ibaPDA-Interface-S7-Xplorer

PLC-Xplorer Data Interface to SIMATIC S7 Systems

Manual

Issue 2.6

Measurement Systems for Industry and Energy

www.iba-ag.com

Manufacturer

iba AG

Koenigswarterstrasse 44

90762 Fuerth

Germany

Contacts

Main office	+49 911 97282-0
Support	+49 911 97282-14
Engineering	+49 911 97282-13
E-mail	iba@iba-ag.com
Web	www.iba-ag.com

Version	Date	Revision	Author	Version SW
2.6	02-2025	Changes in redundancy	rm, mm	8.10.0

Contents

1 About this documentation 6

1.1 Target group and previous knowledge 6

1.2 Notations 7

1.3 Used symbols 8

2 System requirements S7-Xplorer 9

3 PLC-Xplorer data interface to SIMATIC S7 systems 12

3.1 System topologies 13

3.2 Configuration and engineering SIMATIC S7 14

3.2.1 Configuration of LOGO! 14

3.2.2 Configuration of S7-200 16

3.2.3 Configuration of S7-300 and S7-400 18

3.2.4 Configuration of S7-1200 and S7-1500 18

3.3 Configuration & engineering ibaPDA 21

3.3.1 General interface settings 21

3.3.2 Adding a module 23

3.3.3 General module settings 24

3.3.4 Connection settings 26

3.3.5 Signal configuration 27

3.3.5.1 Selection via the absolute address of the operands 28

3.3.5.2 Selection via the symbolic operand addresses 29

3.3.5.3 Selection of CFC connectors 32

3.3.6 Module S7-Xplorer 34

3.3.6.1 Connection mode TCP/IP 35

3.3.6.2 Connection mode PC/CP 37

3.3.6.3 Connection mode TCP/IP S7-1x00 41

3.3.7 Module S7-Xplorer decoder 43

3.3.8 Module S7-Xplorer redundant 45

3.3.9 Module S7-Xplorer SINUMERIK 49

3.3.10 Module S5 (LAN adapter) 50

3.3.11 Module S7-200 52
3.3.11.1 Connection mode TCP/IP 53
3.3.11.2 Connection mode PC/CP 54
3.3.12 Module LOGO! 55
3.3.13 Output module 58
3.3.14 Module diagnostics 60
3.3.15 Address books 60
3.3.15.1 Creating address books offline from S7 project 61
3.3.15.2 Creating address books online from S7-1200/1500 CPU 63
3.3.16 Connection with SIMATIC WinAC (WinLC RTX) 63
3.3.17 Connection with S7-PLCSIM via ibaPDA-S7-Xplorer Proxy 64
3.3.17.1 Installation and application of ibaPDA-S7-Xplorer Proxy 65
3.3.17.2 Configuring ibaPDA-S7-Xplorer Proxy in ibaPDA 66
3.3.18 Connection with S7-PLCSIM Advanced 69
4 Diagnostics 70
4.1 License 70
4.2 Visibility of the interface 70
4.3 Log files 71
4.4 Connection diagnostics with PING 72
4.5 Connection table 72
4.6 Diagnostic modules 74
4.7 Connection diagnostics with PG/PC interface 79
4.8 Error messages 80
5 Appendix 86
5.1 Comparison of the cycle times on different access points 86
5.2 Comparison of the cycle times for different S7-CPUs 87
5.3 ibaPDA-S7-Xplorer Proxy for PC/CP interface 89
5.4 Setting PG/PC interface/defining new access point 90
5.5 S7 routing 93
5.5.1 Routing from Ethernet to Ethernet 93
5.5.1.1 Configuration of STEP 7/NetPro 94
5.5.1.2 Configuration of TIA Portal 96
5.5.1.3 Configuration of ibaPDA 97

5.5.2 Routing from Ethernet to PROFIBUS 99
5.5.2.1 Configuration of STEP 7/NetPro 99
5.5.2.2 Configuration of TIA Portal 101
5.5.2.3 Configuration of ibaPDA 102
5.6 Usage of MPI/DP-TCP-adapters 103
6 Support and contact 104

1 About this documentation

This documentation describes the function and application of the software interface *ibaPDA-Interface-S7-Xplorer*.

Other documentation

This documentation is a supplement to the *ibaPDA* manual. Information about all the other characteristics and functions of *ibaPDA* can be found in the *ibaPDA* manual or in the online help.

1.1 Target group and previous knowledge

This documentation is aimed at qualified professionals who are familiar with handling electrical and electronic modules as well as communication and measurement technology. A person is regarded as professional if he/she is capable of assessing safety and recognizing possible consequences and risks on the basis of his/her specialist training, knowledge and experience and knowledge of the standard regulations.

This documentation in particular addresses persons who are concerned with the configuration, test, commissioning or maintenance of Programmable Logic Controllers of the supported products. For the handling *ibaPDA-Interface-S7-Xplorer* the following basic knowledge is required and/or useful:

- Windows operating system
- Basic knowledge of *ibaPDA*
- Knowledge of configuration and operation of the relevant control system

1.2 Notations

In this manual, the following notations are used:

Action	Notation
Menu command	Menu *Logic diagram*
Calling the menu command	*Step 1 – Step 2 – Step 3 – Step x* Example: Select the menu *Logic diagram – Add – New function block*.
Keys	<Key name> Example: <Alt>; <F1>
Press the keys simultaneously	<Key name> + <Key name> Example: <Alt> + <Ctrl>
Buttons	<Key name> Example: <OK>; <Cancel>
Filenames, paths	`Filename`, `Path` Example: `Test.docx`

1.3 Used symbols

If safety instructions or other notes are used in this manual, they mean:

Danger!

The non-observance of this safety information may result in an imminent risk of death or severe injury:

- Observe the specified measures.

Warning!

The non-observance of this safety information may result in a potential risk of death or severe injury!

- Observe the specified measures.

Caution!

The non-observance of this safety information may result in a potential risk of injury or material damage!

- Observe the specified measures

Note

A note specifies special requirements or actions to be observed.

Tip

Tip or example as a helpful note or insider tip to make the work a little bit easier.

Other documentation

Reference to additional documentation or further reading.

2 System requirements S7-Xplorer

The following system requirements are necessary for the use of the S7-Xplorer data interface:

- *ibaPDA* v8.2.0 or higher
- Base license for *ibaPDA* + license for *ibaPDA-Interface-PLC-Xplorer* or *ibaPDA-Interface-S7-Xplorer*
- For more than 16 connections, you need additional *one-step-up-Interface-S7-Xplorer* licenses for each additional 16 connections.

Note

The *ibaPDA-Interface-PLC-Xplorer* license contains, among others, the license for this interface.

- SIMATIC STEP 7 or SIMATIC NET, in case PC/CP connections are used
- SIMATIC STEP 7 and S7-CFC, in case signals are to be selected using Drag & Drop

For further requirements for the used computer hardware and the supported operating systems, refer to the *ibaPDA* documentation.

Supported controllers:
SIMATIC S7 controller LOGO!, S7-200, S7-300, S7-400, S7-400H, S7-1200, S7-1500 or WinAC RTX or S5

System restrictions

- Restrictions SIMATIC TIA-Portal:
  - SIMATIC TIA-Portal STEP 7 V14 SP1 or higher is a prerequisite for generating address books from TIA Portal projects.
- Restrictions S7-1200 CPU:
  - For the TIA connection mode with S7-1200 CPUs, only firmware versions ≥ V4 are supported by the S7-Xplorer module.
- Restrictions LOGO! and S7-200:
  - Only LOGO!-controllers of the 0BA7 and 0BA8 families are supported.
  - Access to absolute operands only
  - The controllers SIMATIC LOGO! and SIMATIC S7-200 are to be configured via special software tools. Both project types cannot be read by *ibaPDA*. Hence, address books cannot be generated for a signal selection by means of the symbolic names or CFC connectors.
- Restrictions for S5:
  - Since ibaPDA cannot read SIMATIC STEP 5 projects, yet, address books for the signal selection cannot be created with the symbolic names.

Regarding the number of connections, the following technical constraints apply:

The number of S7-, SINAMICS- and SIMOTION-Xplorer connections is limited to 240. TCP and PC/CP connections of the SINAMICS-, SIMOTION- and SINUMERIK-Xplorer are evaluated differently. A SINAMICS-, SIMOTION- or SINUMERIK-Xplorer connection over TCP only counts 1/16. A PC/CP connection is calculated in full.

This means:

Number of S7-Xplorer connections

+ number of SINAMICS-Xplorer-TCP connections not routed / 16 [1)]

+ number of SIMOTION-Xplorer-TCP connections not routed / 16 [1)]

+ number of SINUMERIK-Xplorer-TCP connections not routed / 16 [1)]

+ number of SINAMICS-Xplorer-TCP connections routed

+ number of SIMOTION-Xplorer-TCP connections routed

+ number of SINAMICS-Xplorer-PC/CP connections

+ number of SIMOTION-Xplorer-PC/CP connections

+ number of SINUMERIK-Xplorer-PC/CP connections

≤ 240!

[1)] *Only active connections are taken into account. The division must always be rounded to the next larger number. (One connection resource per started 16 connections.)*

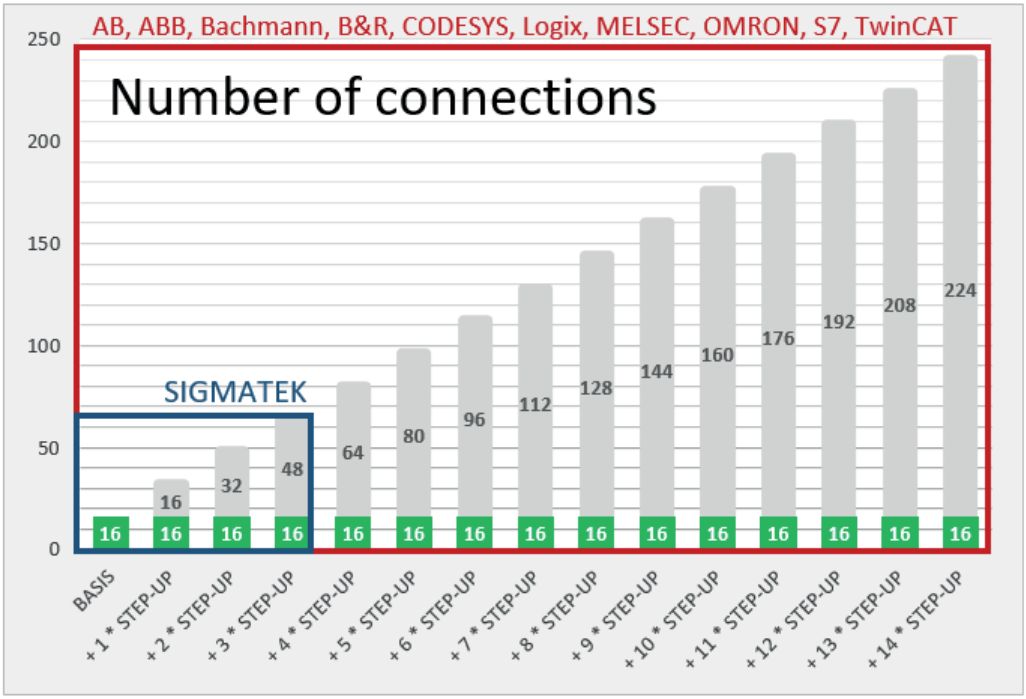
Access to the SINAMICS Control Units which are integrated in the SIMOTION controls is not supported.

License information

Order no.	Product name	Description
31.001042	ibaPDA-Interface-PLC-Xplorer	Extension license for an *ibaPDA* system adding all additional Xplorer data interfaces. (Full specifications under www.iba-ag.com)
31.000001	ibaPDA-Interface-S7-Xplorer	Extension license for an *ibaPDA* system adding the data interface S7-Xplorer (interface to SIMATIC S7) with 16 connections.
31.100001	one-step-up-Interface-S7-Xplorer	Extension license for 16 additional S7-Xplorer connections (max. 14 extension licenses)

Note

To use more than 16 data connections per interface, you can purchase the one-step-up-... extension licenses separately for each interface. Up to 16 further connections to PLCs can be established on each one-step-up-license. Up to 240 connections can be configured and used per data interface with the multiple purchase or multiple release of these licenses (up to 15 in total).
Exception of SIGMATEK: Here, only up to 4 licenses (64 connections) can be activated.



Consider the limitation of the number of signals by the *ibaPDA* base license.

3 PLC-Xplorer data interface to SIMATIC S7 systems

The S7-Xplorer interface is suitable for acquiring measured data via TCP/IP using the standard network interface cards as well as using the PPI, MPI, PROFIBUS, TCP/IP and ISO transport logs via SIMATIC NET interface cards. Access is transparent for the controller. Separate engineering and programming of the control is not necessary.

You can select the signals to be measured either with the absolute operand address or with the symbolic name with support of the *ibaPDA* address book browser. This browser allows to access to all defined symbols of the attached STEP 7 project.

When using the SIMATIC CFC editor (version V 6.0 or higher) on the same computer, you can configure the signals that are to be measured and the connectors from the control program via drag & drop.

Using an additional interface converter, you can also acquire measured data from a SIMATIC S5 controller via its AS511 interface (see ↗ *Module S5 (LAN adapter)*, page 50).

3.1 System topologies

You can establish the connections to the control systems via standard interfaces of the computer or corresponding CP modules.

The different connection types are categorized as follows:

- TCP/IP connection via the standard network interface
- SIMATIC specific connections via standard network adapters, special coupling modules (CP) or coupling adapters.

Controller	TCP/IP connection via the standard network interface	PPI/MPI/PROFIBUS connection via CP modules (PC/CP)	MPI/PROFIBUS connection via adapter (PC/CP)	TCP/IP and ISO connection (PC/CP)
S5	X[1)]	-	-	-
LOGO!	X[2)]	-	-	-
S7-200	X[3)]	X[5)]	X[6)]	-
S7-300	X[4)]	X[5)]	X[7)]	X[4)]
S7-400	X[4)]	X[5)]	X[7)]	X[4)]
S7-400H	X[4)]	X[5)]	X[7)]	X[4)]
S7-1200	X	-	-	-
S7-1500	X	-	-	-

Table 1: Available connection types

[1] On side of the SIMATIC S5, the AS511 interface is used with an additional interface converter.

[2] Only modules with Ethernet interface TCP/IP are supported (starting from …-0BA7)

[3] Only with CP243-1

[4] Only CPU/CP, whose Ethernet interface supports TCP/IP or ISO respectively

[5] Only with CP55xx (MPI), CP56xx (PPI/MPI/PROFIBUS) or CP57xx (PPI/MPI/PROFIBUS)

[6] PPI Multimaster cable is NOT supported.

[7] MPI/DP adapter serial, USB, TCP/IP

Additional Siemens software (e.g. SIMATIC NET or SIMATIC STEP 7) is needed for operation, in case the connection to the control system is established via a SIMATIC NET card in the computer to an integrated Ethernet interface of the CPU (if available) or to a CP module in the PLC. Please take into account that further licenses are required which are not part of *ibaPDA-Interface-PLC-Xplorer*.

Note

It is recommended carrying out the TCP/IP communication on a separate network segment to exclude a mutual influence by other network components.

3.2 Configuration and engineering SIMATIC S7

Generally, no particular engineering and programming is required on the controller side. In particular, it is not necessary to call up any program modules.

When using the PC/CP connection mode, you have to configure a suitable access point in the SIMATIC PG/PC interface on the *ibaPDA* computer, see ↗ *Setting PG/PC interface/defining new access point*, page 90.

Only for the LOGO! and S7-200 controller types, you have to configure an Ethernet connection in the control system, see ↗ *Configuration of LOGO!*, page 14 and ↗ *Configuration of S7-200*, page 16.

3.2.1 Configuration of LOGO!

Access to LOGO! CPU is carried out via Ethernet only.

