC.

  Note

  It is not possible to integrate the HMI device into a Modbus network because the HMI device is Modbus-Master.

- Integration of the HMI device into a Modbus Plus network via the "bridge mode" of the Compact, Momentum or Quantum (logical point-to-point communication of the HMI device with a Compact, Momentum or Quantum).

Restrictions

The connection of the HMI device to PLCs of other manufacturers which offer a Modicon Modbus interface is not system-tested and therefore not approved.

If you use another PLC nevertheless, observe the following information:

- These drivers only work for tags with the bit counting method typical for Modicon PLCs from left (bit1 = most significant bit) to right (bit16 = least significant bit in data type INT).
- The address offset displayed in the configuring is subtracted at protocol level in the message frame. E.g. in Holding Register 4x the offset "40001". The configured address "40006" therefore becomes address "5" in the message frame. The address (e.g. "5") transferred in the message frame is transformed to the PLC-specific address range in the different Non-Modicon PLCs.
- A reply message frame without "ExceptionCode" is expected within 500 ms.
- The following function codes are used for the respective data areas:

Reading function codes		Address range	
01	ReadCoilStatus	0x	DIGITAL_OUT
02	ReadInputStatus	1x	DIGITAL_IN
03	ReadHoldingRegisters	4x	USERDATA
04	ReadInputRegisters	3x	ANALOG_IN
20 (14Hex)	ReadGeneralReference	6x	EXTENDEDMEMORY (not for all CPUs)

Writing function codes		Address range	
06	PresetSingleRegister	4x	USERDATA Single
16 (10Hex)	PresetMultipleRegisters	4x	USERDATA Multiple
05	ForceSingleCoil	0x	DIGITAL_OUT with data type Bit
15 (0FHex)	ForceMultipleCoils	0x	DIGITAL_OUT with data type 16 bit group
21 (15Hex)	WriteGeneralReference	6x	EXTENDEDMEMORY (not for all CPUs)

Connectable PLCs

Connections can be implemented for the following Modicon Modbus PLCs:

Modicon Modbus PLC	Supported protocol	
	Modicon Modbus RTU [2]	Modicon Modbus TCP/IP
TSX Compact	x	x [1]
TSX Quantum	x	x
Momentum	x	x
Premium	-	x
Micro	-	x
M340 20x0 (without 2010)	-	x

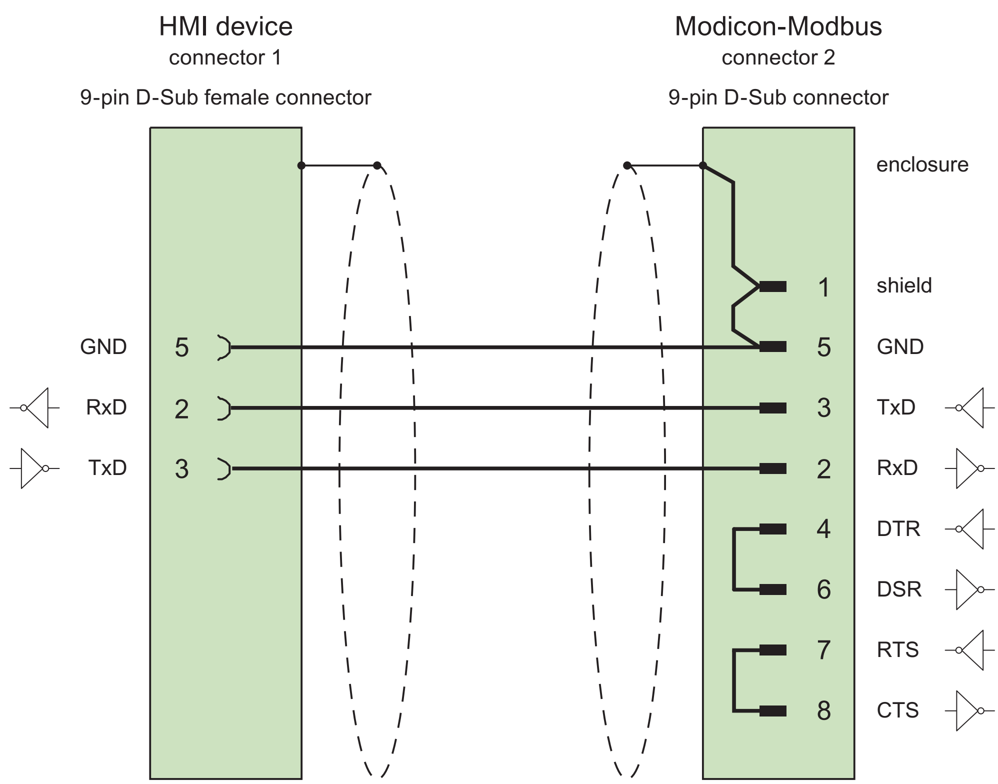
1) Only via Ethernet TCP/IP-Modbus Plus Bridge

2) Communication via RS 232 is tested and enabled for the PLC. In the HMI devices which only have an RS 422/485 interface, the RS 422/232 converter with the order number 6AV6 671-8XE00-0AX0 was tested and approved.

Connecting cables for Modicon Modbus RTU (Basic Panels)

Connecting cable PP1, RS-232, for Modicon (Basic Panels)

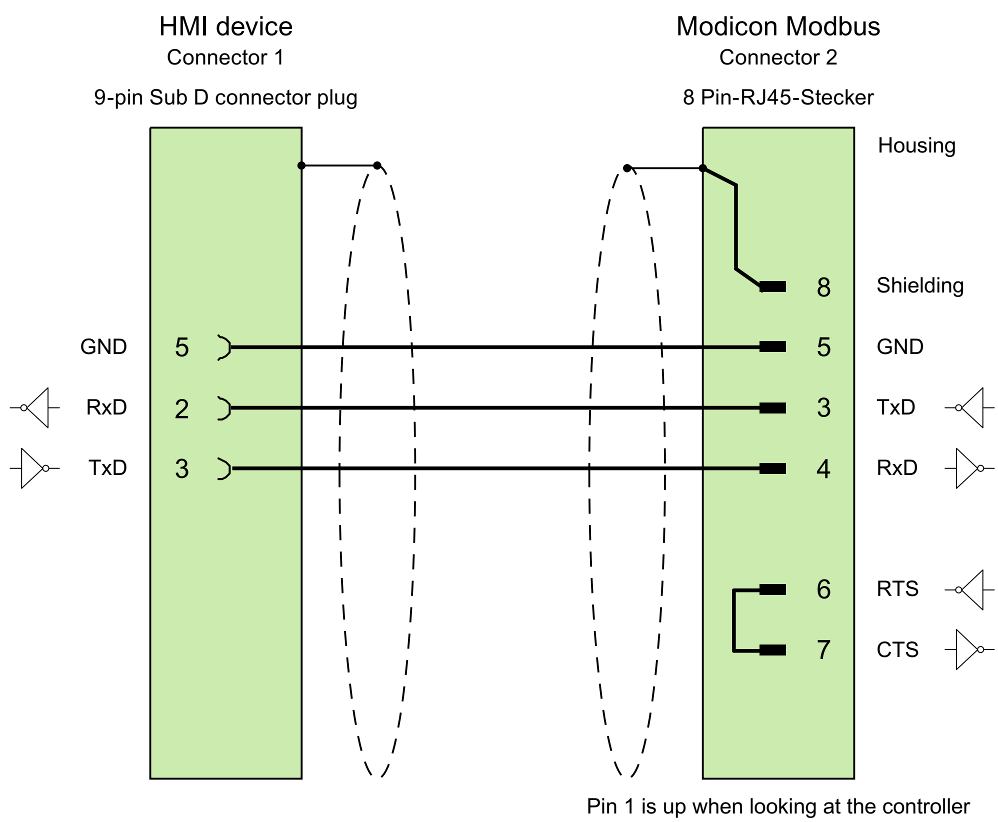
Point-to-point cable 1: PLC > PC ...



Cables: 3 x 0.14 mm², shielded,
max. length 15 m

Connecting cable PP2, RS-232, for Modicon (Basic Panels)

Point-to-point cable 2: PLC (TSX Compact) > PC...



Cables: 3 x 0.14 mm², shielded,
max. length 15 m

Performance features of communication (Basic Panels)

Permitted data types for Modicon Modbus RTU (Basic Panels)

Permitted data types

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Operand type	Length
+/- Double	4x, 6x	4 bytes
+/- Int	3x, 4x, 6x	2 bytes
16-bit group	0x, 1x	2 bytes
ASCII	4x, 6x	0 to 80 characters
Bit [1)]	0x, 1x, 3x, 4x, 6x	1-bit
Double	4x, 6x	4 bytes
Float	4x, 6x	4 bytes
Int	3x, 4x, 6x	2 bytes

1) Note the following for write accesses:

For data type "Bit" with the operand types "4x" and "6x", the entire word is written back to the PLC following a change to the specified bit. There is no check to determine whether any other bits in the word have changed. As a result, the PLC only has read access to the specified word.

The usual bit counting method (16 LSB - 1 MSB) in the following CPU types is only used in the "HMI tags" editor with the selected "Bit" data type:

- Concept ProWORX: Compact, Quantum

The following bit location assignment applies:

	Left byte								Right byte							
Counting with tags	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Format for "Signed"

The placeholder "+/-" stands for the data types "Signed Int" and "Signed Double".

Supported CPU types for Modicon Modbus RTU (Basic Panels)

CPU types

The following CPU types are supported in the configuration of the Modicon Modbus RTU communication driver.

- Compact
- Momentum
- Quantum

Address areas for Modicon Modbus RTU (Basic Panels)

UnityPl7

Address areas	Data types							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
%1	%I0 - %I65535	%I65535 %I0 - %I65520	--	--	--	--	--	--
%M	%M0 - %M65535	%M65535 %M0 - %M65520	--	--	--	--	--	--
%IW	%IW0.0 - %IW65535.15	--	%IW0 - %IW65535	%IW0 - %IW65535	--	--	--	--
%MW	%MW0.0 - %MW65535.15	--	%MW0 - %MW65535	%MW0 - %MW65535	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65535

ConceptProWORX

Address areas	Data types							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
0x	0x1 - 0x65535	0x1 - 0x65520	--	--	--	--	--	--
1x	1x100001 - 1x165535	1x100001 - 1x165520	--	--	--	--	--	--
3x	3x300001.1 - 3x365535.16	--	3x300001 - 3x365535	3x300001 - 3x365535	--	--	--	--

Address areas	Data types							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
4x	4x400001.1 - 4x465535.16	--	4x400001 - 4x465535	4x400001 - 4x465535	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465535
6x	6x60000.1:1 - 6x69999.16:10	--	6x60000:1 - 6x69999:10	6x60000:1 - 6x69999:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69999:10

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance. Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.
- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Data exchange (Basic Panels)

Area pointers for Modicon Modbus (Basic Panels)

Area pointers for connections via Modicon Modbus communication drivers

You use an area pointer to access a data area in the PLC.

For more detailed information on area pointers and their configuration, refer to Section "Data exchange using area pointers (Page 451)".

Special considerations for connections via Modicon communication drivers

You can configure the following area pointers

Area pointers	Modicon Modbus TCP/IP	Modicon Modbus RTU
Screen number	Yes	Yes
Date/time	Yes	Yes
Date/time PLC	Yes	Yes
Coordination	Yes	Yes
Project ID	Yes	Yes
Job mailbox	Yes	Yes
Data record	Yes	Yes

Restrictions Modicon Modbus TCP/IP

The following restrictions apply for configuring area pointers.

CPU type	Data types	File types
Concept, ProWORX: Compact, Quantum, Momentum	+/- Int, Int	4x, 6x
Unity, PL7: Premium, Micro, Quantum, M340	+/- Int, Int	%MW

Modicon Modbus RTU restrictions

The following restrictions apply for configuring area pointers.

CPU type	Data types	File types
Concept, ProWORX: Compact, Quantum, Momentum	+/- Int, Int	4x, 6x

Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configuring alarms for non-integrated connections

Several steps are necessary to configure alarms such as warnings, error messages and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with a Modicon Modbus communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
All Modicon series	Int, +/-Int	16 Bit Group, Int, +/-Int, Double, +/-Double, Float

Arrays and array tags cannot be used for discrete alarms.