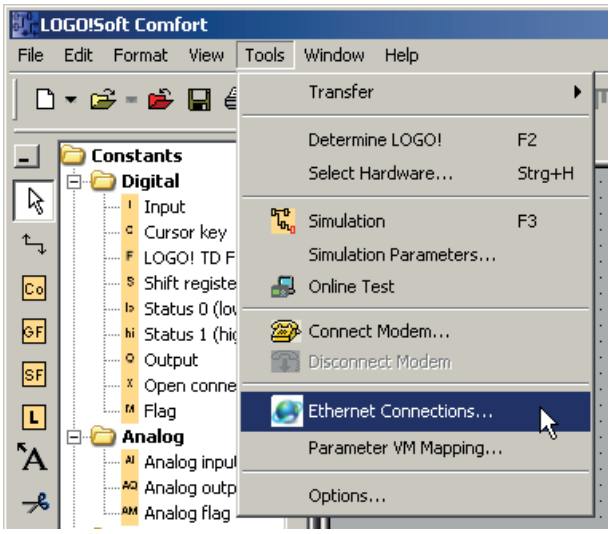
The following LOGO! types are supported:

- SIMATIC LOGO! 0BA7
- SIMATIC LOGO! 0BA8

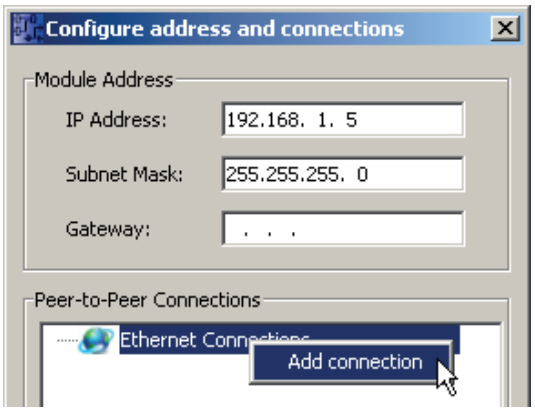
When using LOGO! 0BA7, you first have to establish an Ethernet connection with your LOGO! software or configure an existing connection respectively.

Creating an Ethernet connection

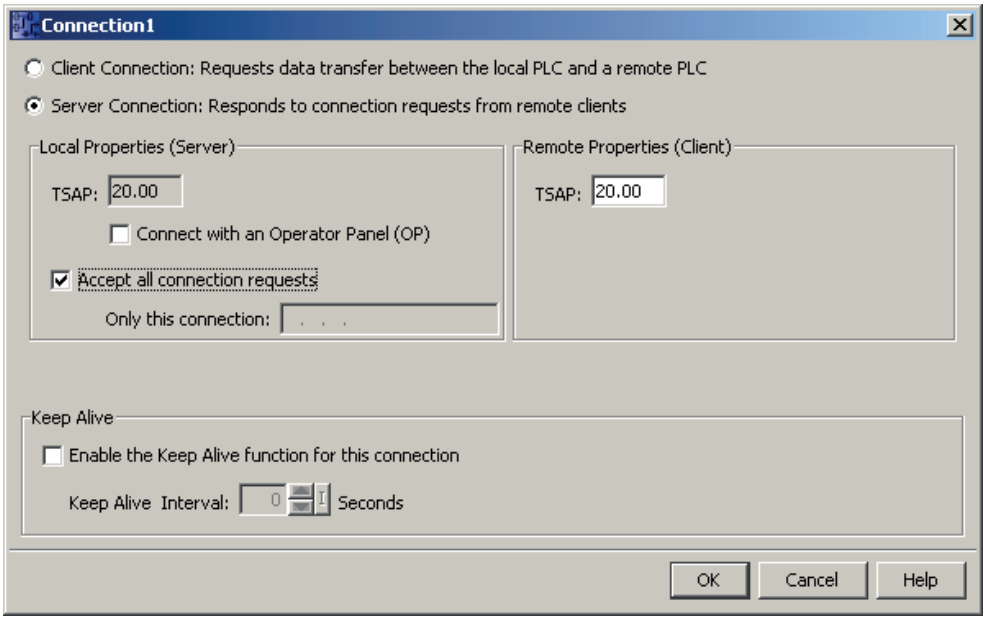
1. Select the menu *Tools – Ethernet connections…* in the LOGO! programming software.



2. Select *Add connection.*



3. Select *Server connection*, enter a TSAP (e.g. "20.00") and activate *Accept all connection requests*. The used TSAP has to be the same TSAP as in *ibaPDA*, see ↗ *Module LOGO!*, page 55.

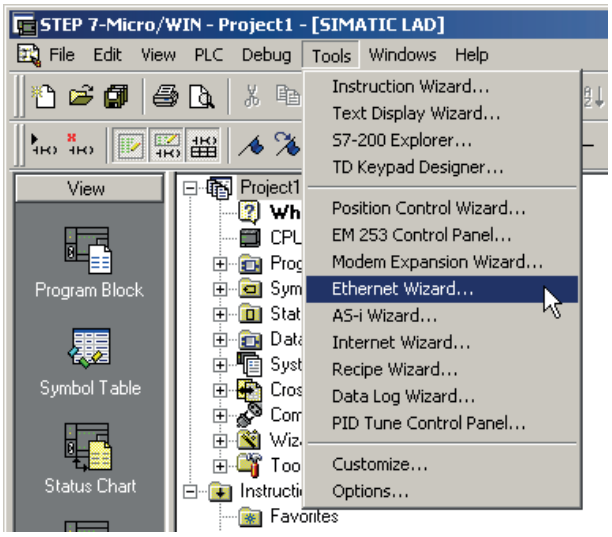


When using a LOGO! 0BA8, this is not required. However, the user has to consider here that every connection to the PLC has a 10 seconds timeout. Hence, an open connection is closed automatically if for a period of 10 seconds no user data are being exchanged.

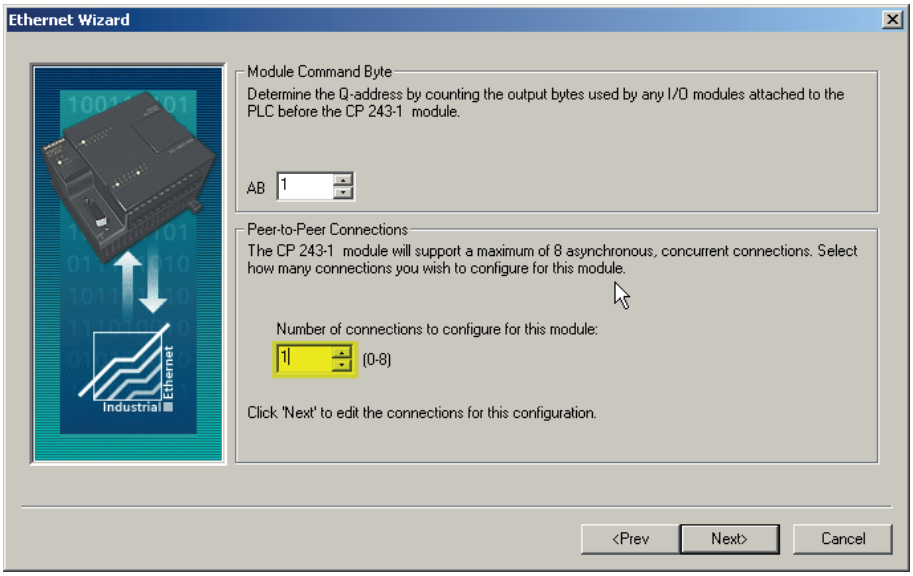
3.2.2 Configuration of S7-200

Access to the S7-200 is carried out via Ethernet. Therefore, you first have to establish a connection using the STEP 7-Micro/WIN software or parameterize an existing connection accordingly. The following figures provide more details on this.

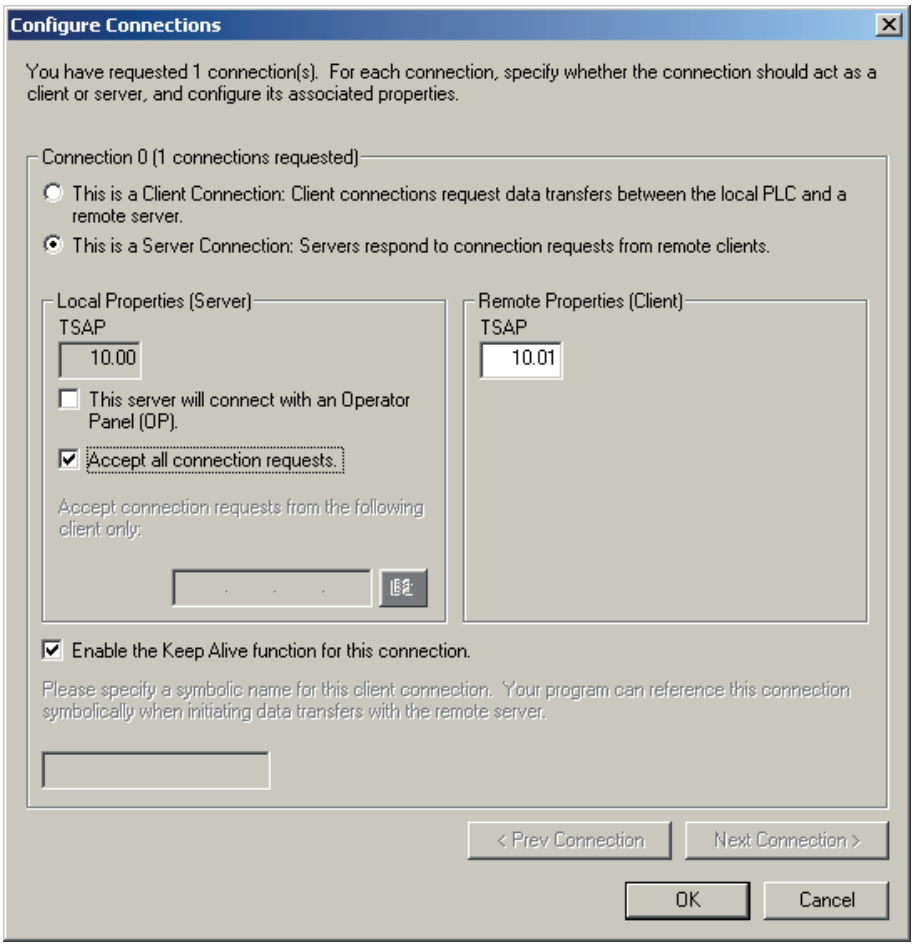
1. Select the menu *Tools – Ethernet Wizard...* in the STEP 7-Micro/WIN software.



2. Create an Ethernet connection.



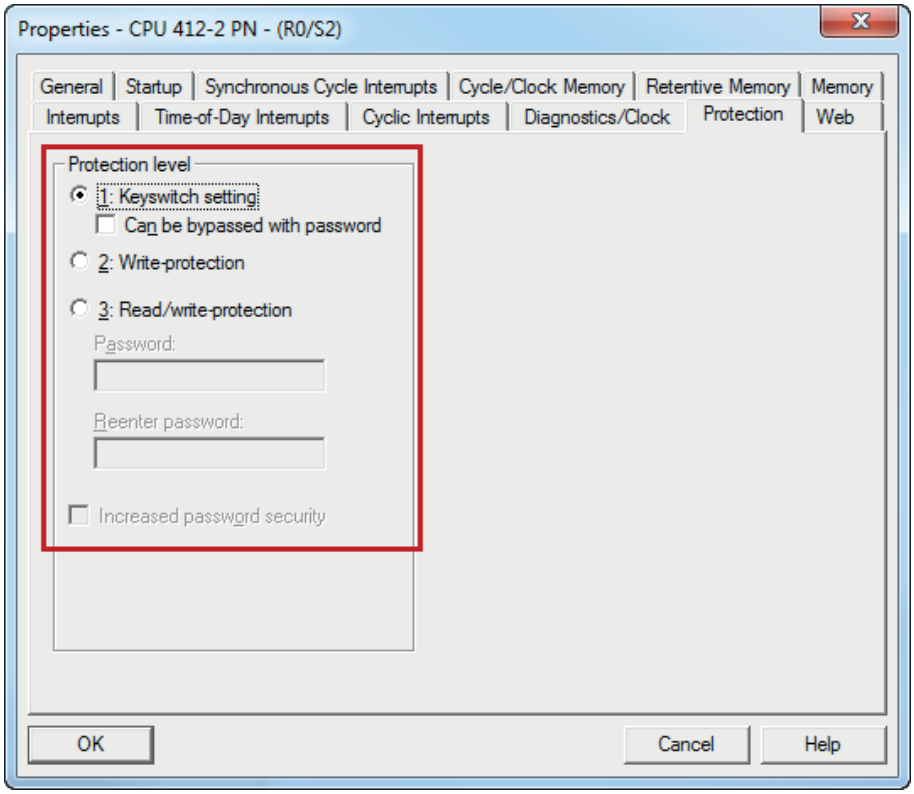
3. Select *Server connection*, enter a TSAP (e.g."10.00") and activate *Accept all connection requests*. The used TSAP has to be the same TSAP as in *ibaPDA*, see ↗ *Module S7-200*, page 52.



3.2.3 Configuration of S7-300 and S7-400

Access protection

You can configure an access protection for S7-300/400 CPUs.



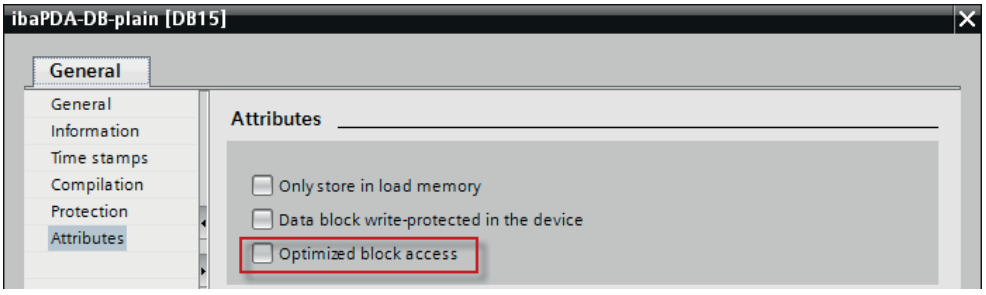
Depending on the settings a read and/or write access on the CPU by *ibaPDA* may be permitted or not.

3.2.4 Configuration of S7-1200 and S7-1500

Note the following when using the *S7-Xplorer* module on the CPUs S7-1200 and S7-1500:

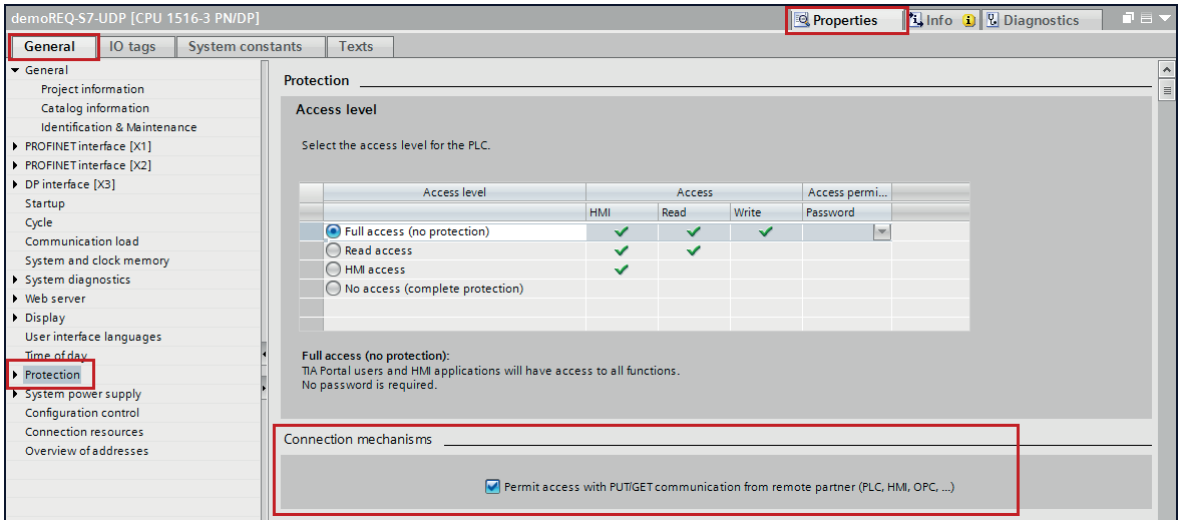
Connection mode TCP/IP or PC/CP

Access to data blocks only without *Optimized block access*.



PUT/GET communication for TCP/IP or PC/CP:
Up to TIA Portal V18, make the following settings in the TIA Portal:

In the CPU properties (*Properties – General – Protection – Connection mechanisms*), activate the access via PUT/GET communication.



For the S7-1200, this option is only available firmware V4.0 or higher.

As of TIA Portal V19 and CPU firmware V3.1 (V4.7 for S7-1200), activate the PUT/GET access for S7-1500 and S7-1200 CPUs as follows:

1. In the project navigation, navigate to *Security settings – Users and roles – Roles* tab.
2. Add a new role and enter a name, e.g. "Put/Get".
3. In the *Runtime rights* tab, select your PLC under *Function rights categories*.
4. Under *Function rights*, activate the access level *HMI access*.
5. Under *Users and Roles*, go to the *Users* tab.
6. Activate the user "Anonymous".
7. Confirm the following message with <OK>.
8. Under *Assigned roles*, assign the newly created role to the anonymous user.
9. In the *Device view*, open the properties of the CPU.
10. In the section navigation, navigate to *Protection & Security – Connection mechanisms*.
11. Activate the option *Permit access with PUT/GET communication from remote partner*.
12. Save and compile the configuration and load the changes into the CPU.

You can find further information in the SiePortal under
https://support.industry.siemens.com/cs/ww/en/view/109925755.

Connection mode TCP/IP S7-1x00

Nothing to observe

Access protection

You can configure an access protection for an S7-1200 and S7-1500 CPU. The following dependencies on *ibaPDA* apply:

Access level	**CPU access**	***ibaPDA* reads symbolic from CPU**	**S7 access**
Full access (no protection)	HMI, read, write	OK	OK
Read access	HMI, read	OK	OK
HMI access	HMI	No	OK
No access (complete protection)	-	No	No

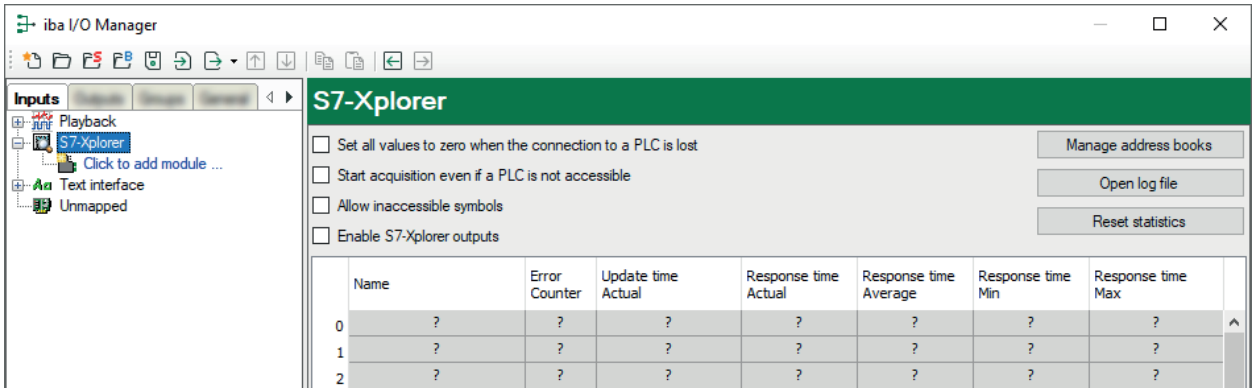
3.3 Configuration & engineering ibaPDA

The engineering for *ibaPDA* is described in the following. If all system requirements are fulfilled, *ibaPDA* displays the *S7-Xplorer* interface in the interface tree of the I/O Manager.

3.3.1 General interface settings

If the Xplorer interface is selected in the tree, you can see an overview of diagnostics information on the configured connections between *ibaPDA* and the controllers.

The interface has the following features and configuration options.



Set all values to zero when the connection to a PLC is lost
If this option is enabled, all measured values of the PLC are set to zero as soon as the connection is lost. If this option is disabled, *ibaPDA* keeps the last valid measured value in memory when the connection is lost.

Start acquisition even if a PLC is not accessible
If this option is enabled, the acquisition starts even if the controller is not accessible. A warning is prompted in the validation dialog, not an error. If the system was started without a connection to the controller, *ibaPDA* periodically tries to connect to the PLC.

Allow inaccessible symbols
If this option is enabled, the acquisition starts even if symbols are not accessible. These inaccessible symbols are prompted as warnings in the validation dialog box, not as errors. This can only occur if the address book is not up-to-date.

If this option is disabled, measurement does not start when inaccessible symbols are present.

Enable S7-Xplorer outputs
Enable this option to activate output modules. With the S7-Xplorer outputs you can write directly on S7 operands and S7 symbols.

Caution

Consider that writing data to the PLC might influence programs running on the PLC and can result in unwanted reactions and processes. Therefore ensure that activating the S7-Xplorer outputs does not involve any risks.

<Manage address books>
For more details about working with S7 address books, see ↗ *Address books*, page 60.

<Open log file>
If connections to controllers have been established, all connection specific actions are recorded in a text file. Using this button, you can open and check this file. In the file system on the hard disk, you find the log files of this interface in the path …\ProgramData\iba\ibaPDA\Log. The file name of the current log file is InterfaceLog.txt; the name of the archived log files is InterfaceLog_yyyy_mm_dd_hh_mm_ss.txt.

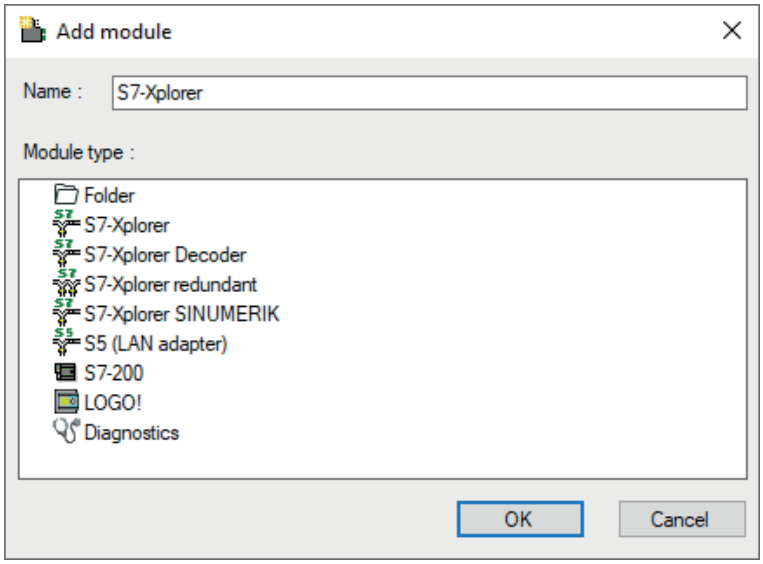
<Reset statistics>
Click this button to reset the calculated times and error counters in the table to 0.

Connection table
For each connection, the table shows the connection status, the current values for the update time (actual value, average, min. and max.) as well as the data size. In addition, there is an error counter for the individual connections during the acquisition.

See ↗ *Connection table*, page 72.

3.3.2 Adding a module

1. Click on the blue link *Click to add module* located under each data interface in the *Inputs* or *Outputs* tab.
2. Select the desired module type in the dialog box and assign a name via the input field if required.
3. Confirm the selection with <OK>.

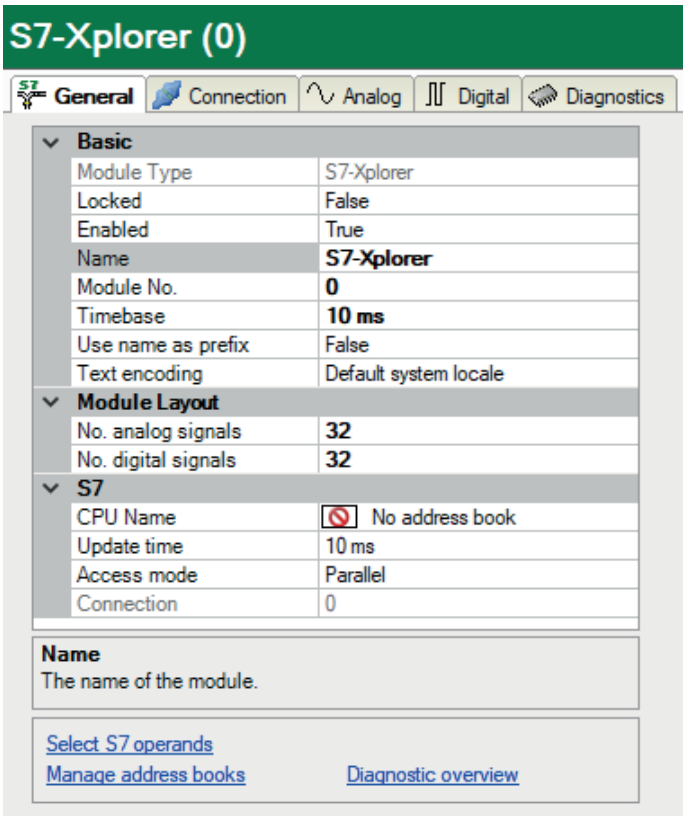


Module name	Description
S7-Xplorer	Connection to SIMATIC S7-300, S7-400, S7-1200 and S7-1500
S7-Xplorer Decoder	Acquisition of digital signals from BYTE, WORD and DWORD variables
S7-Xplorer redundant	redundant connection to SIMATIC S7-300, S7-400, S7-400H, S7-1200 and S7-1500 via the standard network interface
S7-Xplorer SIUMERIK	Preconfigured S7-Xplorer connection to integrated S7 controller of SINUMERIK
S5 (LAN adapter)	Connection to SIMATIC S5 via AS511 (adapter required)
S7-200	Connection to SIMATIC S7-200
LOGO!	Connection to SIMATIC LOGO!
Diagnostics	allows the recording of connection related diagnostic data

3.3.3 General module settings

To configure a module, select it in the tree structure.

All modules have the following setting options.



Basic settings

Module Type (information only)
Indicates the type of the current module.

Locked
You can lock a module to avoid unintentional or unauthorized changing of the module settings.

Enabled
Enable the module to record signals.

Name
You can enter a name for the module here.

Comment
You can enter a comment or description of the module here. This will be displayed as a tooltip in the signal tree.

Module No.
This internal reference number of the module determines the order of the modules in the signal tree of *ibaPDA* client and *ibaAnalyzer*.

Timebase
All signals of the module are sampled on this timebase.

Use module name as prefix
This option puts the module name in front of the signal names.

Text encoding
You can select the type of text encoding or the code page here for a correct interpretation and display of the received text data for inputs as well as of the text data to be sent for outputs. Available for selection are, beside system locale according to the Windows system settings (default) and UTF-8 Unicode, all other encodings.

Module Layout

No. analog signals/No. digital signals
Define the number of configurable analog and digital signals in the signal tables. The default value is 32 for each. The maximum value is 1000. The signal tables are adjusted accordingly.

Note

Observe the maximum number of signals permitted by your base license for *ibaPDA*.

Note

Take into consideration that the number of signals, which are read by a CPU, influences the minimum achievable update cycle. The more signals acquired, the longer the achievable update time.

S7

CPU name (not available with all modules)
Address book to which this module is connected. This is where the symbolic operands are selected.

Update time
Specifies the reference update time in which the data is requested from the PLC. During measurement, the real current update time may be higher than the specified value if the PLC needs more time to transmit the data. You can check in the connection table how fast the data is actually updated.

Access mode
The amount of data that an S7 can read in a single message is limited. Therefore, for larger amounts of data, several messages are used which can be controlled via the following send behavior:

- *Parallel* mode: Messages are sent directly in succession without waiting for a response from the S7 (fast mode, but not supported by some PLCs or adapters).
- *Sequential* mode: Messages are only sent consecutively after the S7 has responded to the first message.

Connection (read only)
Connection ID that is used to access S7.

Other documentation

You can find further information in the *ibaPDA* manual.

Link "Select symbols"
Click on this link after the connection has been successfully established to configure the signals to be measured.

3.3.4 Connection settings

Configure the connection of the module to the controller in the *Connection* tab.

The connection settings differ according to the controller type. The following chapters contain detailed descriptions:

↗ *Module S7-Xplorer*, page 34

↗ *Module S7-Xplorer redundant*, page 45

↗ *Module LOGO!*, page 55

↗ *Module S7-200*, page 52

↗ *Module S5 (LAN adapter)*, page 50

3.3.5 Signal configuration

The signals to be acquired are selected in the I/O Manager. There are 3 different ways for selecting the signals to be acquired:

- Selection via the absolute address of the S7 operands
- Selection via the S7 symbol addresses (symbol table and symbols from DBs) in a symbol browser
- Selection via the CFC connectors (when programming the CPU with S7-CFC V6.0 or higher)

ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
S5	X	-	-	-	-	-	-	-	-
LOGO!	X	-	-	-	-	-	-	-	-
S7-200	X	-	-	X	-	-	-	-	-
S7-300	X	X	X	X	X	X	-	-	-
S7-400	X	X	X	X	X	X	-	-	-
S7-400H	X	X	X	X	X	X	-	-	-
S7-1200	X[1)]	-	-	-	-	-	-	X	-
S7-1500	X[1)]	-	-	-	-	-	-	X	-

Explanation: **ABS**olute address, **SYM**bolic, **CFC**connector

Note

When using the TIMER operand type with S7-200 controllers, you have to consider that this type has different fixed time bases depending on the operand address. Unlike the S7-300/S7-400 controllers, this type cannot be selected in the configuration and cannot be read by *ibaPDA*.

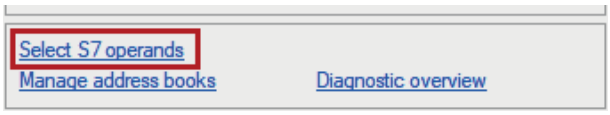
This is why you have to multiply the read value with the time base determined from the S7-200 manual for getting an absolute time value in milliseconds in *ibaPDA*.

[1)] no access on optimized blocks

3.3.5.1 Selection via the absolute address of the operands

You have two options to select the measurement values via the operand address:

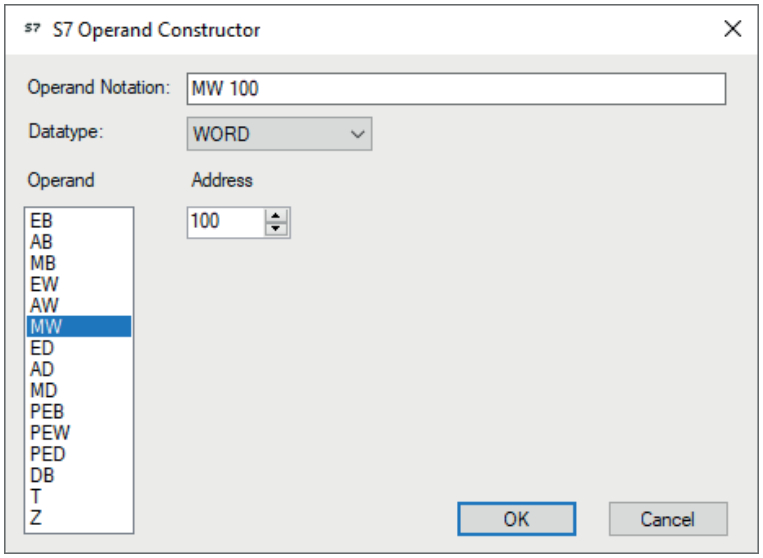
- In the module's *General* tab, click on the *Select S7 operands* link.



  The S7 operand editor opens.

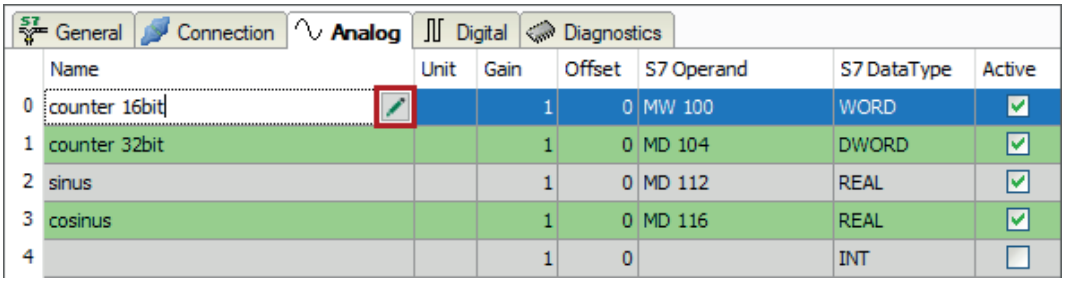
- In the *Analog* or *Digital* tab, click in a cell in the *S7 operand* column.

  The button <...> appears. Click the button to open the S7 operand editor.



After you have set the desired operand address, click <OK> to exit the dialog.

Thereafter, you can enter the signal name in the *Name* column.



You can enter the desired operand address also directly in the *S7 Operand* column without using the S7 operand editor.

3.3.5.2 Selection via the symbolic operand addresses

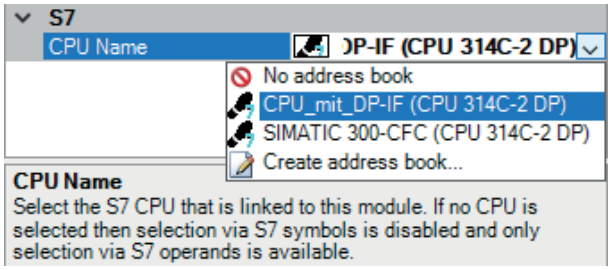
An advantage of this way of access is that the symbol addresses are applied automatically in *ibaPDA* as signal names.