How the bit positions are counted

For connections with a Modicon Modbus communication driver, the following counting method applies:

How the bit positions are counted	Left byte								Right byte							
In WinCC you configure:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

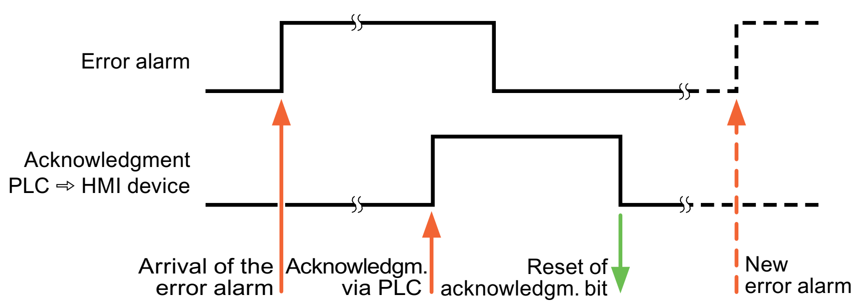
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment

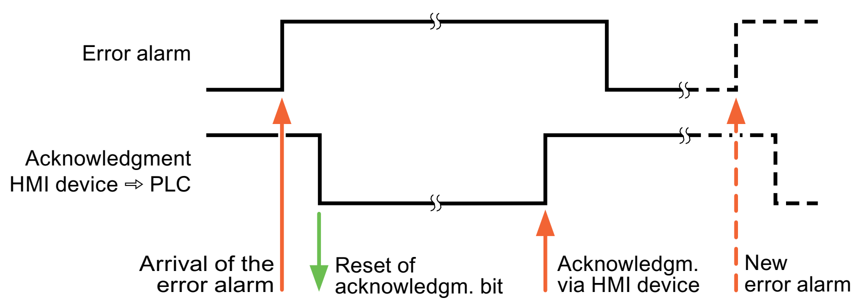
tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.13.3.4 Omron (Basic Panels)

Omron communication drivers (Basic Panels)

Introduction

This section describes the communication between an HMI device and PLCs that use Omron communication drivers.

The following communication drivers are supported:

- Omron Host Link

Data exchange

Data is exchanged by means of tags or area pointers.

- Tags
  The PLC and the HMI device use process values for data exchange. You create tags in the configuration that point to addresses in the PLC. The HMI device reads the value from the defined address, and then displays it. The operator may also enter values on the HMI device, which are then written to the address in the PLC.
- Area pointers
  Area pointers are used to exchange specific data and are only set up when these data are used.

Omron Host Link (Basic Panels)

Configuring a connection via Omron Host Link (Basic Panels)

Introduction

You configure a connection to a PLC with an Omron Host Link communication driver in the "Connections" editor of the HMI device.

Note

Connection with Omron Host Link

A connection will not automatically be established when runtime is started if you have configured a connection via Omron.

A tag which is in the valid PLC memory area must be configured in the runtime start screen.

The connection will otherwise only be established once a corresponding screen has been selected.

This tag will be accessed when runtime is started and a connection will then be established.

The interfaces are named differently depending on the HMI device.

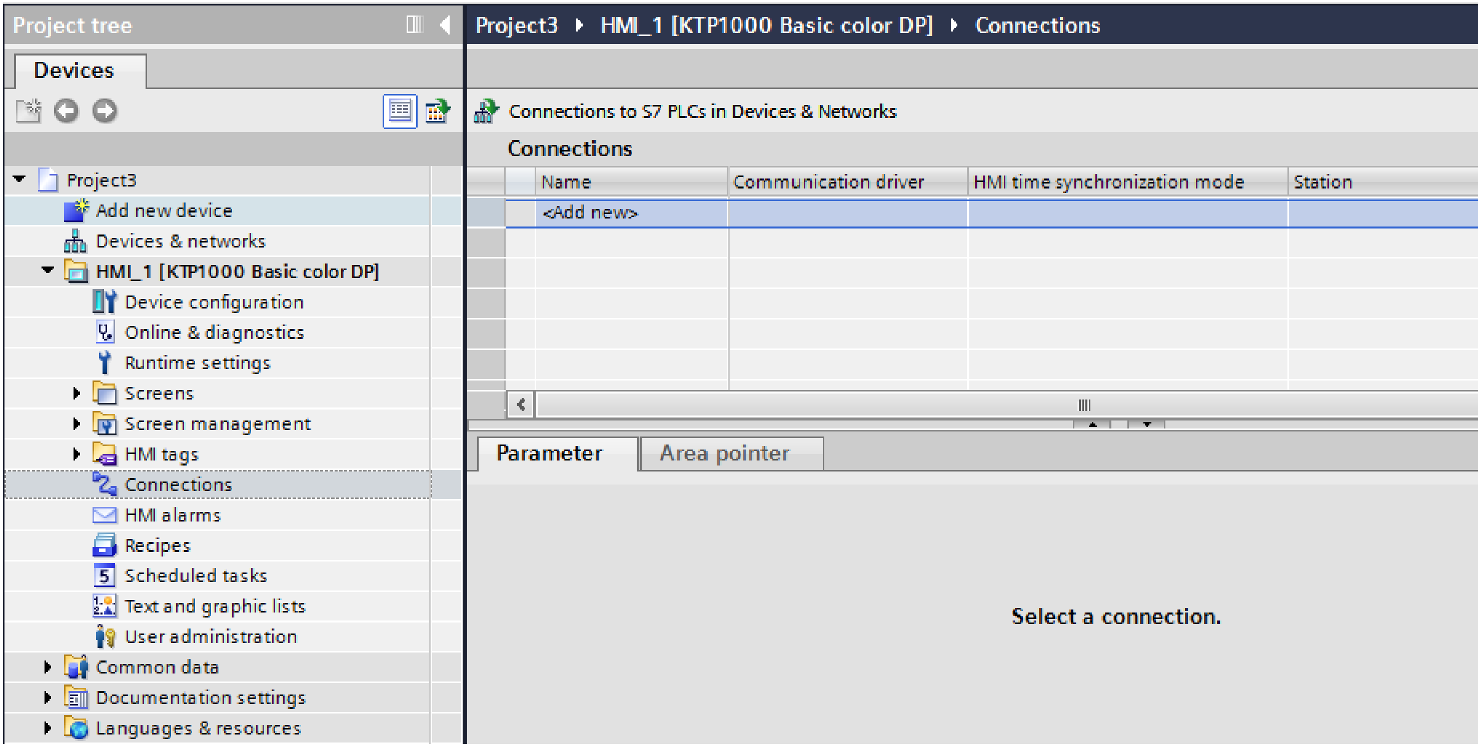
Requirements

- A project is open.
- An HMI device has been created.

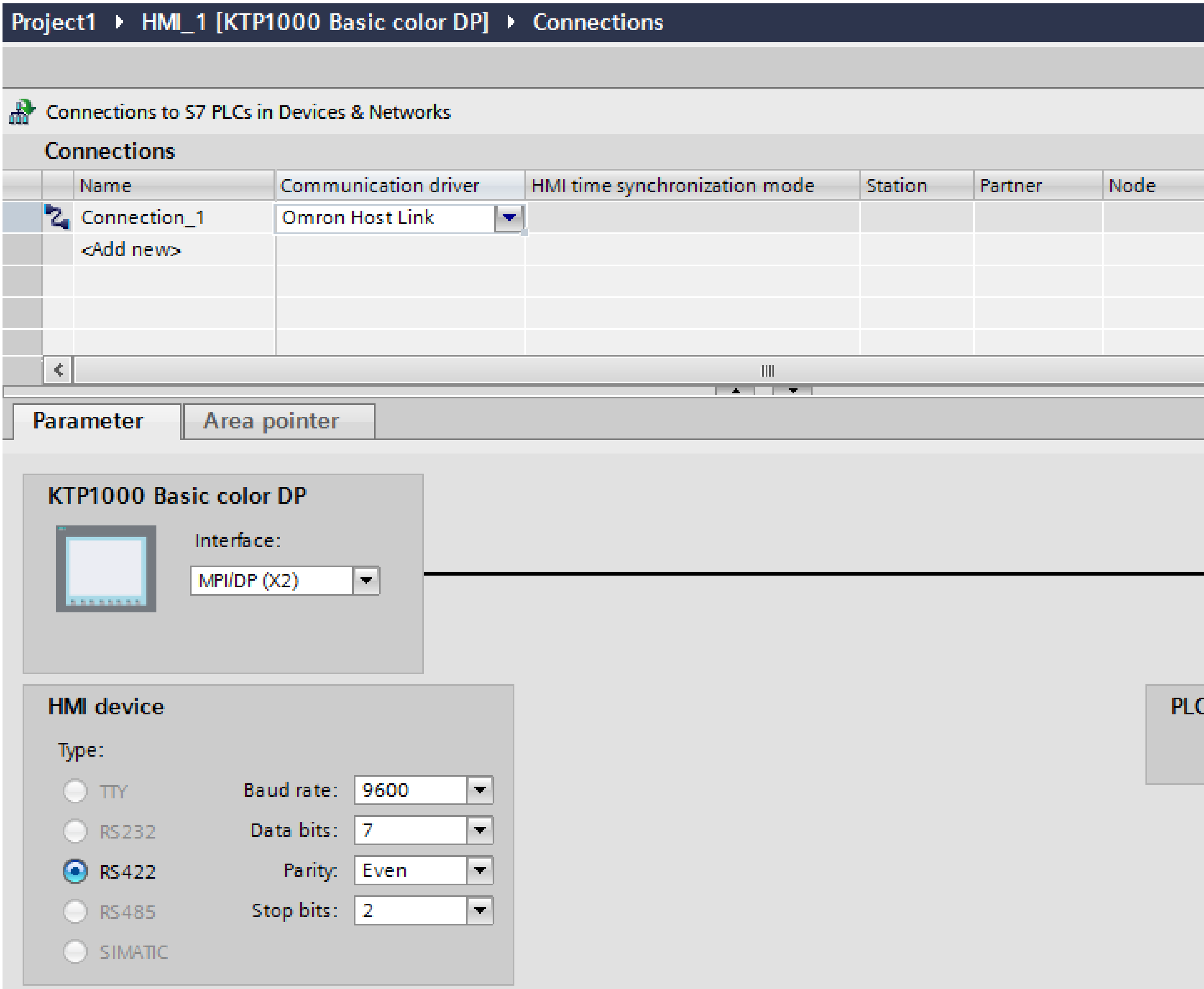
Procedure

1. Double-click the HMI device under "Devices" in the project tree.
2. Double-click the "Connections" item.

3. Double-click "<Add>" in the "Connections" editor.



4. Select the "Omron Host Link" driver in the "Communication driver" column.



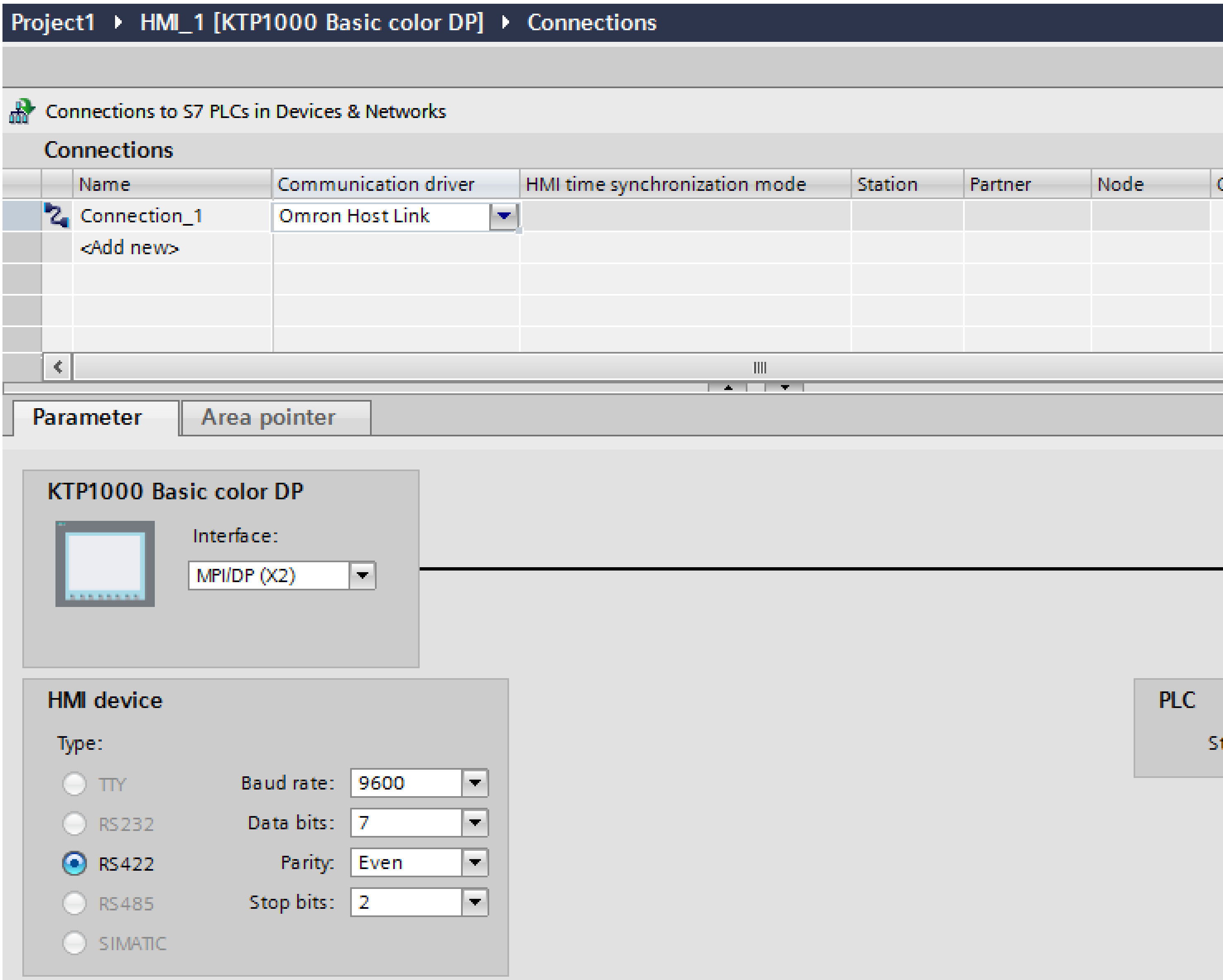
5. Select all necessary connection parameters for the interface in the inspector window under "Parameters".