Requirements for this method of access:

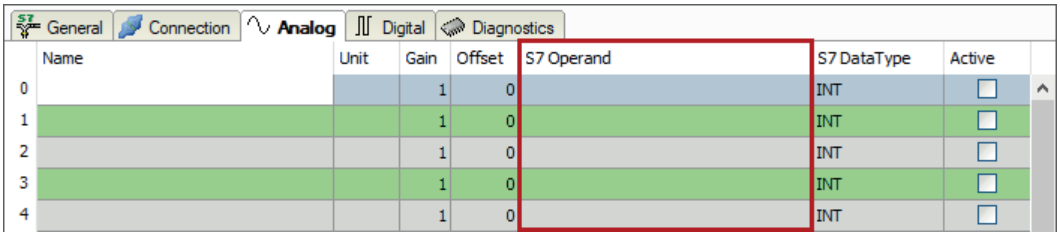
- The signals to be measured already have an entry in the S7 symbol table, the PLC variable list or in a data block.
- An address book has been created (see chapter ↗ *Address books*, page 60).

Integrating an address book into a module

- In the module's *General* tab under *CPU Name*, select the S7-CPU you want to assign this module to.



→ In the *Analog* and *Digital* tabs, an additional column *S7 Symbol* is displayed.

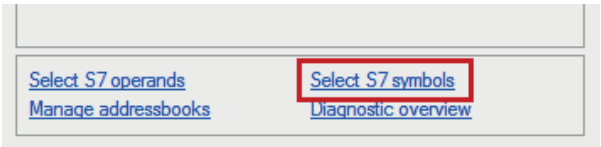


→ Now you can access symbol addresses using the S7 CFC- and Symbol Browser (in short: symbol browser).

Selecting signals via the symbol browser

You have two options to select the signals to be measured:

- In the module's *General* tab, click on the *Select S7 symbols* link.



  The symbol browser opens.

  In the symbol browser, you can select all symbols of the address book. *ibaPDA* enters the selected signals automatically in the appropriate table *Analog* or *Digital*. You can add several signals successively.

- On the *Analog* or *Digital* tab, click in a cell of the *S7 Symbol* column.

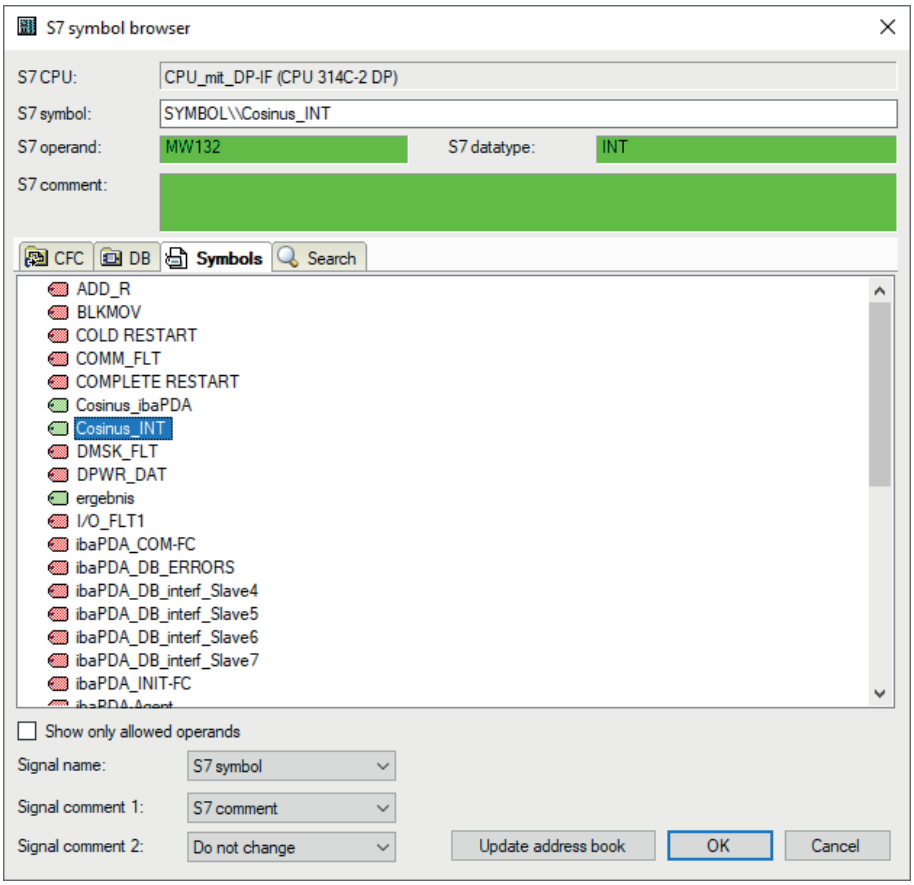
  The button <...> appears. Click this button to open the symbol browser.

  In the symbol browser, you can only select symbols with the data type matching the table. *ibaPDA* enters the symbol in the previously selected row of the signal table. After every selection, the symbol browser closes.

User interface of the symbol browser

In the symbol browser, you have the following options:

- CFC variables:
  On the *CFC* tab, you can select the configured CFC variables consisting of the configured names of chart, block and connector.
- DB variables:
  On the *DB* tab, you can select the individual data blocks and their variables.
- Symbol table:
  On the *Symbols* tab, you can select the entries of the S7 symbol table.
- *Search* tab:
  You can search the variables with a part of their name.



After selecting the variable, the symbol browser shows the operand address, the data type and a comment.

The variables have the following colors:

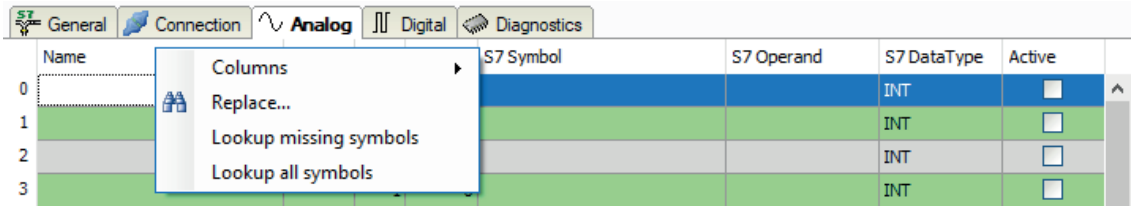
Green	The operand is valid. You can transfer it to the signal table with <Add> or <OK>.
Yellow	The operand has a data type that does not match the selected row or table, e. g. in case you have selected a Boolean variable as analog value or an integer value as digital value.
Red	The operand has a data type that is not supported by *ibaPDA*, or the operand is a constant.

Show only allowed operands
If you activate this option, the symbol browser shows only operands and symbols that *ibaPDA* supports or that match the signal table from which you opened the browser (i.e. no red and yellows ones).

Signal name, signal comment 1 and 2
Usually, *ibaPDA* adopts the symbolic signals name from STEP 7 as signal name in the I/O Manager. Using these three drop-down menus, you can change the signal name and both comments.

Please select an option from the alternatives offered. If a signal name or comment in the signal table should not be changed, select *Do not change*.

Looking up symbols in the signal table



ibaPDA can look up the symbol that corresponds with an operand.

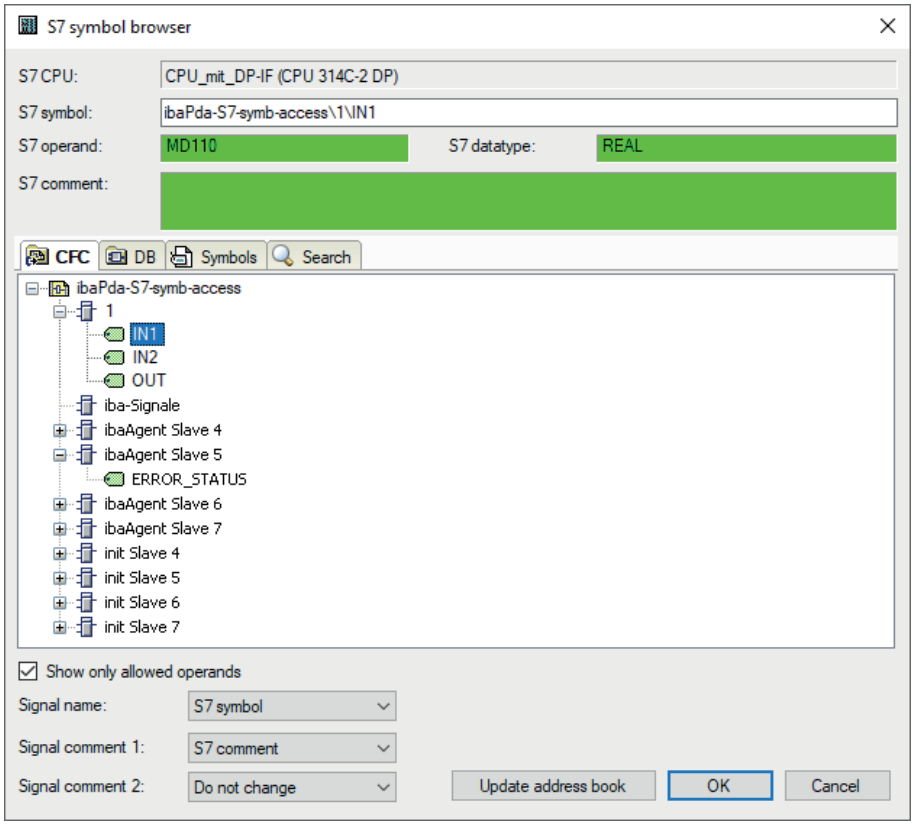
Right-click on the table header of the signal table.

- Select *Lookup missing symbols* to search only for the missing symbols.
- Select *Lookup all symbols* to search and replace all symbols. The command executes a backward resolution of the S7 symbols out of the S7 operands. *ibaPDA* searches the symbol table first, then CFC and finally the DBs for the operand.

3.3.5.3 Selection of CFC connectors

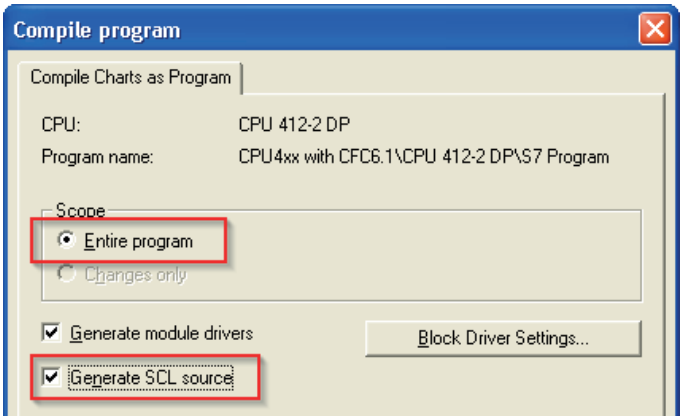
In order to select CFC connectors for the measurement, open the symbol browser first, see ↗ *Selection via the symbolic operand addresses*, page 29.

In the symbol browser, you open the *CFC* tab and select the signal. The connectors are listed hierarchically by chart name, module name and connector name:



Note

If no connectors are shown in the CFC tab, the SCL sources might not have been translated in the STEP 7 project. Activate the following options in the dialog for compiling the program in the SIMATIC software:



Afterwards, generate the address books again.

Note

When compiling a CFC program, DB addresses are assigned automatically in STEP 7 to the connectors. It may happen that other DB addresses are assigned to connectors, depending on the scope of the program changes that have been made between two compilation runs.

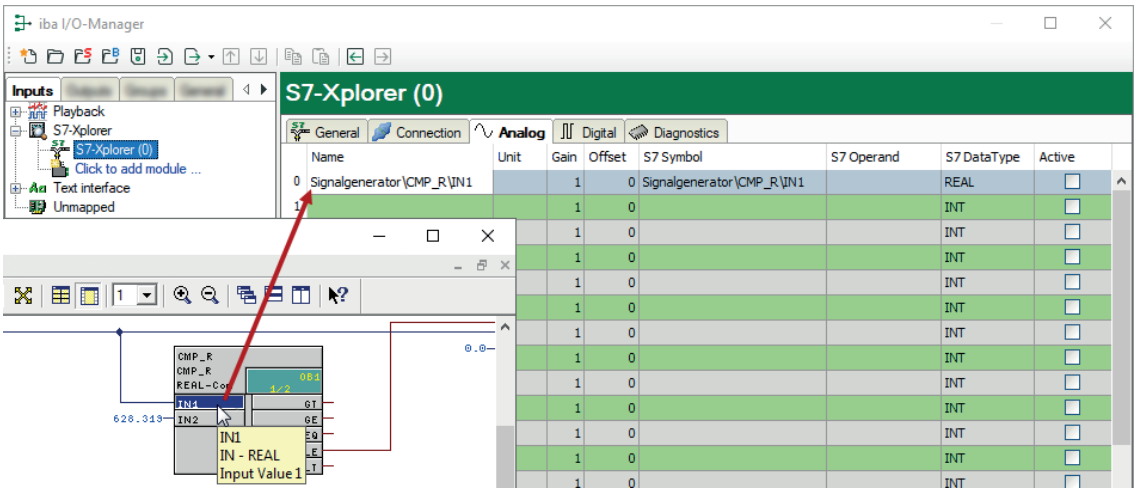
In this case, also the address book for *ibaPDA* has to be generated again. The symbolically configured signals are automatically checked in the I/O Manager whereas the related absolute S7 operands are updated.

Special function Drag & Drop

A convenient way to select signals is to drag & drop them from the CFC chart into the I/O Manager of *ibaPDA*.

1. Open the I/O Manager of *ibaPDA* and the signal table of the required module.
2. Start the CFC editor on the same computer as the *ibaPDA* client.
3. Drag the connector from the CFC editor into the desired line of the signal table in the I/O Manager of *ibaPDA*.

→ The CFC connector is now a measured signal in the signal table of the module.



3.3.6 Module S7-Xplorer

This module establishes the connection to various SIMATIC S7 controllers via the standard network interface of the computer or an additional network interface card.

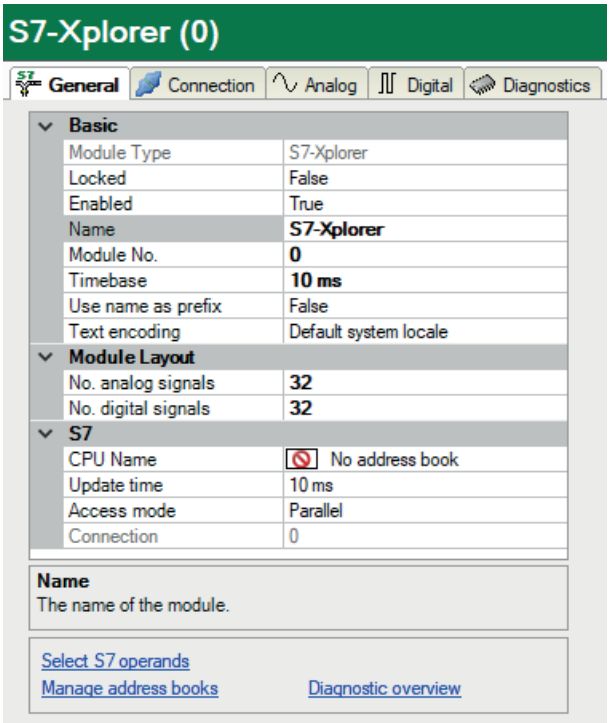
ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
S7-300	X	X	X	X	X	X	-	-	-
S7-400	X	X	X	X	X	X	-	-	-
S7-1200	X	-	-	-	-	-	-	X	-
S7-1500	X	-	-	-	-	-	-	X	-

Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector

ibaPDA supports the following operand types:

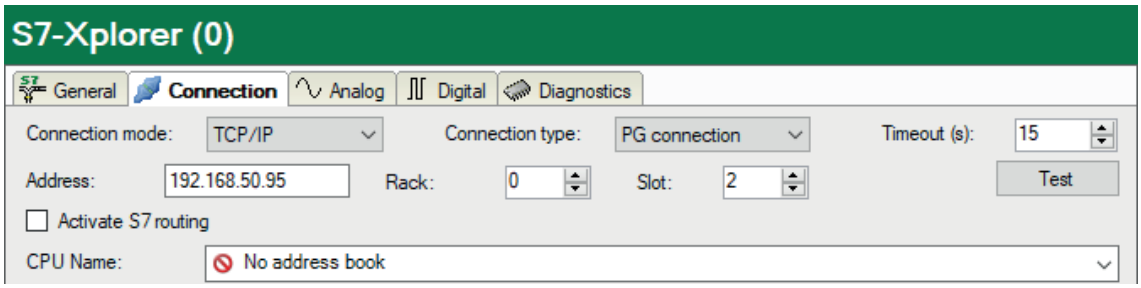
Selection via	Analog	Digital
S7-300 S7-400	EB, AB, MB, PEB, DBB EW, AW, MW, PEW, DBW ED, AD, MD, PED, DBD T, Z	E, A, M, DBX
S7-1200 S7-1500	EB, AB, MB, DBB EW, AW, MW, DBW ED, AD, MD, DBD	E, A, M, DBX



For more information on the module settings see ↗ *General module settings*, page 24.