Parameters for the connection (Omron Hostlink) (Basic Panels)

Parameters to be set

To set the connection parameters, such as addresses and profiles, click the connection that you have created in the "Connections" editor.

The communication partners are displayed schematically in the Inspector window under "Parameters". The "HMI device" and "PLC" areas are available for assigning parameters according to the interface used.



Parameters for the HMI device

You can select an interface for the HMI device in the Inspector window under "Parameters". Depending on the HMI device, there are several interfaces available.

- Type
  Specifies the physical connection used.
- Baud rate
  For "Baud rate", you set the transmission speed of the HMI device to OMRON. A baud rate of 19200 or 9600 can be selected for the communication.
- Data bits
  For "Data bits", you can choose between "7 bits" and "8 bits".
- Parity
  For "Parity", you can choose from "None", "Even", and "Odd".
- Stop bits
  For "Stop bits", you can choose between 1 and 2 bits.

Parameters for the PLC

- Station address
  For "Station address", set the station number of the connected PLC.

Connecting HMI device to PLC (Basic Panels)

Connections via Omron Host Link (Basic Panels)

Connection

The connection of the HMI device to an OMRON PLC is limited primarily to the physical connection of the HMI device. Special blocks for the connection are not required in the PLC.

Connection cable

The following connecting cables are available to connect the HMI device to an Omron PLC.

Interface to the HMI device	Omron PLC			
	RS232, 9-pin	RS232 I/O port	RS422, 9-pin	RS422, terminals/ pins
RS232, 9-pin	PP1	Programming cable (standard cable of Omron)	—	—
RS232 via converter	—	—	—	Multi-point cable 1
RS422, 9-pin	—	—	PP2	Multi-point cable 2

Refer to the relevant device manual to determine which HMI device interface is to be used.

Communication types (Basic Panels)

Approved communication types

The connection from a HMI device to an OMRON-CPU with the Omron Host Link protocol via RS232 and via RS 422 is system-tested and approved by Siemens AG.

This concerns the following CPU types:

- CP1x (CP1L, CP1H, CP1E)
- CJ1x (CJ1M, CJ1H, CJ1G)
- CJ2H
- CS1x (CS1G, CS1H, CS1D)
- CPM2C

Note

Only the following CPU types have been tested and released for Basic Panels, TP 177A and OP 77A:

- CP1x (CP1L, CP1H, CP1E)
- CJ1x (CJ1M, CJ1H, CJ1G)

Multipoint connection

A multipoint connection to the up to 4 approved OMRON PLCs in a RS422-four-wire connection can be implemented with communication modules on the PLCs and is system-tested and approved by Siemens AG.

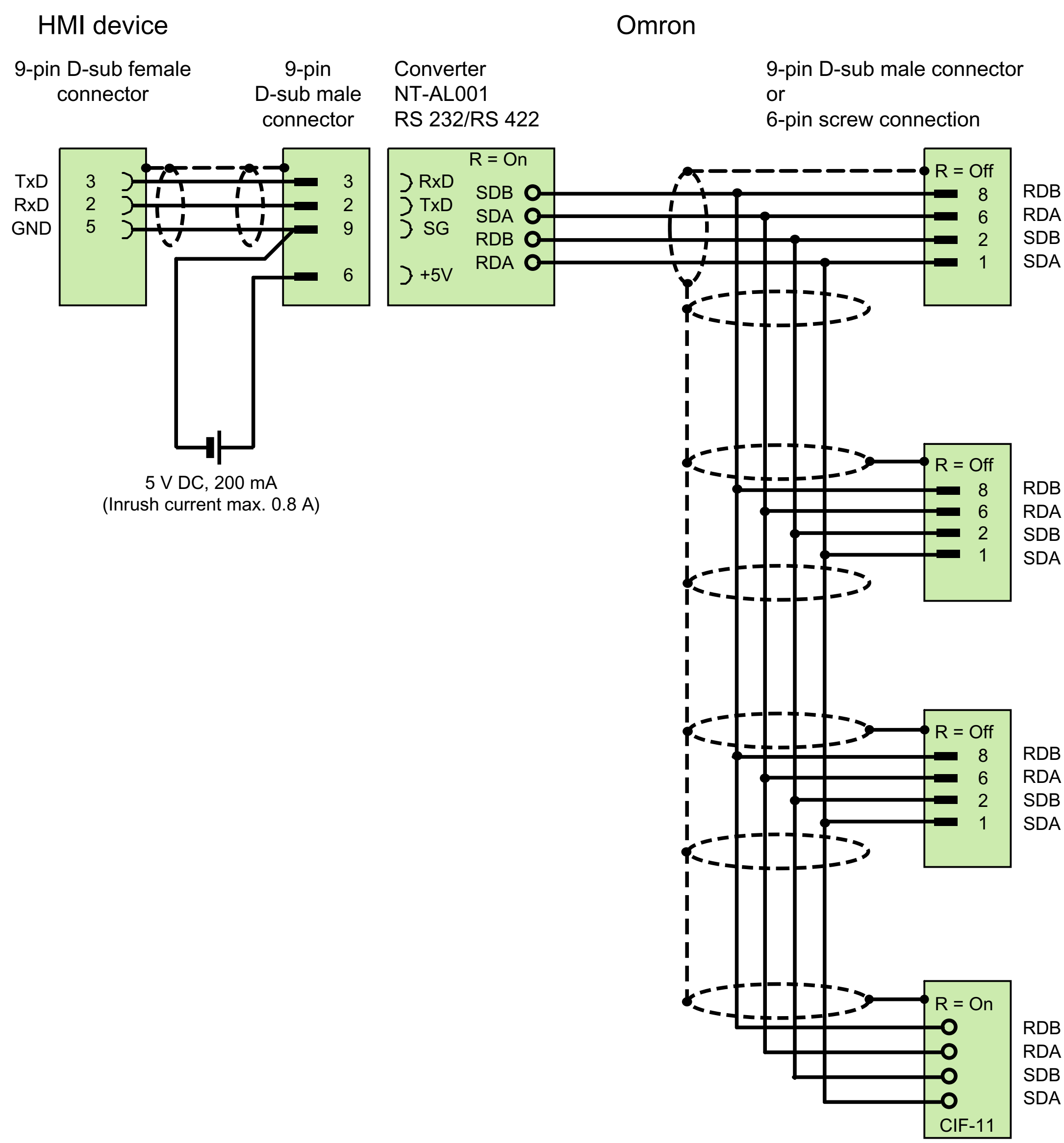
Note

The HMI device can only be operated as a master. Exactly one master is possible in the RS422-four-wire-Multidrop connection.

Connecting cable (Basic Panels)

Connecting cable MP1, RS-232, over converter, for Omron (Basic Panels)

Multipoint cable 1: MP/TP/PC > PLC

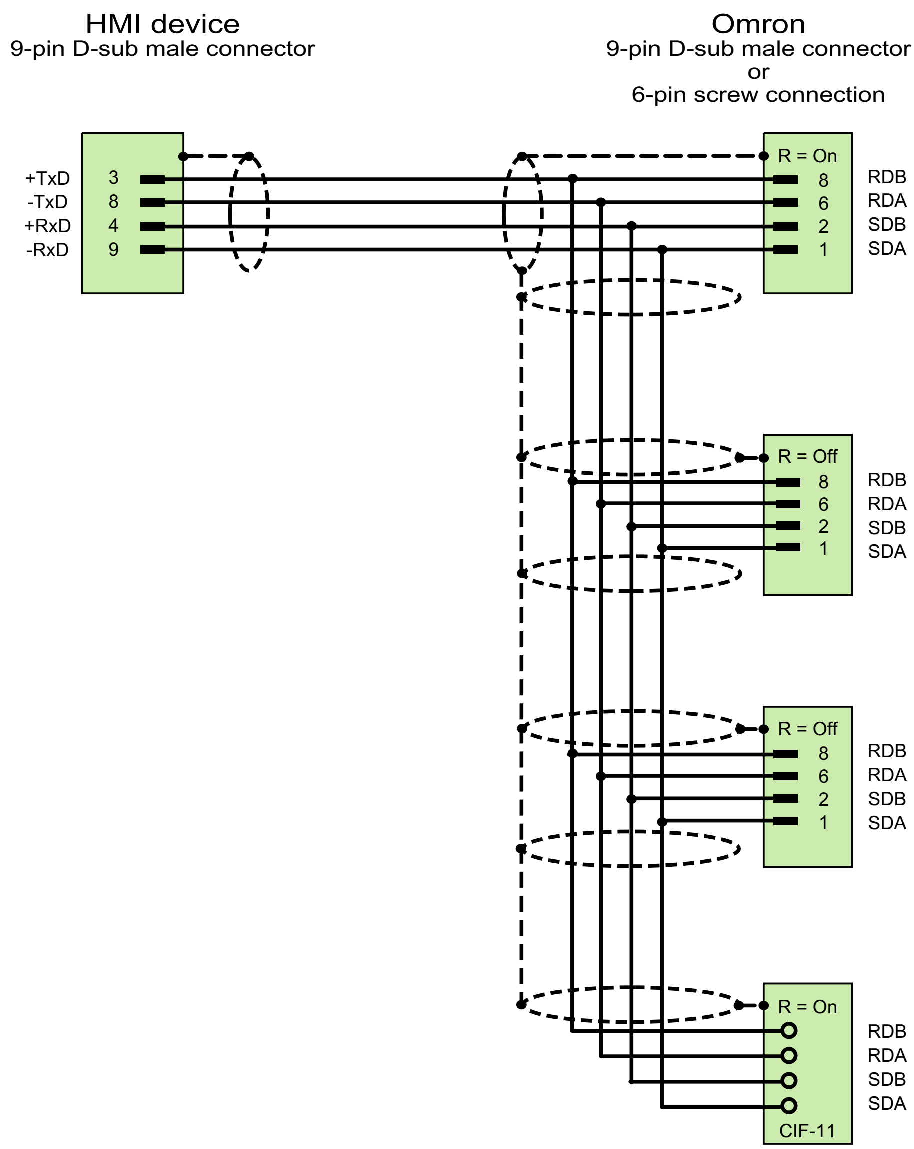


[1)] Inrush current max. 0.8 A

shielded, max. length 500 m

Connecting cable MP2, RS-422, for Omron (Basic Panels)

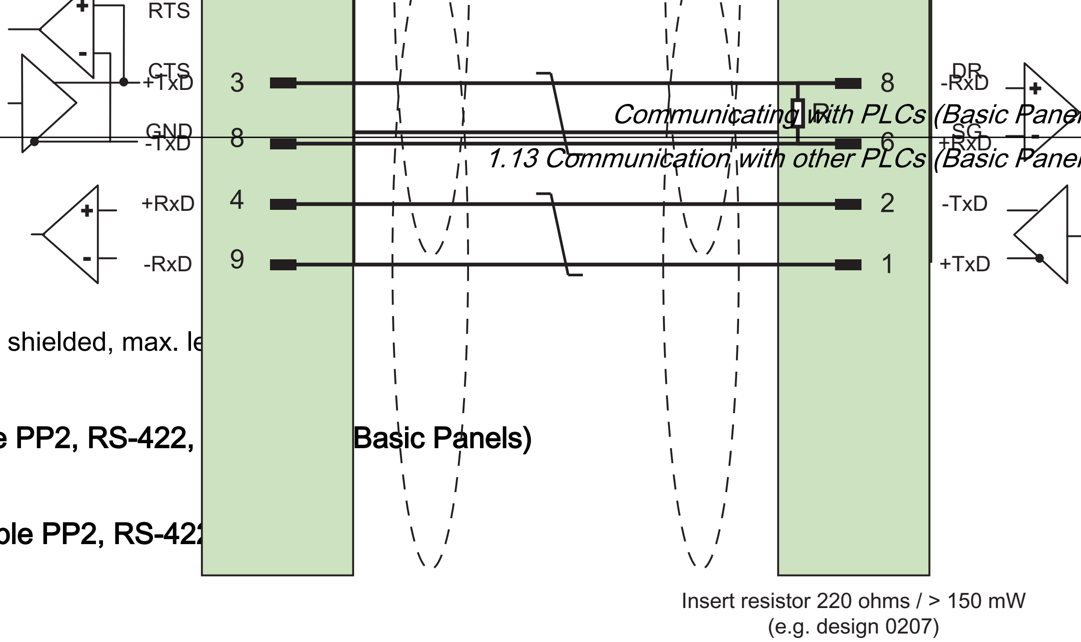
Multipoint cable 2: RS422, MP/TP/PC > SPS_



shielded, max. length 500 m

Connecting cable PP1, RS-232, for Omron (Basic Panels)

Point-to-point cable PP1, PC/TP/OP - PLC



shielded, max. le

Connecting cable PP2, RS-422, Basic Panels)

Point-to-point cable PP2, RS-422

shielded, max. length 500 m

Performance features of communication (Basic Panels)

Permissible data types for Omron Host Link (Basic Panels)

Permitted data types

The table lists the data types that can be used when configuring tags and area pointers.

Data type	Operand type	Length
Bool	I/O, HR, AR, LR, DM, T/C bit, CPU status	1-bit
Byte	CPU type	1 byte
DInt	HR, AR, LR, DM	4 bytes
Int	I/O, HR, AR, LR, DM, T/C Val	2 bytes
Real	HR, DM	4 bytes
String	HR, AR, LR, DM	0 to 80 characters
UDInt	HR, AR, LR, DM	4 bytes
UInt	I/O, HR, AR, LR, DM, T/C Val	2 bytes

Note

Read and write operations of all data areas in the OMRON PLC can only be reliably carried out in "STOP" or "MONITOR" mode.

"I/O" refers either to the IR/SR area or the CIO area depending on the PLC series. The operand types "LR", "HR" and "AR" are not available in all PLC series.

Note

Note the following for write accesses:

For the "Bool" data type with the operand types "I/O", "HR", "AR", "LR" and "DM", the entire word is written back into the PLC when the specified bit is changed. There is no check to determine whether any other bits in the word have changed. As a result, the PLC only has read access to the specified word.

Operand type old PLC	Operand type CS and CJ PLC
CPU Status	CPU Status
I/O	CIO
HR	H Range 0-511
AR	A
LR	n/a 1)
DM	D
T/C	T/C
CPU type	CPU type

1) You do not get an error message when you read or write the LR area in the following PLCs

- CS
- CJ
- CP

Supported CPU types for Omron Host Link (Basic Panels)

CPU types

The following CPU types are supported in the configuration of the Omron Host Link communication driver.

- CP1
  - CP1L
  - CP1H
  - CP1E
- CJ1
  - CJ1M
  - CJ1H
  - CJ1G

- CJ2
  - CJ2H
- CS1
  - CS1G
  - CS1H
  - CS1D
- CPM
  - CPM2C

Addressing in Omron Host Link (Basic Panels)

Addressing of PLCs in Omron Host Link

In PLCs of the series CS, CP and CJ, the timers 0-4095 are addressed with T/C 0-2047.

The counters 0-4095 must be addressed with an offset of 2048 (T/C 2048-4095 correspond to the counters 0-2047). Counters and timers with addresses > 2047 cannot be addressed via Host Link.

Counters and timers with addresses > 2047 cannot be addressed via Host Link.

Example:

If you want to address counter C20, you must address T/C 20+2048 = T/C 2068.

Address areas for Omron Host Link (Basic Panels)

Omron

Address areas	Data types							
	Bool	Byte	UInt	Int	UDInt	DInt	Real	String
I/O	I/O 0.0 - I/O 9999.15	--	I/O 0 - I/O 9999	I/O 0 - I/O 9999	--	--	--	--
HR	HR 0.0 - HR 9999.15		HR 0 - HR 9999	HR 0 - HR 9999	HR 0 - HR 9998	HR 0 - HR 9998	HR 0 - HR 9999	HR 0 - HR 9999
AR	AR 0.0 - AR 9999.15		AR 0 - AR 9999	AR 0 - AR 9999	AR 0 - AR 9998	AR 0 - AR 9998		AR 0 - AR 9999
LR	LR 0.0 - LR 9999.15		LR 0 - LR 9999	LR 0 - LR 9999	LR 0 - LR 9998	LR 0 - LR 9998		LR 0 - LR 9999
DM	DM 0.0 - DM 9999.15		DM 0 - DM 9999	DM 0 - DM 9999	DM 0 - DM 9998	DM 0 - DM 9998	DM 0 - DM 9999	DM 0 - DM 9999
T/C Bit	T/C Bit 0 - T/C Bit 4095							

Address areas	Data types							
	Bool	Byte	UInt	Int	UDInt	DInt	Real	String
T/C Val			T/C Val 0 - T/C Val 4095	T/C Val 0 - T/C Val 4095				
CPU Status	RUN, MONITOR							
CPU type		CPU type						

Commissioning components (Basic Panels)

Transferring a project to the HMI device

1. Switch the HMI device to "transfer mode".
2. Set all necessary transfer parameters.
   – Interface
   – Transfer parameters
   – Target storage location
3. Start the project transfer.
   The project is compiled automatically.
   All compilation and transfer steps are logged to a message window.

Interconnecting the PLC with the HMI device

1. Interconnect the PLC with the HMI device using a suitable cable.
2. The message "Connection to PLC .... is established" is output to the HMI device.

Optimizing the configuration (Basic Panels)

Acquisition cycle and update time

The acquisition cycles for the "Area pointers" and of the tags defined in the configuration software are decisive in terms of the update times which can actually be achieved.

The update time is equivalent to the acquisition cycle + transmission time + processing time.

Items to observe when optimizing the update times in configuration data:

- Optimize the maximum and minimum size of the data areas.
- Acquisition cycles which are too short lead to unnecessary load on overall performance.
  Set the acquisition cycle according to the rate of change of the process values. The rate of temperature changes at a furnace, for example, is significantly slower compared to the speed rate of an electrical drive. A time of approx. 1 second is a benchmark for the acquisition cycle.

- Avoid any gaps when entering the alarm or screen tags in a data area.
- Changes in the PLC can only be detected reliably if these are available at least within the actual acquisition cycle.

Screens

The refresh rate of screens is determined by the type and volume of data to be visualized.

Only configure short acquisition cycles for objects which actually require shorter refresh cycles. This procedure reduces update times.

Trends

The HMI device always updates all bit-triggered trends whose group bit is set in the "Trend transfer area". It resets the bits in the next cycle.

The group bit in the PLC program can only be set again after the HMI device has reset all bits.

Job mailboxes

A high rate and volume of job mailboxes transferred may lead to overload in communication between the HMI device and the PLC.

The HMI device confirms acceptance of the job mailbox by entering the value zero in the first data word of the job mailbox. The HMI device now processes the job for which it requires a certain time slice. It may take the HMI device some time to process a new job mailbox which is transferred in immediate succession to the job mailbox. The next job mailbox is only accepted if sufficient computing resources are available.

Data exchange (Basic Panels)

Area pointers for Omron (Basic Panels)

Area pointers in connections via Omron communication drivers

You use an area pointer to access a data area in the PLC.

For more detailed information on area pointers and their configuration, refer to Section:

"Data exchange using area pointers".

Special features of connections via Omron Host Link

Area pointers can only be created in the following "File types": "DM", "I/O", "HR", "AR", and "LR".

See also

Data exchange using area pointers (Page 451)

Trends (Basic Panels)

Trends

A trend is the graphical representation of one or more values from the PLC. The value is read out time-triggered for Basic Panels.

For additional information see:

AUTOHOTSPOT

Time-triggered trends

The HMI device reads in the trend values cyclically at an interval specified in the configuration.

Time-triggered trends are suitable for continuous curves, such as the operating temperature of a motor.

Alarms (Basic Panels)

Configuring alarms (Basic Panels)

Configuring alarms for non-integrated connections

Several steps are necessary to configure alarms such as warnings, error messages and acknowledgement.

- Step 1: Create tags
- Step 2: Configure alarms
- Step 3: Configure acknowledgment

You can find additional information in the section:

AUTOHOTSPOT

Distinctive features when configuring alarms

If you are configuring connections of HMI devices to PLCs of other manufacturers, note the following distinctive features when configuring:

- Data types of the tags
- Addressing of tags
- How the bit positions are counted

Data types

For connections with an Omron communication driver, the following data types are supported:

PLC	Permitted data types	
	Discrete alarms	Analog alarms
CP1, CJ1, CJ2, CS1, CPM	Uint, int	UInt, Int, UDInt, DInt

How the bit positions are counted

For connections with an Omron communication driver, the following counting method applies:

How the bit positions are counted	Left byte								Right byte							
In Omron PLCs	15							8	7							0
In WinCC you configure:	15							8	7							0

Only tags for the "DM", "I/O", "HR", "AR", and "LR" file types are allowed for use as a trigger tag for discrete alarms.

Configuring discrete alarms

Use arrays for discrete alarms and append each individual alarm to one bit of the array tags themselves and not to the individual subelements.

Only tags for the "DM", "I/O", "HR", "AR", LR" areas and the "Int" and "UInt" file types are permitted for discrete alarms and arrays.

Acknowledgment of alarms (Basic Panels)

Procedure

Create suitable tags on the PLC to acknowledge an error alarm. You assign these tags to an alarm in the "Bit messages" editor. You make the assignment in "Properties > Acknowledgment".

Distinction in terms of acknowledgment:

- Acknowledgment by the PLC
- Acknowledgment on the HMI device

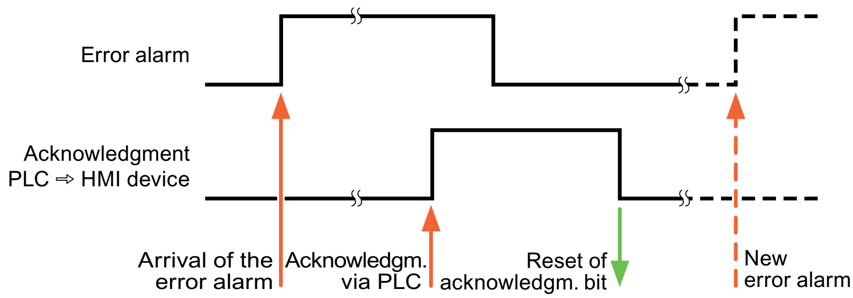
Acknowledgment by the PLC

In "PLC acknowledgment tag", configure the tag or array tag and the bit number that the HMI device uses to identify a PLC acknowledgment.

A bit set in the tag triggers acknowledgment of the assigned error alarm bit at the HMI device. This tag bit returns a function similar to acknowledgment on the HMI device which is triggered by pressing the "ACK" button, for example.

The acknowledgment bit must be located in the same tag as the bit for the error alarm.

Reset the acknowledgment bit before setting the bit in the alarm area again. The figure below shows the pulse diagram.



Acknowledgment on the HMI device

In the "HMI acknowledgment tag" area, configure the tag or array tag as well as the bit number that the HMI device writes to the PLC after acknowledgment. Make sure when you use an array tag that it is not longer than 6 words.