3.3.6.1 Connection mode TCP/IP

This mode activates a connection via the standard network interface of the computer.



Connection mode
Selection of the TCP/IP connection mode

Connection type
Selection of the connection type PG, OP, or other connections (determines which type of connection resource is occupied on the CPU).

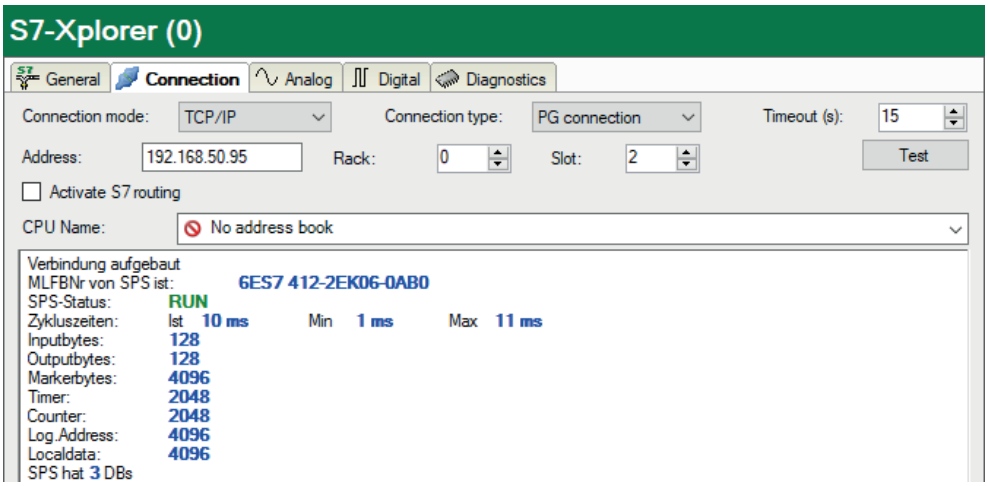
Timeout
Specify a value for the timeout in seconds for establishing the connection and for read access. If the time set here is exceeded, *ibaPDA* declares the controller as not accessible or not responsive.

Address
IP address of the controller

Rack
Rack number of the controller (default: 0)

Slot
Slot number of the controller in the rack
(Use "0" for S7-1500 CPUs.)

<Test>
ibaPDA tests the connection to the CPU and displays available diagnostic data.



Activate S7 routing
Activate this option if the S7-CPU and the *ibaPDA* computer are not in the same network, but only communicate over a gateway that supports S7 routing. Such a gateway can be e.g. an IE/PB link, over which a S7-CPU can be reached without an Ethernet connection.

Two additional input fields appear:

- Address of device acting as gateway: Enter the IP address of the gateway.
- S7 subnet ID of target net: Enter subnet ID from STEP 7 NetPro or TIA Portal.

For more information on S7 routing, see ↗ *S7 routing*, page 93.

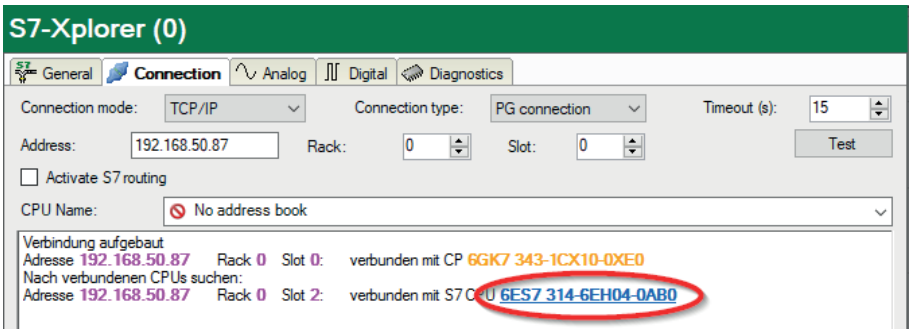
CPU Name
Selection of the linked address book

Tip

If you do not know the rack number and/or slot number, enter "0" and click the <Test> button.

ibaPDA then tries to establish a connection. If the connection is successfully established, *ibaPDA* scans the relevant rack for available CPUs. *ibaPDA* lists the CPUs found with rack number and slot numbers. The CP to which *ibaPDA* is directly connected is always at the top.

When you click on an S7-CPU (hyperlink), *ibaPDA* establishes the connection and displays the CPU diagnostic data.



Note

For the communication with the CPU, port 102 must be allowed in the target system. If the data traffic runs via an external firewall, then you must also allow port 102 through this firewall.

3.3.6.2 Connection mode PC/CP

This mode activates a connection over the interface cards of the computer that are configured using SIMATIC Net.

You can use the interfaces configured in SIMATIC Net, e.g.:

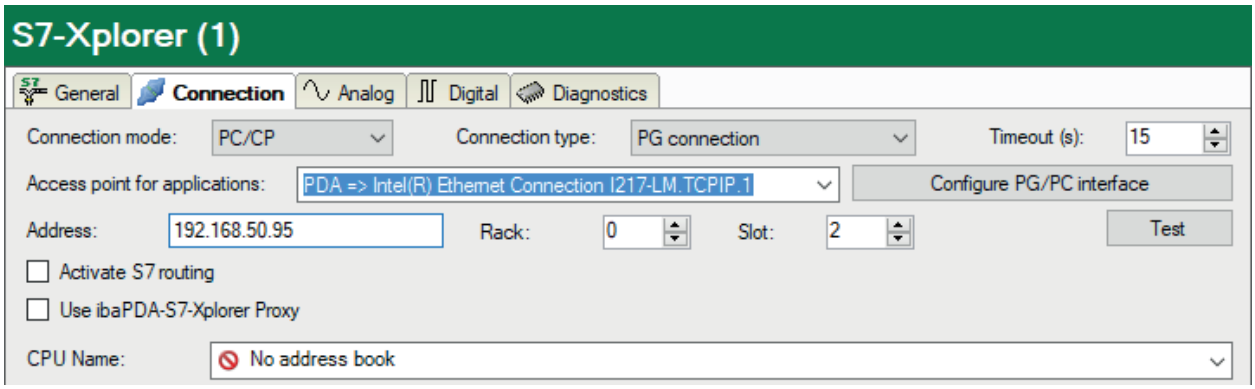
- MPI adapter (COM)
- MPI adapter (USB)
- PROFIBUS (CP5611, CP5622)
- TCPIP (RFC1005)
- ...

Note

If you want to use this connection type, the Siemens software SIMATIC Net (e.g. SIMATIC Manager or Softnet) has to be installed. When using the modules CP55..., CP56... and the MPI adapter, the installation of the device drivers is sufficient.

Note

ibaPDA does not support the SIMATIC PC/PPI cable as well as the SIMATIC USB/PPI cable for connecting S7-200 CPUs. For this purpose, either use a CP56xx (PPI) on the computer side, or a CP243-1 with TCP/IP on the PLC side.



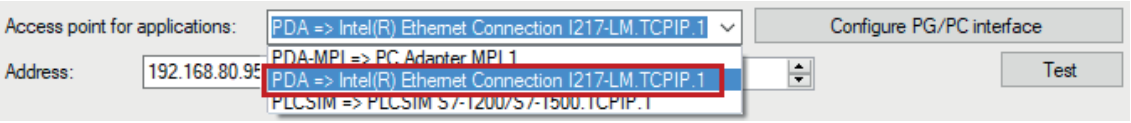
Connection mode
Selection of the PC/CP connection mode

Connection type
Selection of the connection type PG, OP, or other connections (determines which type of connection resource is occupied on the CPU).

Timeout
Specify a value for the timeout in seconds for establishing the connection and for read access. If the time set here is exceeded, *ibaPDA* declares the controller as not accessible or not responsive.

Access point for applications
Selection of the access point to be used

For more information on creating and configuring an access point, see ↗ *Setting PG/PC interface/defining new access point*, page 90.



Note

Configure available access points in SIMATIC Net with the "PG/PC interface settings" tool by Siemens.

For the connection of *ibaPDA-Interface-S7-Xplorer* to SIMATIC S7 via PC/CP connections, iba generally recommends setting a specific access point for *ibaPDA* when *ibaPDA-Interface-S7-Xplorer* and SIMATIC Manager run on the same computer. With an own access point, there is no longer the risk that the access for *ibaPDA-Interface-S7-Xplorer* will be disturbed in case the standard access point is changed in the SIMATIC Manager.

<Configure PG/PC interface>
This button opens the dialog box for setting the PG/PC interface of SIMATIC STEP 7.

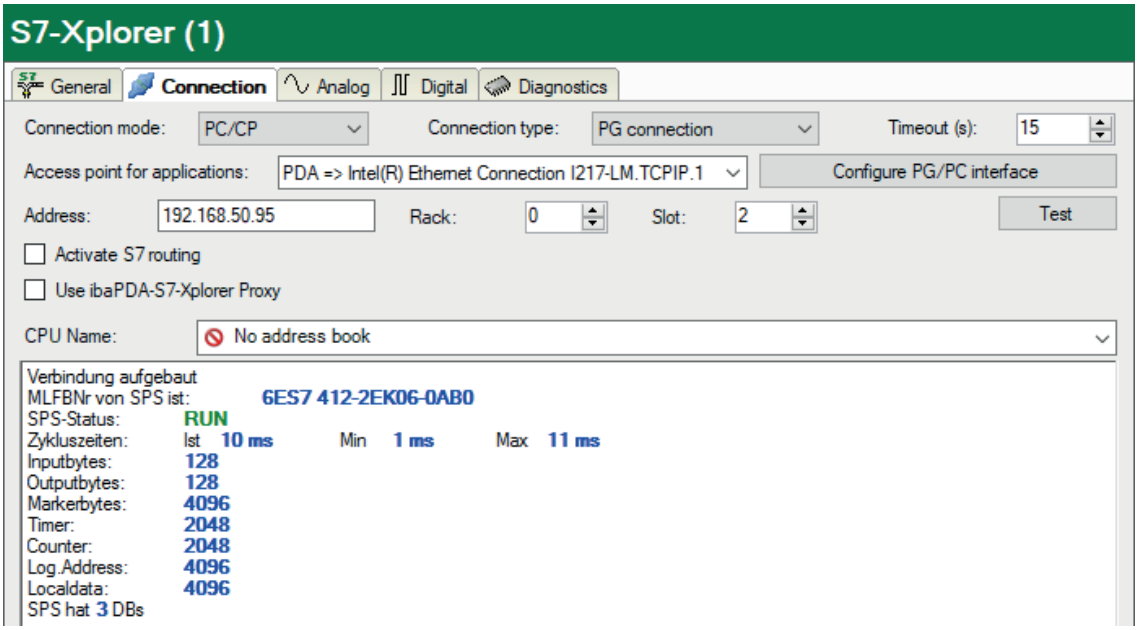
Address
Address of the controller (MPI, PROFIBUS or IP address depending on the configured access point)

Rack
Rack number of the controller (default: 0)

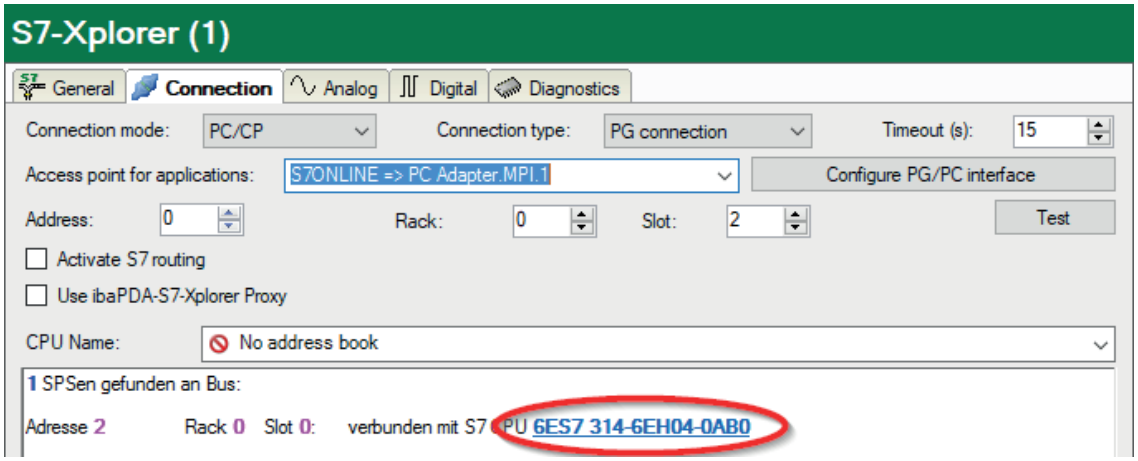
Slot
Slot number of the controller in the rack
(Use "0" for S7-1500 CPUs.)

<Test>

ibaPDA tests the connection to the CPU and displays available diagnostic data.



When you click on an S7-CPU (hyperlink), *ibaPDA* establishes the connection and displays the CPU diagnostic data.



Activate S7 routing

Activate this option if the S7-CPU and the *ibaPDA* computer are not in the same network, but only communicate over a gateway that supports S7 routing. Such a gateway can be e.g. an IE/PB link, over which a S7-CPU can be reached without an Ethernet connection.

Two additional input fields appear:

- Address of device acting as gateway: Enter the IP address of the gateway.
- S7 subnet ID of target net: Enter subnet ID from STEP 7 NetPro or TIA Portal.

For more information on S7 routing, see ↗ *S7 routing*, page 93.

Use ibaPDA-S7-Xplorer Proxy
For a connection to S7-PLCSIM it is necessary to use the *ibaPDA-S7-Xplorer Proxy*, because due to operating system reasons it is not possible to establish a direct connection between the *ibaPDA* service and S7-PLCSIM.

Two additional input fields appear:

- IP address or hostname of the *ibaPDA-S7-Proxy* computer
- Number of the port used by the *ibaPDA-S7-Proxy*

For more information on *ibaPDA-S7-Xplorer Proxy* see also ↗ *Connection with S7-PLCSIM via ibaPDA-S7-Xplorer Proxy*, page 64 and ↗ *ibaPDA-S7-Xplorer Proxy for PC/CP interface*, page 89.

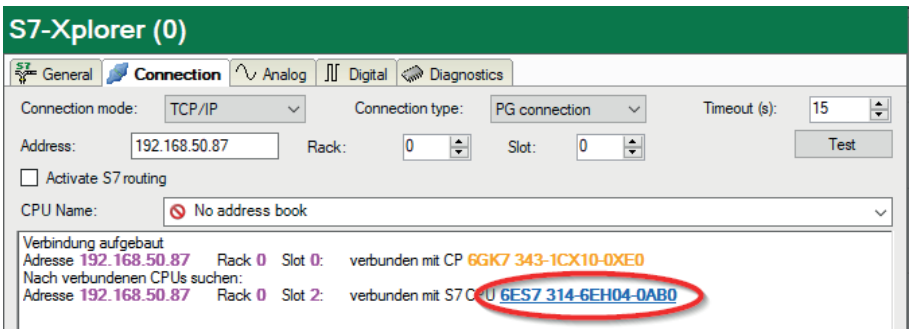
CPU Name
Selection of the linked address book

Tip for S7-300 and S7-400 CPUs

If you do not know the rack number and/or slot number, enter "0" and click the <Test> button.

ibaPDA then tries to establish a connection. If the connection is successfully established, *ibaPDA* scans the relevant rack for available CPUs. *ibaPDA* lists the CPUs found with rack number and slot numbers. The CP to which *ibaPDA* is directly connected is always at the top.

When you click *S7-CPU* (hyperlink), *ibaPDA* establishes the connection and displays the CPU diagnostic data.



Note

If no PLC or CPU is displayed for the MPI access point, check the interface setting in the SIMATIC Manager.

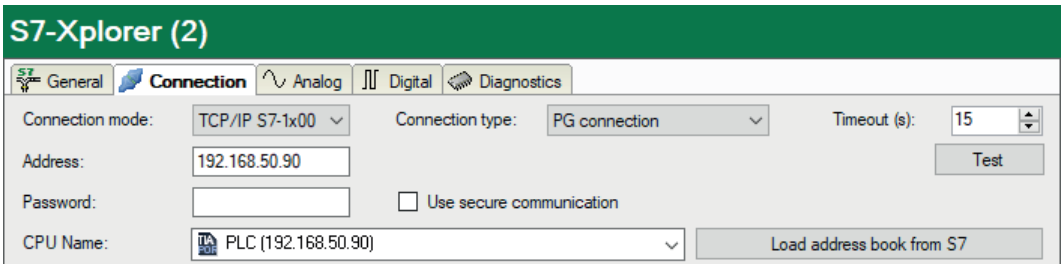
ibaPDA does not detect an online connection if a PG/PC interface with interface parameterization "AUTO" has been set for MPI adapter or CPs in the SIMATIC Manager.