To always create a signal change when setting an assigned acknowledgment bit of a discrete alarm that must be acknowledged, the HMI device will reset the acknowledgment bit assigned to the alarm as soon as it detects an alarm subject to acknowledgment and write the acknowledgment tag in the PLC. There will be a certain delay between detecting the message and writing the acknowledgment tag in the PLC because the HMI device has to process the operations.

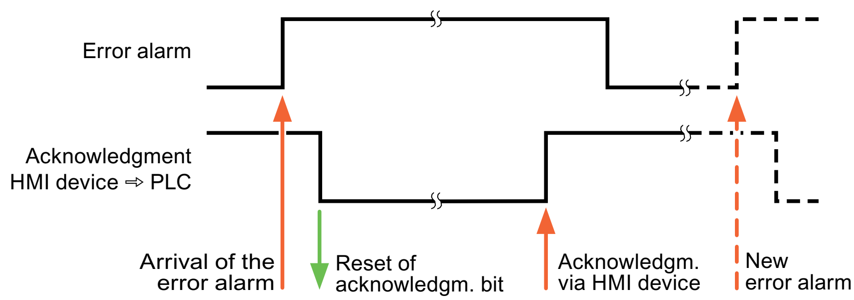
If a discrete alarm subject to acknowledgment is acknowledged by the HMI device, then the corresponding bit in the assigned acknowledgment tag will be set. The entire acknowledgment tag is then written to the PLC by the HMI device. This allows the PLC to recognize that a certain alarm message has been acknowledged at the HMI device.

Note

All alarm bits acknowledged since the last Runtime start will remain in the acknowledgment tag until a new incoming of the respective discrete alarms is detected.

This area should only be read by the PLC because the entire section of the HMI device will be overwritten once the next acknowledgment tag is written.

The figure below shows the pulse diagram.



1.13.3.5 Communication Service Packages (Basic Panels)

CSP_1 (Basic Panels)

Parallel communication (Basic Panels)

Parallel communication of communication drivers

The following table shows an overview of which communication drivers you can use simultaneously on one HMI device.

Note

Parallel communication is not approved for Basic Panels.

Parallel communication over Ethernet interfaces

The approved combinations can be operated via the same Ethernet interface. Several Ethernet interfaces are not required.

Parallel communication only concerns the Ethernet-based communication drivers.

	Allen-Bradley Ether-Net/IP	Mitsubishi MC TCP/IP	Modicon Modbus TCPIP	OPC (DA/ XML DA)	OPC UA (DA)	SIMATIC LOGO !	SIMATIC S7 200	SIMATIC S7 300/400	SIMATIC S7 1200	SIMATIC S7 1500	SIMATIC HTTP protocol	Sinumerik NC
Allen-Bradley Ether-Net/IP	--	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Mitsubishi MC TCP/IP	No	--	No	Yes	Yes	No	No	No	No	No	Yes	No

	Allen-Bradley Ether-Net/IP	Mitsu-bishi MC TCP/IP	Modi-con Mod-bus TCPIP	OPC (DA/ XML DA)	OPC UA (DA)	SI-MAT-IC LOGO !	SI-MAT-IC S7 2 00	SIMAT-IC S7 3 00/400	SIMAT-IC S7 1 200	SIMAT-IC S7 1500	SIMAT-IC HTTP proto-col	Sinu-merik NC
Modi-con Mod-bus TCPIP	No	No	--	Yes	Yes	No	No	No	No	No	Yes	No
OPC (DA/ XML DA)	Yes	Yes	Yes	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OPC UA (DA)	Yes	Yes	Yes	Yes	--	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SIMAT-IC LOGO!	Yes	No	No	Yes	Yes	--	Yes	Yes	Yes	Yes	Yes	Yes
SIMAT-IC S7 2 00	Yes	No	No	Yes	Yes	Yes	--	Yes	Yes	Yes	Yes	Yes
SIMAT-IC S7 3 00/400	Yes	No	No	Yes	Yes	Yes	Yes	--	Yes	Yes	Yes	Yes
SIMAT-IC S7 1 200	Yes	No	No	Yes	Yes	Yes	Yes	Yes	--	Yes	Yes	Yes
SIMAT-IC S7 1500	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes	Yes
SIMAT-IC HTTP proto-col	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	--	Yes
Sinu-merik NC	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	--

Parallel communication over serial interfaces

The following applies for parallel communication over serial interfaces:

- One communication driver per interface.
- One interface per communication driver.

CSP_2 (Basic Panels)

CSP_3 (Basic Panels)

CSP_4 (Basic Panels)

CSP_5 (Basic Panels)

1.13.4 Data exchange using area pointers (Basic Panels)

1.13.4.1 General information on area pointers (Basic Panels)

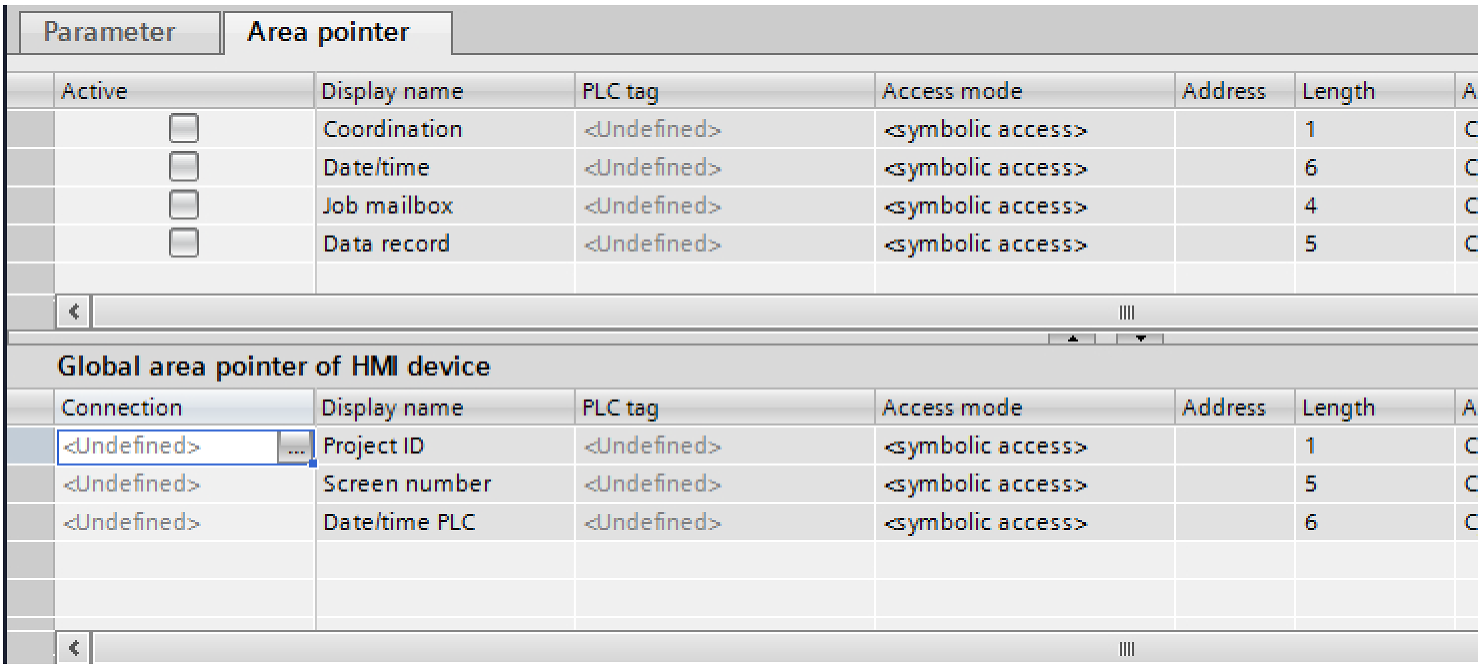
Introduction

You use an area pointer to access a data area in the PLC. During communication, the PLC and the HMI device alternately access these data areas for read and write operations.

The PLC and the HMI device trigger defined interactions based on the evaluation of stored data.

Configuration of area pointers

Before you use the area pointer, you enable it in "Connections ▸ Area pointers". You then assign the area pointer parameters.



- Active
  Enables the area pointer.
- Pointer name
  Name of the area pointer specified by WinCC.
- PLC tag
  Here you select the PLC tag or the tag array that you have configured as the data area for the area pointer.
- Address
  No address is entered into this field because of the symbolic access.
- Length
  WinCC specifies the length of the area pointer.
- Acquisition cycle
  You specify the acquisition cycle in this field for area pointers that are read by the HMI device. Note that a very short acquisition time may have a negative impact on HMI device performance.
- Comment
  Enter a comment, for example, to describe the purpose of the area pointer.

1.13.4.2 Accessing data areas (Basic Panels)

Accessing data areas

The following table shows how HMI devices and PLCs access individual data areas for read (R) or write (W) operations.

Data area	Required for	HMI device	PLC
Screen number	Evaluation by the PLC in order to determine the active screen.	W	R
Data record	Transfer of data records with synchronization	R/W	R/W
Date/time	Transfer of the date and time from the HMI device to the PLC	W	R
Date/time PLC	Transfer of the date and time from the PLC to the HMI device	R	W
Coordination	Requesting the HMI device status in the PLC program	W	R
Project ID	Runtime checks for consistency between the WinCC project ID and the project in the PLC	R	W
Job mailbox	Triggering of HMI device functions by the PLC program	R/W	R/W

1.13.4.3 "Screen number" area pointer (Basic Panels)

Function

The HMI device saves information about the screen called on the HMI device to the "Screen number" area pointer.

This allows the transfer of the current screen contents from the HMI device to the PLC. The PLC can trigger specific reactions such as the call of a different screen.

Use

Configure and enable the area pointer in "Communication > Connections" before you put it into use. You can create only **one** instance of the "Screen number" area pointer and only on **one** PLC.

The screen number is always transferred to the PLC when a new screen is activated or when the focus within a screen changes from one screen object to another.

Structure

The area pointer is a data area in the memory of the PLC with a fixed length of 5 words.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1. word	Current screen type															
2. word	Current screen number															
3. word	Reserved															
4th word	Current field number															
5. word	Reserved															

- Current screen type
  "1" for root screen or
  "4" for permanent window
- Current screen number
  1 to 32767
- Current field number
  1 to 32767

1.13.4.4 "Date/time" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the HMI device to the PLC.

The PLC writes control job "41" to the job mailbox.

When it evaluates the control job, the HMI device writes its current date and the time in the data area configured in the "Date/time" area pointer. All definitions are coded in BCD format.

The "Date/Time" area pointer when used in a project which contains multiple connections must be enabled for each configured connection.

Note

You cannot use the "Date/Time PLC" area pointer if you have configured the "Date/Time" area pointer.

The date/time data area has the following structure:

Data word	Left byte	Right byte	
	15 … 8	7 … 0	
n+0	Reserved	Hour (0 to 23)	Time
n+1	Minute (0 to 59)	Second (0 to 59)	
n+2	Reserved	Reserved	
n+3	Reserved	Weekday (1 to 7, 1=Sunday)	Date
n+4	Day (1 to 31)	Month (1 to 12)	
n+5	Year (80 to 99/0 to 29)	Reserved	

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

1.13.4.5 "Date/time PLC" area pointer (Basic Panels)

Function

This area pointer is used to transfer the date and time from the PLC to the HMI device. Use this area pointer if the PLC is the time master.

The PLC loads the data area of the area pointer. All definitions are coded in BCD format.

The HMI device reads the data cyclically within the configured acquisition cycle and synchronizes itself.

Note

Set an acquisition cycle of sufficient length for the Date/time area pointer in order to avoid any negative impact on HMI device performance.
Recommended: Acquisition cycle of 1 minute if your process can handle it.

"Date/Time PLC" is a global area pointer and may be configured only once per project.

Note

You cannot use the "Date/Time" area pointer if you have configured the "Date/Time PLC" area pointer.

The "Date/time PLC" data area has the following structure:

<table>
<tr><th rowspan="2">Data word</th><th colspan="3">Left byte</th><th colspan="3">Right byte</th></tr>
<tr><th>15</th><th>......</th><th>8</th><th>7</th><th>......</th><th>0</th></tr>
<tr><td>n+0</td><td colspan="3">Year (80 to 99/0 to 29)</td><td colspan="3">Month (1 to 12)</td></tr>
<tr><td>n+1</td><td colspan="3">Day (1 to 31)</td><td colspan="3">Hour (0 to 23)</td></tr>
<tr><td>n+2</td><td colspan="3">Minute (0 to 59)</td><td colspan="3">Second (0 to 59)</td></tr>
<tr><td>n+3</td><td colspan="3">Reserved</td><td colspan="2">Reserved</td><td>Weekday (1 to 7, 1=Sunday)</td></tr>
<tr><td>n+4 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
<tr><td>n+5 1)</td><td colspan="3">Reserved</td><td colspan="3">Reserved</td></tr>
</table>

1) The two data words must exist in the data area to ensure that the data format matches WinCC and to avoid reading false information.

Note

When making entries in the "Year" data area, you should note that values 80 to 99 result in years 1980 through 1999, while the values 0 to 29 result in the years 2000 through 2029.

1.13.4.6 "Coordination" area pointer (Basic Panels)

Function

The "Coordination" area pointer is used to implement the following functionality:

- Detecting the startup of the HMI device in the control program
- detection in the control program of the current HMI device operating mode
- detection in the control program of the HMI devices ready to communicate state

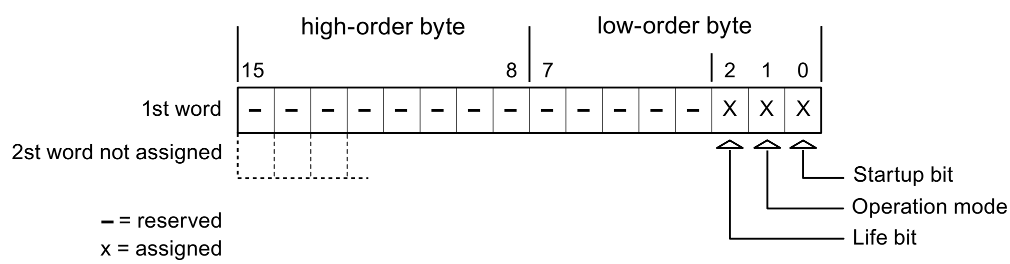
The "Coordination" area pointer has a length of one word.

Use

Note

The HMI device always writes the entire coordination area when updating the area pointer. The control program may not make changes to the coordination area for this reason.

Assignment of bits in the "Coordination" area pointer



Startup bit

The startup bit is set briefly to "0" by the HMI device during startup. It sets the bit permanently to "1" when startup is completed.

Operating mode

The operating mode bit is set to 1 as soon as the user switches the HMI device offline. The state of the operating mode bit is "0" during normal operation of the HMI device. You can determine the current operating mode of the HMI device by reading this bit.

Life bit

The HMI device inverts the life bit at intervals of approximately one second. You can check whether or not the connection to the HMI device is still up by querying this bit in the control program.

1.13.4.7 "Project ID" area pointer (Basic Panels)

Function

You can check whether the HMI device is connected to the correct PLC at the start of Runtime. This check is important when operating with several HMI devices.

The HMI device compares a value stored on the PLC with the value specified in configuration. This ensures compatibility of configuration data with the control program. If discrepancy is detected, a system event is displayed on the HMI device and Runtime is stopped.

Application

To use this area pointer, set up the following during the configuration:

- Define the version of configuration. Possible values between 1 and 255.
  You enter the version in the editor "Runtime settings > General" in the "Identification" area.
- Data address of the value for the version that is stored in the PLC:
  You enter the data address in the editor "Communication > Connections".

Connection failure

A connection failure to a device on which the "Project ID" area pointer is configured results in all the other connections of the device being switched to "offline".

This behavior has the following prerequisites:

- You have configured several connections in a project.
- You are using the "project ID" area pointer in at least one connection.

Causes which may set connections "offline":

- The PLC is not available.
- The connection has been switched offline in the engineering system.

1.13.4.8 "PLC job" area pointer (Basic Panels)

Function

The PLC can use the job mailbox to transfer jobs to the HMI device to trigger corresponding actions on the HMI device. These functions include, for example:

- Display screen
- Set date and time

Data structure

The first word of the job mailbox contains the job number. Depending on the job mailbox, up to three parameters can be transferred.

Word	Left byte	Right byte
n+0	0	Job number
n+1	Parameter 1	
n+2	Parameter 2	
n+3	Parameter 3	

The HMI device evaluates the job mailbox if the first word of this job is not equal to zero. This means that the parameters must be entered in the job mailbox first, followed by the job number.

When the HMI device accepts the job mailbox, the first word is set to 0 again. The execution of the job mailbox is generally not completed at this point in time.

Job mailboxes

All job mailboxes and their parameters are listed below. The "No." column contains the job number of the job mailbox. Job mailboxes can only be triggered by the PLC when the HMI device is online.

Note

Please note that not all HMI devices support job mailboxes.

No.	Function	
14	**Setting the time (BCD coded)**	
	Parameter 1	Left byte: - Right byte: hours (0-23)
	Parameter 2	Left byte: minutes (0-59) Right byte: seconds (0-59)
	Parameter 3	-
15	**Setting the date (BCD coded)**	
	Parameter 1	Left byte: - Right byte: weekday (1-7: Sunday-Saturday)
	Parameter 2	Left byte: day (1-31) Right byte: month (1-12)
	Parameter 3	Left byte: year
23	**User logon**	
	Logs the user on with the name "PLC user" at the HMI device with the group number transferred in Parameter 1. The logon is possible only when the transferred group number exists in the project.	
	Parameter 1	Group number 1 to 255
	Parameter 2, 3	-

No.	Function	
14	**Setting the time (BCD coded)**	
24	**User logoff**	
	Logs off the current user. (The function corresponds to the "logoff" system function)	
	Parameter 1, 2, 3	-
40	**Transfer date/time to PLC**	
	(in the S7 format DATE_AND_TIME) An interval of at least 5 seconds must be maintained between two successive jobs to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
41	**Transfer date/time to PLC**	
	(In OP/MP format) An interval of at least 5 seconds must be maintained between successive jobs in order to prevent overload of the HMI device.	
	Parameter 1, 2, 3	-
46	**Update tags**	
	Causes the HMI device to read the current value of the tags•from the PLC whose update ID matches the value transferred in parameter 1. (Function corresponds to the "UpdateTag" system function.)	
	Parameter 1	1 - 100
49	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Warnings" class from the alarm buffer.	
	Parameter 1, 2, 3	-
50	**Delete alarm buffer**	
	Deletes all analog alarms and discrete alarms of the "Errors" class from the alarm buffer.	
	Parameter 1, 2, 3	-
51	**Screen selection** [1)]	
	Parameter 1	Screen number
	Parameter 2	-
	Parameter 3	Field number
69	**Read data record from PLC**	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	0: Do not overwrite existing data record 1: Overwrite existing data record
70	**Write data record to PLC**	
	Parameter 1	Recipe number (1-999)
	Parameter 2	Data record number (1-65535)
	Parameter 3	-

1) OP 73, OP 77A and TP 177A HMI devices also execute the "Screen selection" job mailbox if the on-screen keyboard is active.

1.13.4.9 "Data record" area pointer (Basic Panels)

"Data mailbox" area pointer (Basic Panels)

Function

When data records are transferred between the HMI device and PLC, both partners access common communications areas on the PLC.

Data transfer types

There are two ways of transferring data records between the HMI device and PLC:

- Transfer without synchronization
- Transfer with synchronization over the data record

Data records are always transferred directly. That is, the tag values are read from an address or written to an address configured for this tag directly, without redirecting the values by means of interim memory.

Initiating the transfer of data records

There are three ways of triggering the transfer:

- Operator input in the recipe view
- Job mailboxes
  The transfer of data records can also be triggered by the PLC.
- Triggering by configured functions

If the transfer of data records is triggered by a configured function or by a job mailbox, the recipe view on the HMI device remains operable. The data records are transferred in the background.

Simultaneous processing of several transfer requests is, however, not possible. In this case, the HMI device rejects the other transfer requests with a system event.

Transfer without synchronization (Basic Panels)

If you select asynchronous transfer of data records between the HMI device and PLC, there is no coordination over the common data areas. It is therefore unnecessary to set up a data area during configuration.

Asynchronous data record transfer can be a useful alternative, for example, when:

- The system is capable of excluding the risk of uncontrolled overwriting of data by the communication peer.
- The PLC does not require information about the recipe number and data record number.
- The transfer of data records is triggered by the operator of the HMI device.

Reading values

When a read job is triggered, the values are read from the PLC addresses and transferred to the HMI device.

- Triggering by the operator in the recipe view:
  The values are downloaded to the HMI device. You can then process, edit, or save these values, for example.
- Triggering by a function or job mailbox:
  The values are saved immediately to the data volume.

Writing values

When a write job is triggered, the values are written to the PLC addresses.

- Triggering by the operator in the recipe view:
  The current values are written to the PLC.
- Triggering by a function or job mailbox:
  The current values are written to the PLC from the data medium.

Transfer with synchronization (Basic Panels)

If you select synchronous transfer, both communication partners set status bits in the common data area. You can use this mechanism to prevent uncontrolled overwriting of data in either direction in your control program.

Application

Synchronous data record transfer can be a useful solution, for example, when:

- The PLC is the "active partner" in the transfer of data records.
- The PLC evaluates the information about the recipe number and data record number.
- The transfer of data records is triggered by means of a Job mailbox.

Requirements

In order to synchronize transfer of data records between the HMI device and the PLC, the following requirements must be met during configuration:

- An area pointer has been set up: "Communication > Connections" editor in "Area pointer".
- The PLC with which the HMI device synchronizes transfer of data records is specified in the recipe:
  "Recipes" editor in the inspector window the option "Coordinated transfer of data records" under "General > Synchronization > Settings"

Structure of the data area

The data area has a fixed length of 5 words. Structure of the data area:

	15		0
1. Word	Current recipe number (1 - 999)		
2. Word	Current data record number (0 - 65535)		
3. Word	Reserved		
4. Word	Status (0, 2, 4, 12)		
5. Word	Reserved		

- Status
  The status word (word 4) can adopt the following values:

Value		Meaning
Decimal	Binary	
0	0000 0000	Transfer permitted, data record free
2	0000 0010	Transferring.
4	0000 0100	Transfer completed without error
12	0000 1100	Transfer completed with error

Sequence of a transfer started by the operator in the recipe display (Basic Panels)

Reading from the PLC started by the operator in the recipe view

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe number to be read and the status "Transferring" in the data record and sets the data record number to 0.	Abort with system event.
3	The HMI device reads the values from the PLC and displays them in the recipe view. If the recipes have synchronized tags, the values from the PLC are also written to the tags.	
4	The HMI device sets the status "Transfer completed."	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC started by the operator in the recipe view

Step	Action	
	Check: Status word = 0?	
1	Yes	No
	The HMI device enters the recipe and data record number to be written and the status "Transferring" in the data record.	Abort with system event.
2	The HMI device writes the current values to the PLC. If the recipes have synchronized tags, the changed values are synchronized between the recipe view and tags and then written to the PLC.	
3	The HMI device sets the status "Transfer completed."	
4	If required, the control program can now evaluate the transferred data.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Sequence of the transfer triggered by a PLC job (Basic Panels)

The transfer of data records between the HMI device and the PLC can be initiated by either one of these stations.

The two job mailboxes No. 69 and No. 70 are available for this type of transfer.

No. 69: Read data record from PLC ("PLC → DAT")

Job mailbox no. 69 transfers data records from the PLC to the HMI device. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	69
Word 2	Recipe number (1-999)	
Word 3	Data record number (1 to 65535)	
Word 4	Do not overwrite existing data record: 0 Overwrite existing data record: 1	

No. 70: Write data record to PLC ("DAT → PLC")

Job mailbox No. 70 transfers data records from the HMI device to the PLC. The job mailbox is structured as follows:

	Left byte (LB)	Right byte (RB)
Word 1	0	70
Word 2	Recipe number (1-999)	
Word 3	Data record number (1 to 65535)	
Word 4	—	

Sequence when reading from the PLC with job mailbox "PLC → DAT" (no. 69)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data record.	Abort without return message.
3	The HMI device reads the values from the PLC and saves these to the data record defined in the job mailbox.	
4	• If "Overwrite" was selected in the job, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "Do not overwrite" was selected in the job, and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data record.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Sequence of writing to the PLC with job mailbox "DAT → PLC" (no. 70)

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the job and the status "Transferring" in the data record.	Abort without return message.
3	The HMI device fetches the values of the data record specified in the function from the data medium and writes the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Sequence of the transfer when triggered by a configured function (Basic Panels)

Reading from the PLC using a configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data record.	Abort with system event.
3	The HMI device reads the values from the PLC and stores them in the data record specified in the function.	
4	• If "Yes" was selected for the "Overwrite" function, an existing data record is overwritten without any prompt for confirmation. The HMI device sets the status "Transfer completed." • If "No" was selected for the "Overwrite" function and the data record already exists, the HMI device aborts the job and enters 0000 1100 in the status word of the data record.	
5	The control program must reset the status word to zero in order to enable further transfers.	