In this case, you need to change the interface or the access point.

3.3.6.3 Connection mode TCP/IP S7-1x00

This mode activates a connection via the standard network interface of the computer. You can only use this mode in combination with S7-1200 CPUs (V4 or higher) and S7-1500 CPUs.

ibaPDA supports access to both optimized and non-optimized blocks.



Connection mode
Selection of the TCP/IP S7-1x00 connection mode.

Connection type
Selection of the connection type PG, OP, or other connections (determines which type of connection resource is occupied on the CPU).

Timeout
Specify a value for the timeout in seconds for establishing the connection and for read access. If the time set here is exceeded, *ibaPDA* declares the controller as not accessible or not responsive.

Address
IP address of the controller

Password
Depending on the configuration in the controller, access to the PLC may be protected by a password. In this case, enter this password here.

Use secure communication
The S7-1500 controller supports secure communication via TLS encryption with TIA Portal v17 or higher. In TIA Portal, you can set secure PG/PC and HMI communication.
If you have activated this option in the controller, you must also activate secure communication in *ibaPDA*.

CPU Name
Selection of the linked address book (only TIA Portal address books available).

Note

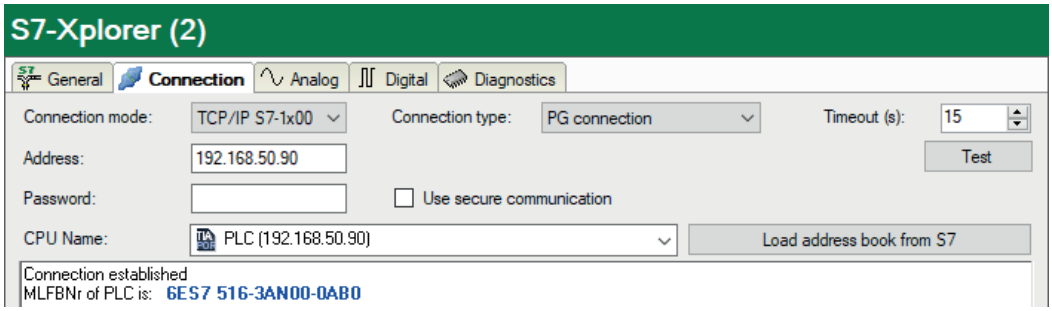
The TCP/IP S7-1x00 connection mode does not support the use of absolute operand addresses.

Note

For the communication with the CPU, port 102 must be allowed in the target system. If the data traffic runs via an external firewall, then you must also allow port 102 through this firewall.

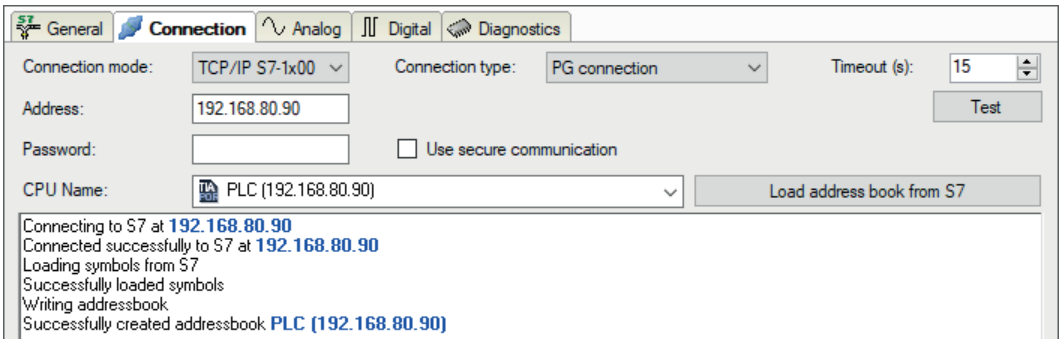
<Test>

ibaPDA tests the connection to the CPU and displays available diagnostic data.



<Load address book from S7>

By clicking on this button *ibaPDA* reads the list of symbols directly from the PLC and stores it in an address book for further use in the symbol browser.



In case of an S7-1500, please consider the option to enable access protection.

For further information, see ↗ *Configuration of S7-1200 and S7-1500*, page 18.

3.3.7 Module S7-Xplorer decoder

The module is suitable for acquiring large quantities of digital signals from a SIMATIC S7 controller.

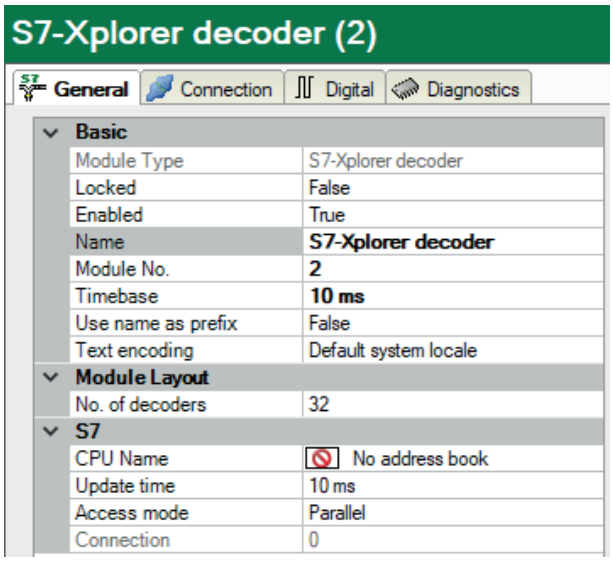
ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
S7-300	X	X	X	X	X	X	-	-	-
S7-400	X	X	X	X	X	X	-	-	-
S7-1200	X	-	-	-	-	-	-	X	-
S7-1500	X	-	-	-	-	-	-	X	-

Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector

ibaPDA supports the following operand types:

Selection via	Analog
S7-300 S7-400	EB, AB, MB, PEB, DBB EW, AW, MW, PEW, DBW ED, AD, MD, PED, DBD T, Z
S7-1200 S7-1500	EB, AB, MB, DBB EW, AW, MW, DBW ED, AD, MD, DBD



For more information on the module settings see ↗ *General module settings*, page 24.

Module layout

No. of decoders

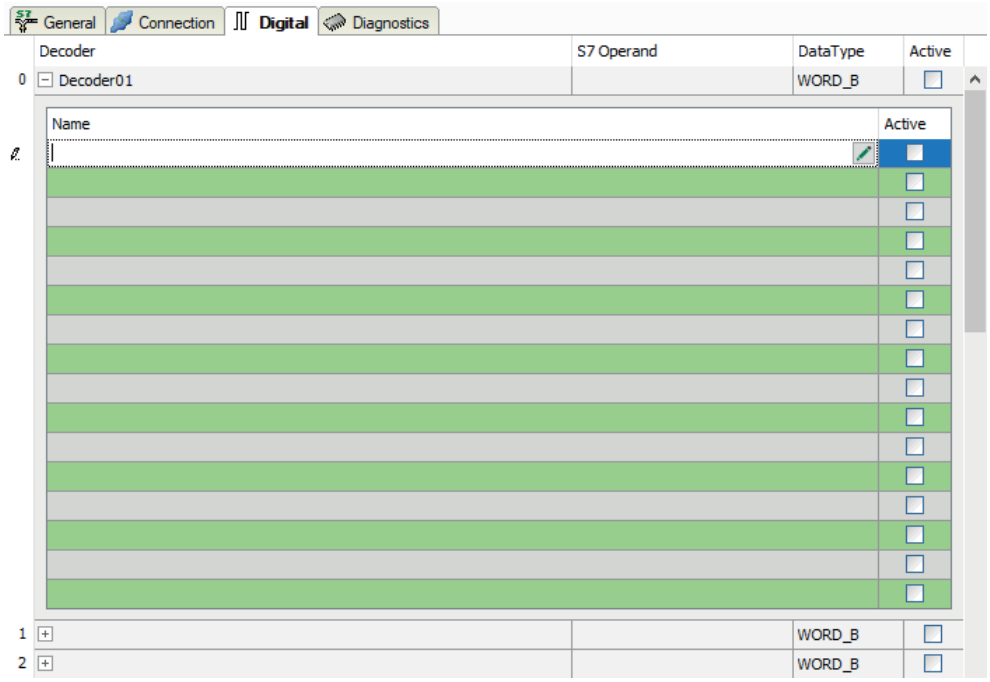
Define the number of configurable decoders in the digital signal table. The default value is 32. The maximum value is 128. The signal tables are adjusted accordingly.

Connection configuration

Set the connection of the *S7-Xplorer Decoder* module in the same way as the connection for an S7-Xplorer module:

↗ *Connection mode TCP/IP*, page 35
↗ *Connection mode PC/CP*, page 37
↗ *Connection mode TCP/IP S7-1x00*, page 41

Digital tab



The declaration of the digital signals is done on two levels. First define the source signals, which are broken down for the digital signals (bits).

You can open each source signal via the <+> button to display the list of associated digital signals. Then define the individual digital signals of the source signal.

The individual columns of the signal table have the following meanings:

Source signal

Decoder
Enter a name for the source signal.

S7 Operand/S7 Symbol
Enter the S7 operand to which the signal is assigned and, if applicable, the S7 symbol.

DataType
Enter the data type of the signal. The data type also determines the number of digital signals. *ibaPDA* automatically derives the possible data type from the S7 operand or S7 symbol.

Active
If you activate the source signal, it is acquired with all digital signals. You can deactivate individual digital signals.

Individual digital signals (bits)

Name
Enter a name for the individual digital signals.

Active
If you activate the signal, the signal is acquired and is also considered when checking the number of licensed signals.

Note

ibaPDA only takes the activated digital signals into account for the number of licensed signals, i.e. no additional signal for the source signal.

3.3.8 Module S7-Xplorer redundant

This module establishes a redundant connection via the standard network interface of the computer or an additional network card.

Note

The module S7-Xplorer redundant seizes two licensed connections. Please check whether the total number of licensed connections is also sufficient when using redundant modules.

ibaPDA always reads data only from one CPU, which is called active CPU. It is decided on the basis of the CPU status and the connection status from which CPU the data are taken.

With only one CPU connected, data is taken from this CPU. If both CPUs are connected and only one of them in the RUN status, *ibaPDA* chooses the latter CPU is chosen. If both are connected and in RUN status, then *ibaPDA* chooses the first CPU (connection 0). This does not apply to a S7-400H system. In that case *ibaPDA* chooses the CPU with the RUN MASTER status.

If there is a change of the CPU status or the connection status of the active connection, *ibaPDA* switches the connection. A change of the CPU status or the connection status of the currently inactive connection, does not cause a switch-over of the connection.

This is the coupling of two "normal" S7-Xplorer connections that *ibaPDA* switches in case of an error. It is not a so-called "fault tolerant connection". This is why there may be signal gaps of 1 to 2 seconds during a switching operation.

The connected controllers may be an S7-400H controller pair (fault tolerant) or two standard S7 controllers. A prerequisite is that the acquired operands occupy in both controllers identical addresses.

ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
S7-300	X	X	X	X	X	X	-	-	-
S7-400	X	X	X	X	X	X	-	-	-
S7-400H	X	X	X	X	X	X	-	-	-
S7-1200	X	-	-	-	-	-	-	-	-
S7-1500	X	-	-	-	-	-	-	-	-

Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector

Note

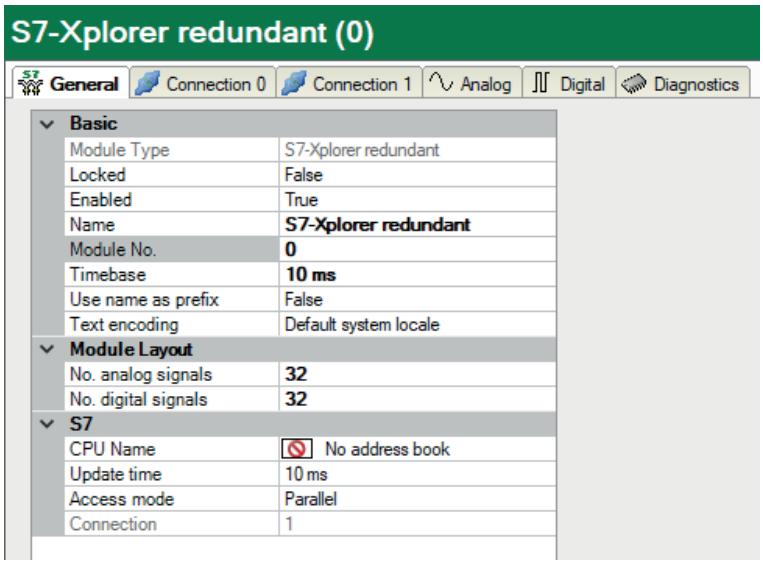
The redundancy module for the S7-1500H is not required. The controller provides a system-wide system IP address, which always allows access to the active CPU.

Note

A download of the HW Config is not possible while *ibaPDA* acquires data. Stop the *ibaPDA* data acquisition first.

ibaPDA supports the following operand types:

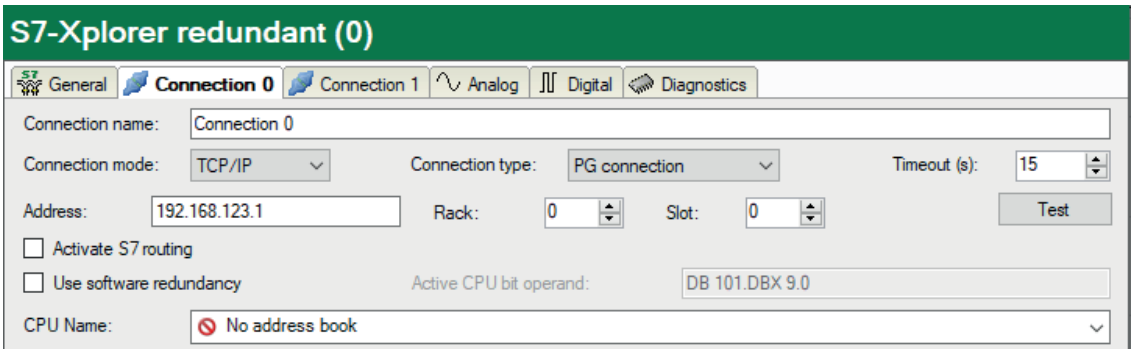
Selection via	Analog	Digital
S7-300 S7-400	EB, AB, MB, PEB, DBB EW, AW, MW, PEW, DBW ED, AD, MD, PED, DBD T, Z	E, A, M, DBX
S7-1200 S7-1500	EB, AB, MB, DBB EW, AW, MW, DBW ED, AD, MD, DBD	E, A, M, DBX



For more information on the module settings see ↗ *General module settings*, page 24.

Connection configuration

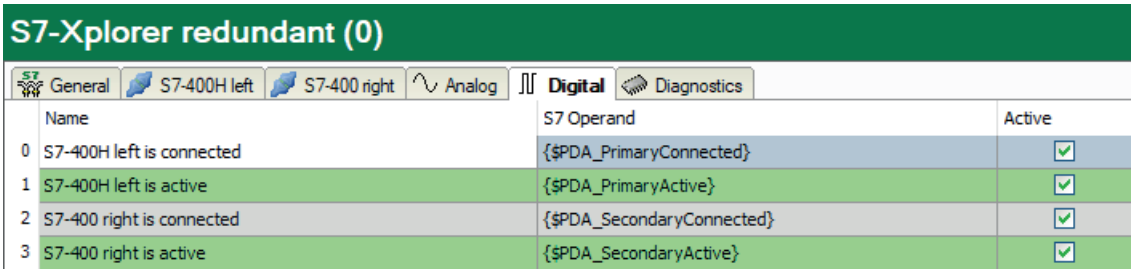
Configure both connections on the tabs *Connection 0* and *Connection 1*. You can change the name of the tabs in the *Connection name* fields.



For details concerning the module settings see ↗ *Module S7-Xplorer*, page 34.

The option *Enable Software Redundancy* refers to a Siemens software package that enables redundancy for standard automation systems of the S7-300 and S7-400. The status bit DB 101. DBX 9.0 indicates the state "Station is Master."

For diagnosis of the status of both connections, there are special signals available on the *Digital* tab. These occupy the first 4 signal positions:



Signal	Meaning
$PDA_PrimaryConnected	Connection 0 established
$PDA_PrimaryActive	Data acquisition active over connection 0
$PDA_SecondaryConnected	Connection 1 established
$PDA_SecondaryActive	Data acquisition active over connection 1

Table 2: Status signals of the redundant connection

One signal out of the two signals {PDA_PrimaryActive} and {PDA_SecondaryActive} is always TRUE even if none of the two connections is active.