Writing to the PLC by means of configured function

Step	Action	
1	Check: Status word = 0?	
	Yes	No
2	The HMI device enters the recipe and data record number specified in the function and the status "Transferring" in the data record.	Abort with system event.
3	The HMI device fetches the values of the data record specified in the function from the data medium and transfers the values to the PLC.	
4	The HMI device sets the status "Transfer completed."	
5	The control program can now evaluate the transferred data. The control program must reset the status word to zero in order to enable further transfers.	

Possible causes of error when transferring data records (Basic Panels)

Possible causes of error

The section below shows possible error causes which lead to the cancellation of data record transfer:

- Tag address not set up on the PLC
- Overwriting data records not possible

- Recipe number does not exist
- Data record number does not exist

Note

The status word may only be set by the HMI device. The PLC may only reset the status word to zero.

Note

The PLC may only evaluate the recipe and data record numbers when data inconsistency is detected if one of the conditions outlined below has been met:

- The data mailbox status is set to "Transfer completed".
- The data mailbox status is set to "Transfer completed with error".

Reaction to an aborted transfer due to errors

If the transfer of data records is aborted due to errors, the HMI device reacts as follows:

- Triggering by the operator in the recipe view
  Information in the status bar of the recipe view and output of system alarms
- Triggered by function
  Output of system alarms
- Triggering by job mailbox
  No return message on the HMI device

You can nonetheless evaluate the status of the transfer by querying the status word in the data record.

Index

A

Absolute Addressing
of a tag, 51
Access level, 77
Accessible devices, 83, 136, 190, 249, 268
Acquisition cycle
Area pointer, 56
Active
Area pointer, 56
Address
Area pointer, 56
Mitsubishi, 387
Omron Host Link, 443
Addressing
Allen-Bradley, 345, 364
Allen-Bradley Ethernet IP, 340
Ethernet/IP, 345, 364
MPI, 206
Valid, 346
Alarm message
Acknowledgment by the PLC, 110, 165, 222, 316, 328, 374, 402, 431, 447
Acknowledgment on the HMI device, 111, 166, 223, 316, 329, 375, 402, 431, 448
Configure acknowledgment, 110, 165, 222, 316, 328, 374, 402, 431, 447
Alarms
Configuring, 109, 164, 221, 315, 327
Data types, 109, 164, 221, 315, 327
Modicon Modbus, 430
Non-integrated connection, 430
Restriction, 401
Allen-Bradley, 331, 354, 355
Allen-Bradley DF1 communication driver, 354, 355
Analog alarm, 373
Area pointer, 371
Basic Panel, 63
Communication drivers, 333
Data type, 373
DF1, 333
EtherNet/IP, 333, 346
Mitsubishi, 401
Allen-Bradley DF1
Configuring a connection, 350
Connection, 350, 354
Connection parameters, 351
CPU type, 364
KF2 module, 354, 355
KF3 module, 354, 355
Valid data type, 362
Allen-Bradley Ethernet IP
Address multiplexing, 344
Addressing, 340
Addressing type, 342
Allen-Bradley EtherNet/IP
Configuring a connection, 333
Connection, 333, 337
Connection parameters, 335
Data type, 338
Analog alarm
Allen-Bradley, 373
Mitsubishi, 401
Omron, 446
Area pointer, 54, 93, 147, 207, 300, 451
Acquisition cycle, 56
Active, 56
Address, 56
Allen-Bradley, 371
Basic Panel, 66
Comment, 56
Connection, 55, 60
Connections editor, 55
Coordination, 456
Data record, 102, 158, 214, 308, 460
Date/time, 148, 150
Job mailbox, 100, 155, 212, 306, 457
Length, 56
PLC tag, 56
Pointer name, 56
Project ID, 99, 154, 211, 305, 457
Screen number, 94, 149, 153, 207, 301, 453
Tab, 56
Trends, 325
Area pointers
Configuring, 58
Coordination, 98, 152, 210, 304
Creating, 58
Date/time, 95, 208, 301, 454
Date/time PLC, 96, 209, 302, 455
Length, 59
Mitsubishi, 399
Modicon Modbus, 428
Omron, 445
Asynchronous
Transferring data, 102, 159, 215, 313, 460

Automation system, 10, 33
Setting up, 10
Autonegotiation
Disabled, 82, 135, 189, 248, 267

B

Basic Panel
Area pointer, 66
Communication drivers, 63
ETHERNET, 65
IF1B, 65
Interface, 65
Mitsubishi, 63
Modicon Modbus, 63
Omron, 63
OPC, 63
Bit-triggered trends, 325
Boundaries, 83, 136, 190, 249, 268

C

Cabling rules (PROFINET), 82, 135, 189, 248, 267
Change word order
Modicon MODBUS TCP/IP, 409
CJ1, 442
CJ2, 442
Comment
Area pointer, 56
Communication, 9
Basics, 10
Definition, 9
ET 200 CPU, 255
S7 1200, 119
S7 1500, 67
S7 200, 290
S7 300, 177
S7 400, 177
SIMATIC LOGO!, 319
SIMATIC S7-1500 Software Controller, 237
Third-party drivers, 331
using area pointers, 14
Communication drivers, 14
Allen-Bradley, 333
Basic Panel, 63
Mitsubishi, 375
Modicon Modbus, 403
Omron, 432
Communication network, 33
PROFINET, 34
Communication partners
HMI device, 9
Networking, 37
PLC, 9
SIMATIC S7, 118, 177, 235, 288, 297, 300
Communication types
Allen-Bradley, 337
Connectable PLCs, 411, 422
Enabled, 381, 394, 410, 421, 438
Ethernet/IP, 337
Restriction, 410, 421
Compact, 413, 426
CompactLogix, 340
Components
Commissioning, 348, 370, 389, 398, 415, 427, 444
Configure acknowledgment
Alarm message, 110, 165, 222, 316, 328, 374, 402, 431, 447
Configure alarms
Allen-Bradley, 373
Data types, 401
Non-integrated connection, 400
Special considerations, 401
Configuring
Alarms, 109, 164, 315, 327
Connections, 225, 228, 229, 290, 320
Configuring a connection
Allen-Bradley DF1, 350
Allen-Bradley EtherNet/IP, 333
Mitsubishi FX, 390
Mitsubishi MC TCP/IP, 376
Modicon Modbus RTU, 416
Modicon Modbus TCP, 404
Omron Host Link, 433
Configuring the network with Ethernet, 128, 186, 264
Creating private subnets, 129, 187, 265
Linking networks, 129, 187, 265
Relationship between IP address and subnet mask, 129, 187, 264
Setting the IP address, 128, 186, 264
Setting the subnet mask, 128, 186, 264
Connecting
PLC, 337, 380, 395, 409, 437
Connecting cable
6XV1440 - 2P for Mitsubishi PG protocol, 396
Allen-Bradley cable 1784-CP10, 359
Mitsubishi FX, 395
Connecting cable 9-pole
Sub D, RS 422, 358

Connecting cables
Point-to-point cable 1,
Point-to-point cable 2, 424
Connection, 13
Allen-Bradley DF1, 350, 354
Allen-Bradley EtherNet/IP, 333
Area pointer, 55, 60
Configuring, 38, 41, 44, 333, 350, 376, 390, 404, 416, 433
Creating, 38, 41, 44
Highlighting, 40
Integrated, 38
integrated connection, 51
Mitsubishi MC TCP/IP, 376
Modicon, 420
Modicon Modbus RTU, 416
Modicon Modbus TCP, 404
offline, 99, 155, 212, 305
Omron communication drivers, 445
Omron Host Link, 433
Parameters, 335, 351, 378, 392, 406, 418, 435
Password, 80, 92, 133, 146, 245
Routed, 44
S7 200, 290, 320
Table, 37
Connection cable
Allen-Bradley cable 1747-CP3, 360
Allen-Bradley cable 1761-CBL-PM02, 361
Modicon, 421
Multipoint cable 1:MP/TP/PC, 439
Multipoint cable 2:RS422, MP/TP/PC, 440
Omron Host Link, 437
Point-to-point cable PP2 for Omron, 440, 441
Connection information
Diagnostics, 21
Connection parameters
Allen-Bradley DF1, 351
Allen-Bradley EtherNet/IP, 335
Mitsubishi FX, 392
Mitsubishi MC TCP/IP, 378
Modicon Modbus RTU, 418
Modicon Modbus TCP, 406
Omron Host Link, 435
Connections
Allen-Bradley DF1, 354, 355
Configuring, 225, 228, 229, 290, 320
Editor, 225, 228, 229
ControlLogix, 340
Coordination, 14
CP1, 442
CPM, 442
CPU type
Allen-Bradley DF1, 364
Allen-Bradley EtherNet/IP, 340
FX3 series, 386
Mitsubishi FX, 397
Mitsubishi MC TCP/IP, 386
Modicon Modbus RTU, 426
Modicon Modbus TCP, 413
Omron Host Link, 442
Create
Connection, 120, 256
Creating
Connection, 38, 41, 44, 69, 179, 239
CS1, 442
Cyclic operation, 237, 292, 324

D

Data areas
Area pointer, 58, 453
Reading, 58, 453
Writing, 58, 453
Data exchange, 9
Tags, 13
Trends, 325
using area pointers, 14
Data record, 14
Data type
Allen-Bradley, 373
Allen-Bradley EtherNet/IP, 338
Discrete alarm, 373, 401, 446
Internal tag, 53
Mitsubishi, 401
Mitsubishi FX, 396
Mitsubishi MC TCP/IP, 385
Modicon Modbus RTU, 425
Modicon Modbus TCP, 412
Omron, 446
Omron Host Link, 441
Valid, 338, 385, 396, 412, 425, 441
Data types
ET 200 CPU, 282
S7 1500, 113
S7 200, 318
S7 300/400, 225
S7-1200 V1, 168
Trends, 109, 327
Valid, 113, 168, 225, 282, 318
Date/time, 14
Area pointer, 148
Date/time PLC, 14

Device
Inserting, 11
Device dependency
S7 200, 317
S7-1200 V1, 167
S7-1200 V2, 167
S7-1200 V3, 167
S7-1200 V4, 167
S7-300/400, 224
SIMATIC LOGO!, 329
SIMATIC S7-1500 V1.0, 112, 249
Device information
Diagnostics, 21
Device view
Hardware and network editor, 24
Devices
Connecting, 13
Networking, 13, 37
Devices & Networks, 13, 37
Devices and networks, 13, 37
Connections, 38, 44
HMI connections, 38, 44
Diagnostics
Connection, 21
Connection information, 21
Device, 21
Device information, 21
Discrete alarm
Allen-Bradley, 373
Mitsubishi, 401
Omron, 446
DP, 145, 198

E

Editor
Connection, 41
Connections, 225, 228, 229
Devices and networks, 40
Enable autonegation, 81, 134, 188, 247, 266
End of discovery of accessible devices, 83, 136, 190, 249, 268
End of sync domain, 83, 136, 190, 249, 268
End of topology discovery, 83, 136, 190, 249, 268
ET 200 CPU
Communication, 255
Data types, 282
HMI connection, 256
Ethernet
Basic Panel,
Parameters, 116, 175, 233, 289, 295, 323
EtherNet/IP
Allen-Bradley, 346
External tags
Data exchange, 13

F

FAQs, 32
Functions
Updating tag value, 51
FX1 series, 397
FX2 series, 397
FX3
PLC, 382
FX3 series, 386

G

Global data block
Configuring, 59
Creating, 59

H

Hardware and network editor
Device view, 24
Network view, 17
Topology view, 26
Hardware catalog
Task card, 30
Hardware editor
Components, 15
Function, 15
Hardware catalog, 30
Inspector window, 28
Highlighting
Connection, 40
HMI connection, 13
Creating, 69, 120, 179, 239, 256
ET 200 CPU, 256
MPI, 199
MPI parameters, 201
Password, 80, 92, 133, 146, 245
PROFIBUS, 85, 192
PROFIBUS parameters, 88, 140, 194, 271
PROFINET, 68, 69, 119, 120, 178, 179, 238, 239, 255, 256
PROFINET parameters, 71, 123, 181, 240, 258
S7 1200, 120
S7 1500, 69

S7 300/400, 179
SIMATIC S7-1500 Software Controller, 239
HMI connections
Devices & Networks, 38, 44
HMI device
MPI parameters, 203
PROFIBUS parameters, 89, 142, 195, 273
PROFINET parameters, 72, 124, 182, 242, 260
HMI device on a PLC
Commissioning, 348, 370, 389, 398, 415, 427, 444

I

IF1B
Basic Panel, 65
Industrial Ethernet, 33
Inspector window
Hardware and network editor, 28
Integrated
Connection, 38, 41
Integrated connection, 51
Interface
Basic Panel, 65
Internal
Ethernet port, 385
Internal Ethernet port
Open settings, 385
Internal tags
Data exchange, 13
IP protocol, 126, 262
iQ series, 386

J

Job mailbox, 14
Transferring data, 104, 160, 218, 310, 463

L

Length
Area pointer, 56
Area pointers, 59
LLDP (Link Layer Discovery Protocol), 83, 136, 190, 249, 268

M

Manuals, 32
Micro, 413
MicroLogix, 340, 364
Mitsubishi, 331
Address, 387
Analog alarm, 401
Basic Panel, 63
Communication drivers, 375
FX, 375
TCP/IP, 375
Mitsubishi FX
Configuring a connection, 390
Connection, 390, 395
Connection parameters, 392
CPU type, 397
Data type, 396
Mitsubishi MC TCP/IP, 381, 394
Mitsubishi MC TCP/IP
Configuring a connection, 376
Connection, 376, 380
Connection parameters, 378
CPU type, 386
Data type, 385
Modicon
Approved communication with Modbus RTU, 421
Connection, 420
Connection cable, 421
Restrictions with Modbus RTU, 421
Modicon M340, 413
Modicon Modbus, 331
Basic Panel, 63
Communication drivers, 403
RTU, 403
TCP, 403
Modicon Modbus connection
Data types, 430
Modicon Modbus RTU, 426
Configuring a connection, 416
Connection, 416
Connection parameters, 418
Data type, 425
Modicon Modbus TCP, 413
Configuring a connection, 404
Connection, 404, 409
Connection parameters, 406
Data type, 412
Modicon MODBUS TCP/IP
Change word order, 409
Momentum, 413, 426
Monitor, 81, 134, 188, 247, 266
Moving the view
Overview navigation, 18, 25, 27
MPI, 33
Addressing, 206

Connection, 229
HMI connection, 199
Network, 35
Network architecture, 35
Parameters, 201, 203, 205
S7 200, 235, 298
S7 300/400, 199
MPI address
S7 300, 206
S7 400, 206
MPI communication
S7 300/400, 199
MPI parameters
S7 300/400, 201, 203, 205
Multiplexing
Address multiplexing Allen-Bradley Ethernet IP, 344
Multi-point connection
Allen-Bradley DF1, 357, 358

N

Network, 9
Ethernet, 34
MPI, 35
PPI, 36
PROFIBUS, 34
PROFINET, 34
Network architecture
PPI, 36
Network editor
Components, 15
Function, 15
Hardware catalog, 30
Inspector window, 28
Network view, 37
Hardware and network editor, 17
Networking
Communication partners, 37
Devices, 37
Non-integrated
Connection, 41

O

Omron, 331
Analog alarm, 446
Basic Panel, 63
Communication drivers, 432
Host Link, 432
Omron Host Link, 442
Address, 443
Configuring a connection, 433
Connection, 433, 437
Connection cable, 437
Connection parameters, 435
Data type, 441
OP 73
Baud rate on PROFIBUS, 117, 176, 287
OP 77A
Baud rate on PROFIBUS, 117, 176, 287
OP 73
Baud rate on PROFIBUS, 234, 236, 296, 298, 299
OP 77A
Baud rate on PROFIBUS, 234, 236, 296, 298, 299
OPC
Basic Panel, 63
Operator input in the recipe view
Transferring data, 159, 217, 309, 462
Optimizing the configuration
Allen-Bradley DF1, 349, 370, 389, 398, 415, 427, 444
Other PLCs
Addressing, 332
Data types, 332
Special features, 332
Overview navigation, 18, 25, 27

P

Parallel communication
Communication drivers, 449
Parameter
SIMATIC LOGO!, 321
Parameters
Connection, 335, 351, 378, 392, 406, 418, 435
MPI, 201, 203, 205
PROFIBUS, 88, 89, 91, 140, 142, 144, 194, 195, 197, 271, 273, 274
S7 1200, 174
S7 1500, 114
S7 200, 231, 235, 293, 298, 299
Password
HMI connection, 80, 92, 133, 146, 245
Password protection, 77, 130, 132
Permitted data types
Trends, 109, 327
Pin assignment
6XV1440 - 2P for Mitsubishi PG protocol, 396
Allen-Bradley, 358

Allen-Bradley cable 1747-CP3, 360
Allen-Bradley cable 1761-CBL-PM02, 361
Allen-Bradley cable 1784-CP10, 359
Multipoint cable 1:MP/TP/PC, 439
Multipoint cable 2:RS422, MP/TP/PC, 440
Point-to-point cable 1,
Point-to-point cable 2, 424
Point-to-point cable PP1 for Omron, 440
Point-to-point cable PP2 for Omron, 441
PLC
Connecting, 337, 380, 395, 409, 437
MPI parameters, 205
PROFIBUS parameters, 91, 144, 197, 274
PROFINET parameters, 74, 126, 184, 243, 262
PLC5, 364
Pointer name
Area pointer, 56
Point-to-point connection
Allen-Bradley DF1, 356
Port options, 82, 135, 189, 248, 267
Enable autonegation, 81, 134, 188, 247, 266
Monitor, 81, 134, 188, 247, 266
Transmission rate / duplex, 81, 134, 188, 247, 266
Possible cause of error
Transferring data, 107, 163, 220, 312, 465
PPI, 33
Network, 36
Network architecture, 36
S7 200, 299
Premium, 413
Product support, 32
PROFIBUS, 33, 145, 198
HMI connection, 85, 138, 192, 269
Network, 34
OP 73, 117, 176, 287
OP 77A, 117, 176, 287
OP 73, 234, 236, 296, 298, 299
OP 77A, 234, 236, 296, 298, 299
Parameters, 88, 89, 91, 117, 140, 142, 144, 176, 194, 195, 197, 234, 271, 273, 274, 287, 296
S7 1200, 137
S7 1500, 84
S7 300/400, 85, 191, 192
S7-1200, 138
SIMATIC ET 200 CPU, 269
Standard, 145, 198
Universal, 145, 198
PROFIBUS communication
S7 1200, 137
S7 1500, 84
S7 300/400, 191
SIMATIC ET 200 CPU, 269
PROFIBUS DP, 34
Connection, 228
PROFIBUS parameters
S7 1200, 140, 142, 144, 271, 273, 274
S7 1500, 88, 89, 91
S7 300/400, 194, 195, 197
PROFIBUS profile, 145, 198
Different profiles on the same subnet, 145, 198
Effect on the transmission rate, 145, 198
Meaning of profiles, 145, 198
PROFINET, 33
Connection, 225
HMI connection, 68, 69, 119, 120, 178, 179, 238, 239, 255, 256
S7 1200, 119
S7 1500, 68
S7 300/400, 178
SIMATIC ET 200 CPU, 255
WinAC 1500, 238
PROFINET IO
Port options, 81, 134, 188, 247, 266
PROFINET parameters
HMI connection, 71, 123, 181, 240, 258
HMI device, 72, 124, 182, 242, 260
PLC, 74, 126, 184, 243, 262
S7 1200, 123, 126, 258, 262
S7 1500, 71, 72, 74, 240, 242, 243
S7 300/400, 181, 182, 184
Project
Multiple connections, 99, 155, 212, 305
Project ID, 14
Protection concept, 77, 131
Protection level, 77, 130, 132

Q

Q
PLC, 383
Q series, 386
Q0xUDEH CPU
PLC, 385
Quantum, 413

R

Read-only, 132
Restriction
Modicon Modbus RTU, 429
Modicon Modbus TCP/IP, 429

Routed
    Connection, 44

S

S7 1200
    Communication, 119
    Connection parameters, 174
    HMI connection, 120
    Parameters, 174
    PROFIBUS, 137
    PROFIBUS parameters, 140, 142, 144, 271, 273, 274
    PROFINET, 119
S7 1500
    Communication, 67
    Connection parameters, 114
    Data types, 113
    HMI connection, 69
    Parameters, 114
    PROFIBUS, 84, 85
    PROFIBUS parameters, 88, 89, 91
    PROFINET, 68, 238
    PROFINET parameters, 71, 72, 74, 240, 242, 243
S7 200
    Communication, 290
    Connection, 290, 320
    Connection parameters, 293
    Data types, 318
    MPI, 235, 298
    Parameters, 235, 293, 298, 299
    PPI, 299
S7 300
    Communication, 177
    MPI address, 206
S7 300/400
    Connection parameters, 231
    Data types, 225
    HMI connection, 179
    MPI, 199
    MPI parameters, 201, 203, 205
    Parameters, 231
    PROFIBUS, 191, 192
    PROFIBUS parameters, 194, 195, 197
    PROFINET, 178
    PROFINET parameters, 181, 182, 184
S7 400
    Communication, 177
    MPI address, 206
S7-1200
    PROFIBUS, 138
S7-1200 V2
    Data types, 168
Screen number, 14
Script
    Updating tag value, 51
Service & Support, 32
SIMATIC LOGO!
    Communication, 319
    Connection parameters, 321
    Parameter, 321
SIMATIC S7
    Communication partners, 118, 177, 235, 288, 297, 300
SIMATIC S7-1500 Software Controller
    Communication, 237
    HMI connection, 239
SIMATIC ET 200 CPU
    PROFIBUS, 269
    PROFINET, 255
SLC, 364
Special considerations
    Omron Host Link, 445
Standard
    PROFIBUS, 145, 198
Switch buffer, 108, 326
Switchport, 82, 135, 189, 248, 267
Symbolic addressing
    of a tag, 52
Sync domain, 83, 136, 190, 249, 268
Synchronize, 50
Synchronous
    Transferring data, 103, 216, 313, 461
Syntax
    Addressing, 342

T

Table
    Connection, 37
Tag
    absolute addressing, 51
    External tag, 49
    Internal tag, 53
    symbolic addressing, 52
Tag table
    Default, 48
    for HMI devices, 48
    user-defined, 48
Tags
    Basics, 47
    Data exchange, 13

Task card
Hardware catalog, 30
Third-party drivers
Communication, 331
Special features, 332
Time synchronization, 251
Configuring, 252, 253
integrated connection, 252
Non-integrated connection, 253
Restriction, 251
System limits, 251
Time-triggered trends, 325
Topology discovery, 83, 136, 190, 249, 268
Topology view
Hardware and network editor, 26
Transferring data
Area pointer, 54, 93, 147, 207, 300, 451
Area pointer screen number, 94, 149, 153, 207, 301, 453
Coordination area pointer, 98, 152, 210, 304, 456
Data record are pointer, 460
Data record area pointer, 102, 158, 214, 308
Date/time area pointer, 95, 148, 150, 208, 301, 454
Date/time PLC area pointer, 96, 209, 302, 455
Job mailbox, 104, 160, 218, 310, 463
Job mailbox area pointer, 100, 155, 212, 306, 457
Operator input in the recipe view, 159, 217, 309, 462
Possible cause of error, 107, 163, 220, 312, 465
Project ID area pointer, 99, 154, 211, 305, 457
Triggering by means of a configured function, 106, 162, 219, 311, 465
With synchronization, 103, 216, 313, 461
Without synchronization, 102, 159, 215, 313, 460
Transmission rate / duplex, 81, 134, 188, 247, 266
Trend request
Trend transfer, 108, 326
Trend request area, 108, 326
Trend transfer area, 108, 326
Trend transfer area 1
Trend transfer area 2, 108, 326
Triggering by means of a configured function
Transferring data, 106, 162, 219, 311, 465

U

Universal
PROFIBUS, 145, 198

V

Valid
Data type, 338, 385, 396, 412, 425, 441
Data types, 113, 168, 225, 282, 318
Valid data type
Allen-Bradley DF1, 362

Z

Zoom
Adjusting the zoom setting, 18, 25, 27