3.3.9 Module S7-Xplorer SINUMERIK

This module establishes a connection via the standard network interface of the computer or an additional interface card to various S7 controllers installed in SINUMERIK devices.

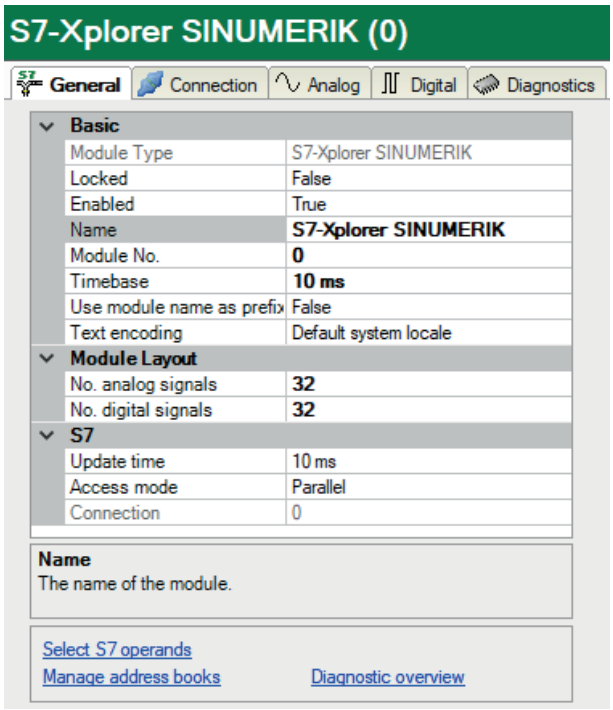
ibaPDA supports the following controllers, connection modes and selection methods:

SINUMERIK	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
808D	X	X	X	X	X	X	-	-	-
828D	X	X	X	X	X	X	-	-	-
840D	X	X	X	X	X	X	-	-	-
ONE	X	-	-	-	-	-	-	X	-

Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector

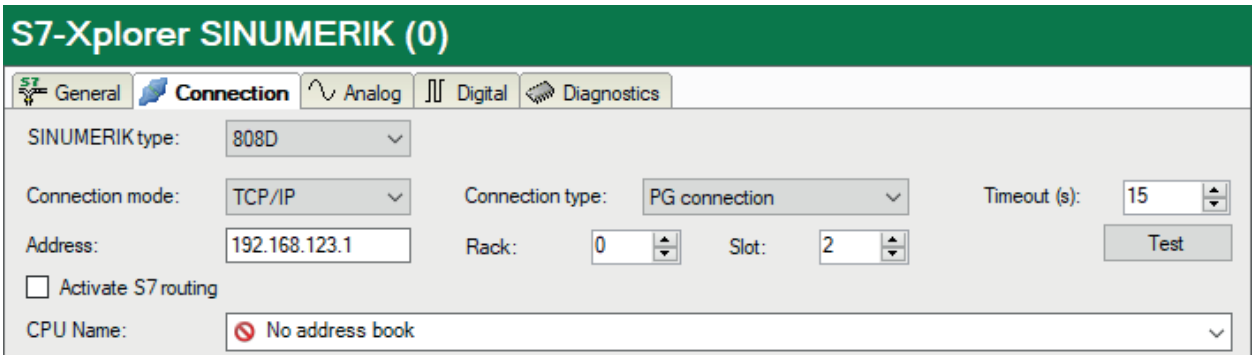
ibaPDA supports the following operand types:

Selection via	Analog	Digital
808D 828D 840D	EB, AB, MB, PEB, DBB EW, AW, MW, PEW, DBW ED, AD, MD, PED, DBD T, Z	E, A, M, DBX
ONE	EB, AB, MB, DBB EW, AW, MW, DBW ED, AD, MD, DBD	E, A, M, DBX



For more information on the module settings see ↗ *General module settings*, page 24.

Connection configuration



Select the *SINUMERIK type* first.

SINUMERIK type
Select the used SINUMERIK type. *ibaPDA* automatically sets the correct slot of the S7 controller. Depending on the SINUMERIK type, different connection modes are available.

Configure the other connection settings of the *S7-Xplorer SINUMERIK* module in the same way as the connection for an S7-Xplorer module:

↗ *Connection mode TCP/IP*, page 35
↗ *Connection mode PC/CP*, page 37
↗ *Connection mode TCP/IP S7-1x00*, page 41

3.3.10 Module S5 (LAN adapter)

This module establishes the connection to an AS511 interface of a SIMATIC S5 via the standard network interface of the computer or an additional network interface card. For this purpose, an additional interface converter for converting the signals (from AS511 to TCP/IP) is needed.

The following devices are tested and approved:

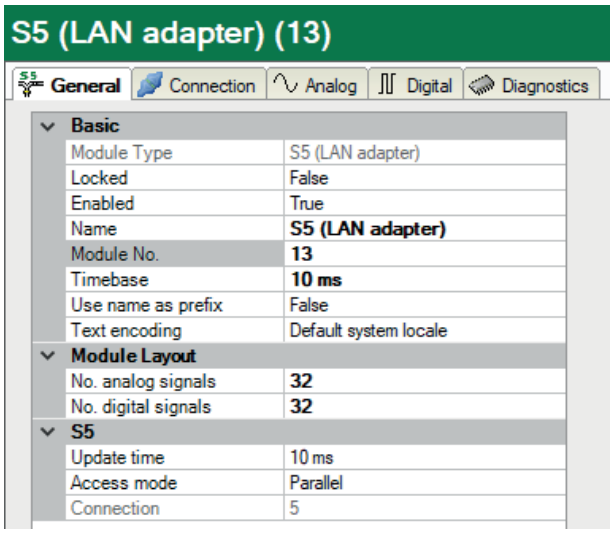
- ACCON-S5-LAN, Deltalogic
  Article no. 11800
- IBH Link S5++, IBHsoftec
  Article no. 20284

ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
S5	X	-	-	-	-	-	-	-	-

Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector

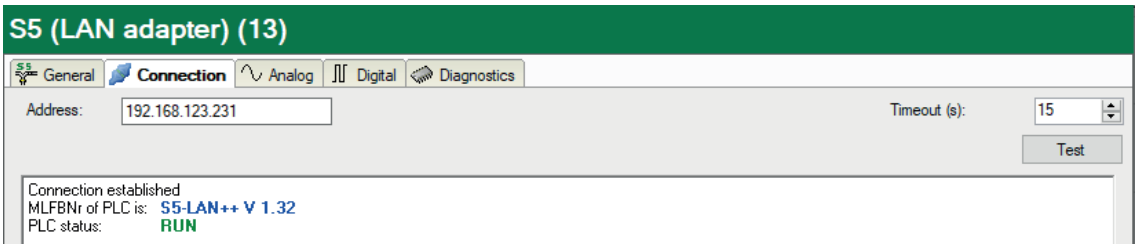
The module does not support any timers, counters or S markers.



For more information on the module settings see ↗ *General module settings*, page 24.

Connection configuration

Enter the IP address of the S5-LAN-adapter on the *Connection* tab in the *Address* field and click <Test>.



Note

As the AS511 interface is operated with only 9600 Baud, the data throughput that can be reached is correspondingly low.

Measured update times with a CPU948:

- 32 bytes: 220 ms
- 64 bytes: 260 ms
- 96 bytes: 290 ms
- 128 bytes: 330 ms
- 160 bytes: 450 ms

Other documentation

For more information about the usage of the adapters, see the specific product documentation.

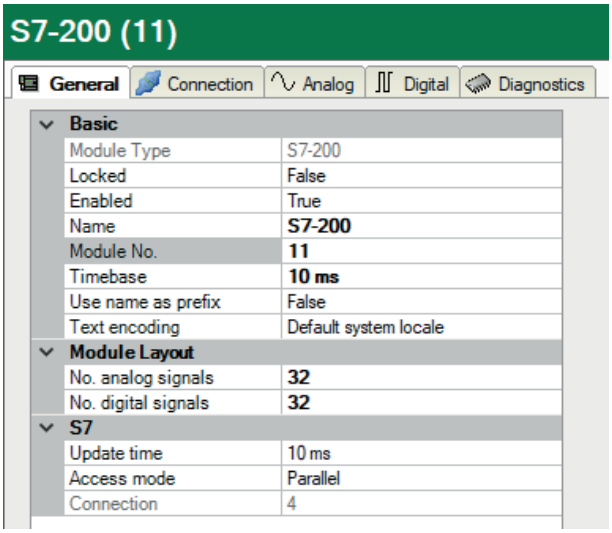
3.3.11 Module S7-200

Select this module for connecting a SIMATIC S7-200.

ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
S7-200	X	-	-	X	-	-	-	-	-

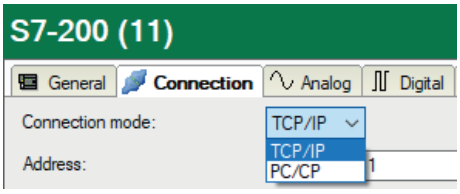
Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector



For more information on the module settings see ↗ *General module settings*, page 24.

Connection configuration

Select the desired connection mode in the *Connection* tab:

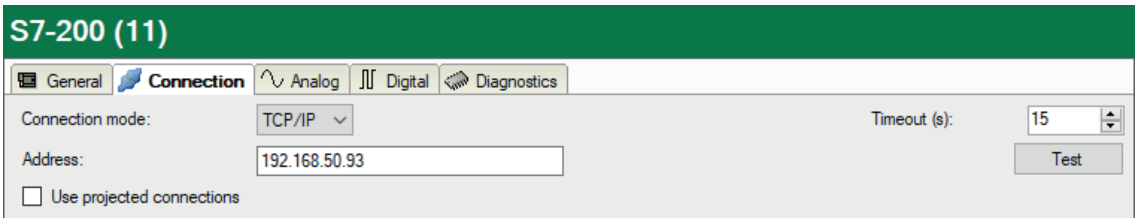


Note

If you access an S7-200, the programming device access is always used internally. However, this can only be used once simultaneously. This means that if you access the S7-200 that way, STEP 7-Micro/WIN cannot access at the same time. To avoid this, you can use a projected connection in *ibaPDA*.

3.3.11.1 Connection mode TCP/IP

This mode activates a connection via the standard network interface of the computer.

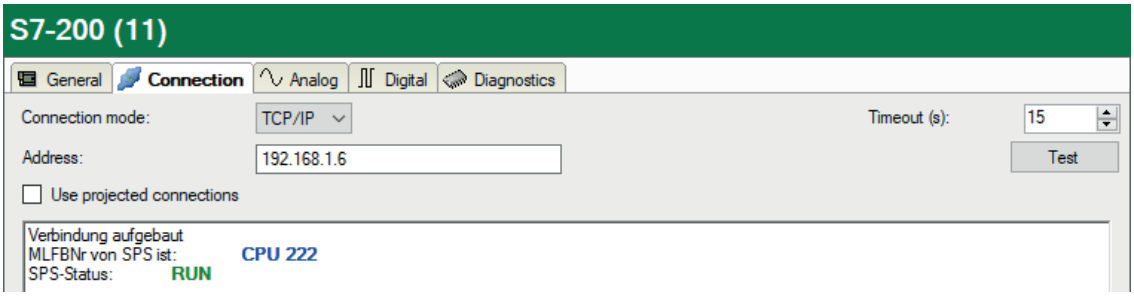


Connection mode
Selection of the TCP/IP connection mode

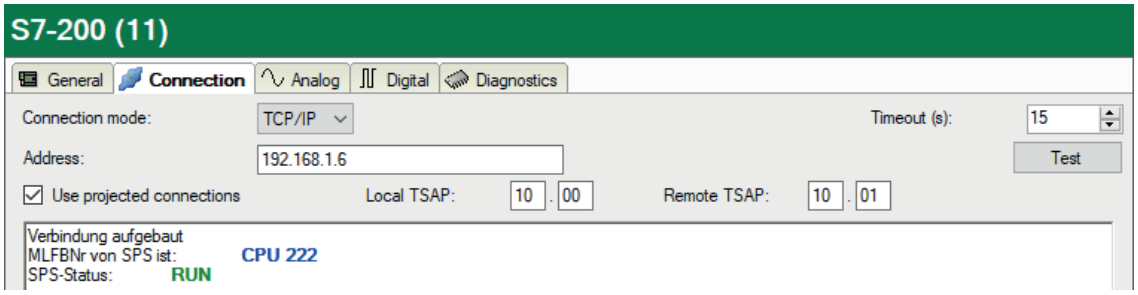
Timeout
Specify a value for the timeout in seconds for establishing the connection and for read access. If the time set here is exceeded, *ibaPDA* declares the controller as not accessible or not responsive.

Address
IP address of the controller

<Test>
ibaPDA tests the connection to the CPU and displays available diagnostic data.



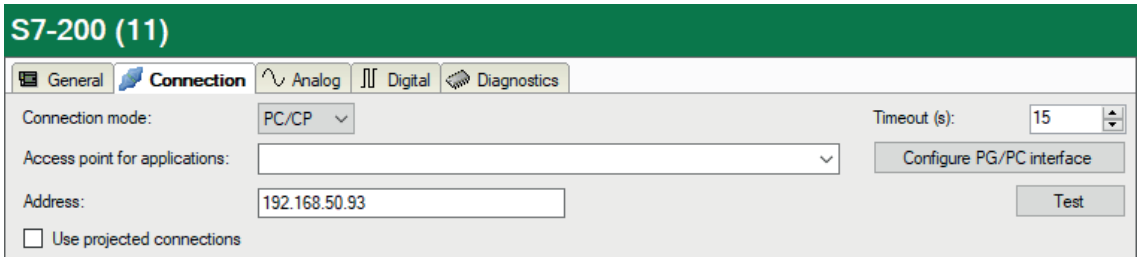
When you enable the *Use projected connections* option, you have to enter *Local TSAP* and *Remote TSAP* as well.



Local TSAP and remote TSAP
The values are described in the connection configuration in STEP 7-Micro/WIN (see ↗ *Configuration of S7-200*, page 16).

3.3.11.2 Connection mode PC/CP

This mode activates a connection over the interface cards of the computer that are configured using SIMATIC Net.



Connection mode
Selection of the PC/CP connection mode

Connection type
Selection of the connection type PG, OP, or other connections (determines which type of connection resource is occupied on the CPU).

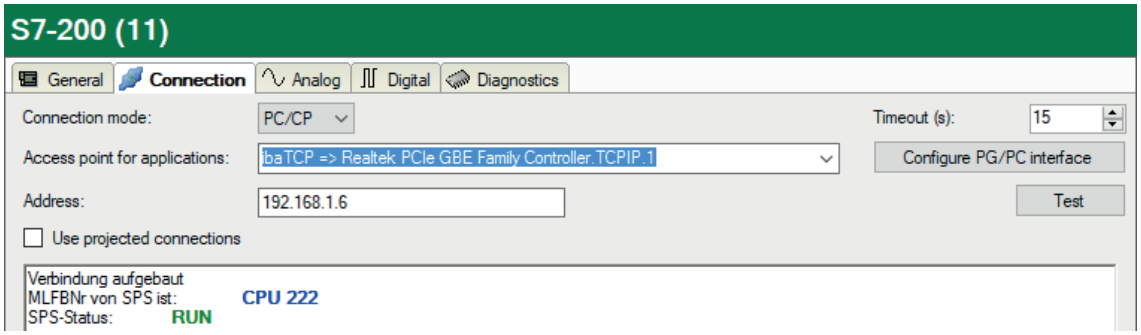
Timeout
Specify a value for the timeout in seconds for establishing the connection and for read access. If the time set here is exceeded, *ibaPDA* declares the controller as not accessible or not responsive.

Access point for applications
Selection of the access point that is to be used

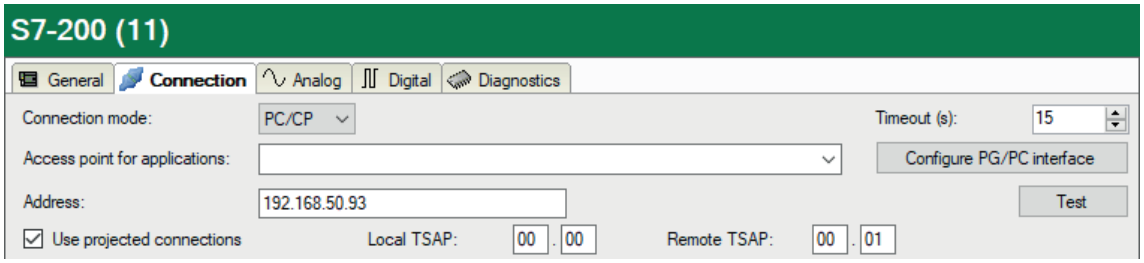
For more information on creating and configuring an access point, see ↗ *Setting PG/PC interface/defining new access point*, page 90.

Address
IP address of the controller

<Test>
ibaPDA tests the connection to the CPU and displays available diagnostic data.



When you enable the *Use projected connections* option, you have to enter *Local TSAP* and *Remote TSAP* as well.



Local TSAP and remote TSAP
The values are described in the connection configuration in STEP 7-Micro/WIN (see ↗ *Configuration of S7-200*, page 16).

3.3.12 Module LOGO!

Choose this module for connecting one of the following LOGO! controllers.

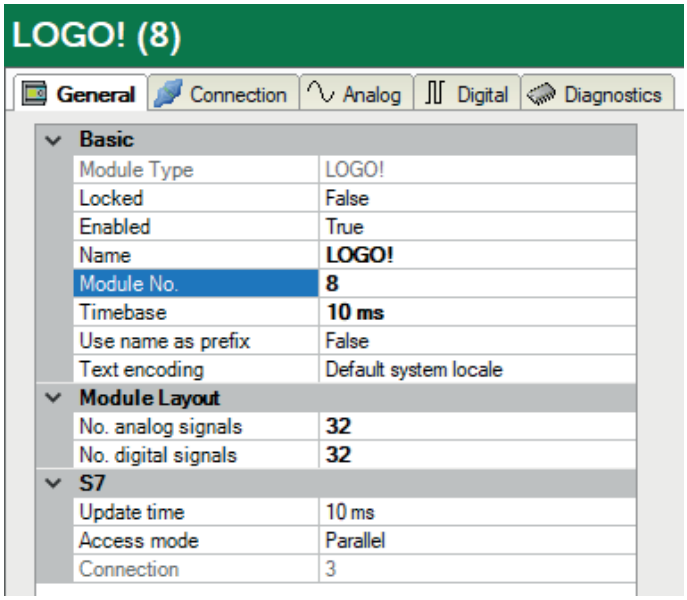
- SIMATIC LOGO! 0BA7
- SIMATIC LOGO! 0BA8

This module establishes the connection via the standard network interface of the computer or an additional network interface card.

ibaPDA supports the following controllers, connection modes and selection methods:

Controller	Connection mode and signal selection								
	TCP/IP			PC/CP			TCP/IP S7-1x00		
	ABS	SYM	CFC	ABS	SYM	CFC	ABS	SYM	CFC
LOGO!	X	-	-	-	-	-	-	-	-

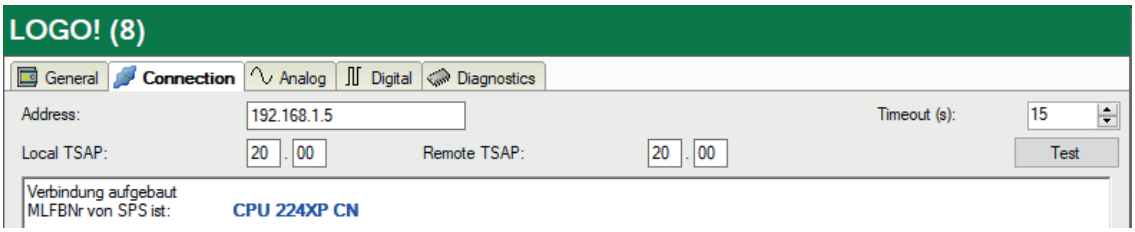
Explanation: **ABS**olute address, **SYM**bolic, **CFC** connector



For more information on the module settings see ↗ *General module settings*, page 24.

Connection configuration

Enter the IP address of the LOGO! in the *Address* field on the *Connection* tab and the *Local TSAP* and *Remote TSAP* as configured in the LOGO! (see ↗ *Configuration of LOGO!*, page 14). Click <Test>.



ibaPDA establishes a connection. The LOGO! found is displayed.

When connecting a LOGO! 0BA8, use "01.00" as local TSAP and remote TSAP.

Via the variable memory (VM), you can access all address ranges of a LOGO!. As interface for the S7 communication, the VM operates as local communication buffer for the data exchange via a connection configuration or data transfer configuration.

When connecting a LOGO! 0BA7, the variable storage range is also mapped on the DB1. This is why you can access the variables with S7 addresses (see table below). Via direct addressing, you can access also the marker bits, input bits and output bits of a LOGO!.

Overview of variable ranges (VM) LOGO! 0BA7

Block type	Address range	VM-address	S7-address	Direct address
Digital input	I1 – I24	923 – 925	DB1.DBX923.0 – 925.7	E0.0 – E2.7
Analog input	AI1 – AI8	926 – 941	DB1.DBW926 – 940	-
Digital output	Q1 – Q16	942 – 943	DB1.DBX942.0 – 943.7	A0.0 – A1.7
Analog output	AQ1 – AQ2	944 – 947	DB1.DBW944 – 946	-
Digital markers	M1 – M27	948 – 951	DB1.DBX948.0 – 951.2	M0.0 – M3.2
Analog markers	AM1 – AM16	952 – 983	DB1.DBW952 – 983	-

Overview of variable ranges (VM) LOGO! 0BA8

Block type	Address range	VM-address	S7-address	Direct address
Digital input	I	1024 – 1031	DB1.DBX1024.0 – 1032.7	E0.0 – E7.7
Analog input	AI	1032 – 1063	DB1.DBW1032 – 1062	-
Digital output	Q	1064 – 1071	DB1.DBX1064.0 – 1071.7	A0.0 – A7.7
Analog output	AQ	1072 – 1103	DB1.DBW1072 – 1102	-
Digital markers	M	1104 – 1117	DB1.DBX1104.0 – 1117.7	M0.0 – M13.7
Analog markers	AM	1118 – 1245	DB1.DBW1118 – 1244	-
Network input	NI	1246 – 1261	-	-
Analog network input	NAI	1262 – 1389	-	-
Network output	NQ	1390 – 1405	-	-
Analog network output	NAQ	1406 – 1469	-	-

3.3.13 Output module

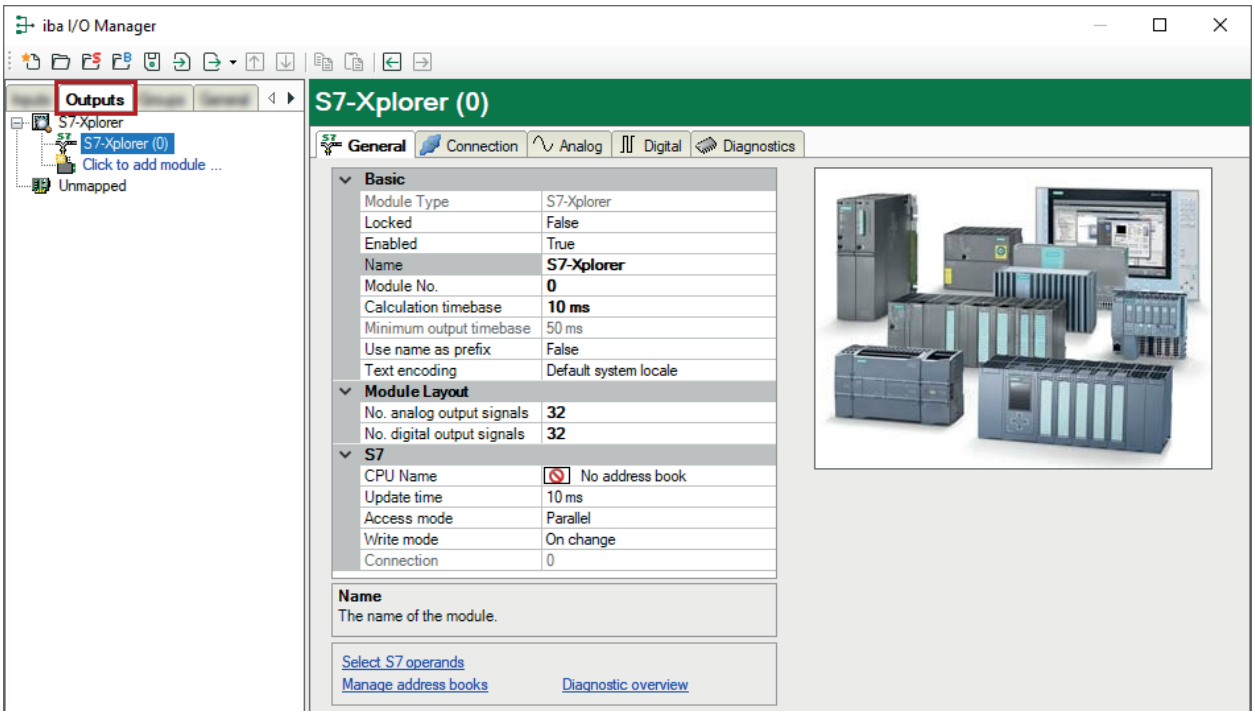
The output module is not an autonomous module, but rather an extension of the *S7-Xplorer* module. With the output module, you can write data from *ibaPDA* to a controller.

Note

Depending on the configured protection-level access of the S7 CPU, writing values into the CPU may be not possible.

Enable the S7 outputs in the interface settings, see ↗ *General interface settings*, page 21.

You can configure the module via the *Outputs* tab. You do not have to add it separately. The module is available as soon as you add an *S7-Xplorer* module in the *Inputs* register.



For more information on the module settings see ↗ *General module settings*, page 24.

Module-specific settings

Calculation timebase
Timebase (in ms) used for the calculation of the output values.

Technically, the calculation timebase is identical to the timebase of the input module. This means a change in the calculation timebase also changes the module timebase of the input side and vice versa!

The calculation timebase is not the same as the output timebase, with which the values are output!

Minimum output timebase
Timebase with which the outputs can be updated as quickly as possible.

The value is acquired automatically by the system based on the current I/O configuration and is only displayed here. The output timebase results from the smallest common multiple of all module timebases or is at least 50 ms.

Number of analog output signals
Define the number of configurable analog signals in the signal table. The default value is 32 for each. The maximum value is 1000. The signal table is adjusted accordingly.

Digital signals are not supported for the output module.

Send mode
Determines when new data is written to the controller:

- Cyclic: Data is written cyclically at the set update time.
- On change: Data is written each time the signal data is changed.
- On trigger: Data is written with every rising edge of the trigger signal.

All signals of a module are always written, regardless of the write mode.

Trigger signal
This field only appears when the "on trigger" send mode is selected. Select here a digital signal. A rising edge on this digital signal writes the signal values taken at the time of the rising edge.

Signal configuration

You can configure the signals to be output in each case via the expression editor. Open the expression editor via the <fx> button in each signal row.

Other documentation

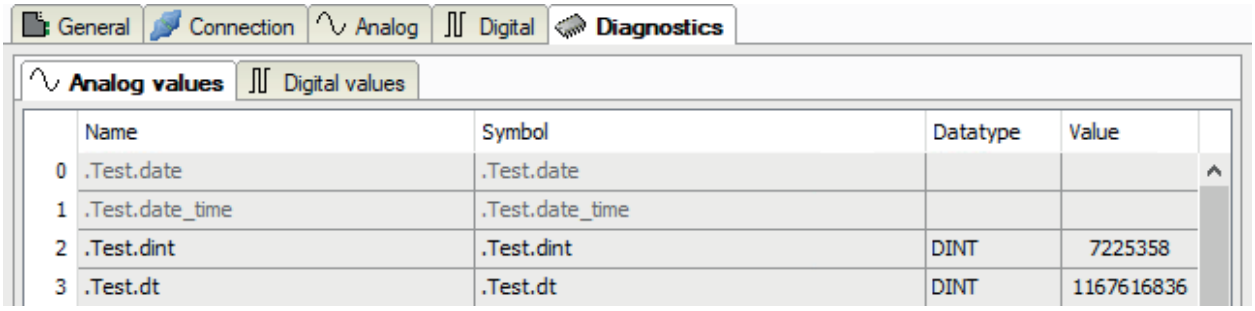
Further information about the expression editor can be found in the *ibaPDA* manual, part 4.

Note

ibaPDA reads and writes all signals for an Xplorer module via a common connection. Therefore, the total number of configured signals influences the update time.

3.3.14 Module diagnostics

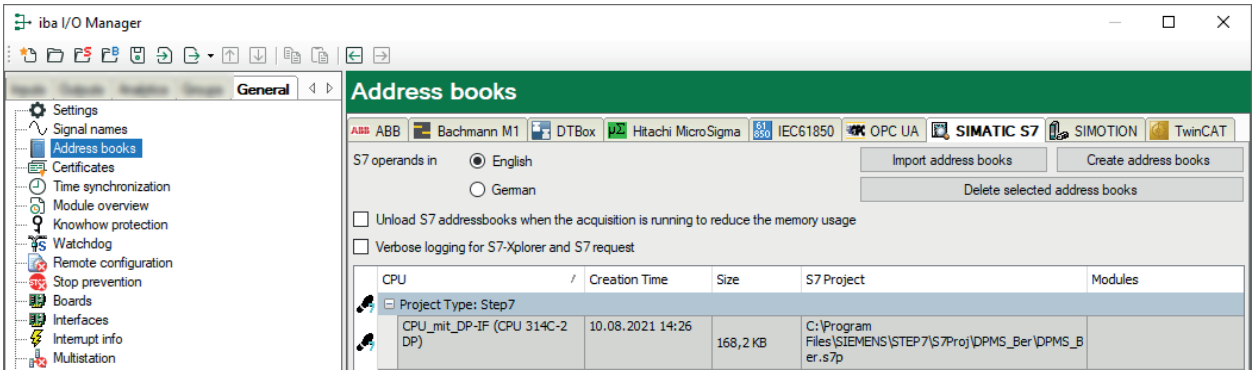
After applying the configuration the actual values of the analog and digital signals are displayed in the *Diagnostics* tab of the relevant module.



Inactive signals are grayed out.

3.3.15 Address books

The address books for SIMATIC S7 controllers are created and managed across modules. You can use one address book in more than one module.



There are different types of address books for the different S7 project types:

- STEP 7: SIMATIC Manager project
  (not for S7-Xplorer modules with connection mode TCP/IP S7-1x00)
- TIA Portal: TIA Portal Project

S7 operands in English/German
Here you can choose the language in which the S7 operands will later be available when browsing through the signal tables.

<Create address books>
This button opens the "S7 address book generator" dialog. You can select the source directory of an S7 project for creating the S7 address book. This can be a local or network drive.

<Import address books>
Import address books which are already available as ZIP files.

<Delete selected address books>
Delete address books from the *ibaPDA* server's directory.

Unload S7 address book when the acquisition is running to reduce the memory usage
By enabling this option, the address book is outsourced to the hard disk during the acquisition in order to free up the main memory for the acquisition.

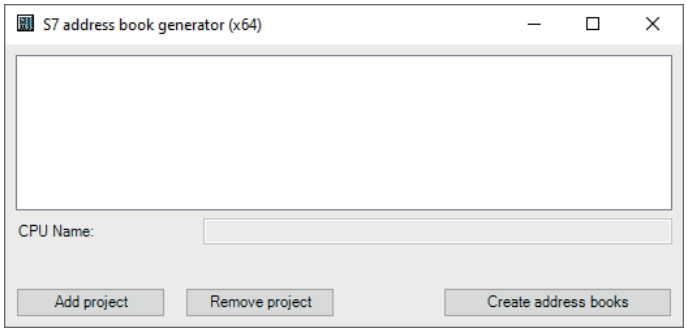
Table
List of all address books that are currently available in the system with name, creation date, path of STEP 7 project or IP address of the CPU in case of online generated address books and location where the address book is used.

3.3.15.1 Creating address books offline from S7 project

For creating an address book, the S7 project has to be available. For the subsequent use, this is not necessary.

You create an address book using the S7 address book generator.

S7 address book generator



CPU Name
Name of the CPU

Step 7 HW Config export
A HW config export file can be selected as an option (useful when using an iba bus monitor in sniffer mode)

Comment language
Selection of the language that is to be imported for comment texts (only available for SIMATIC TIA portal projects)

<Add project>
Adds a new project to the list

Remove project>
Removes the marked project from the list

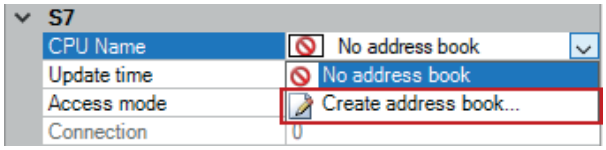
<Create address books>
Creates address books from the selected projects

Note

You can overwrite the entry in the *CPU Name* field. Thus, you can assign a unique name for the CPU that differs from that in the STEP 7 project. This is especially interesting when you use several STEP 7 projects in which the CPUs have the same name.

Creating address books via the S7 address book generator

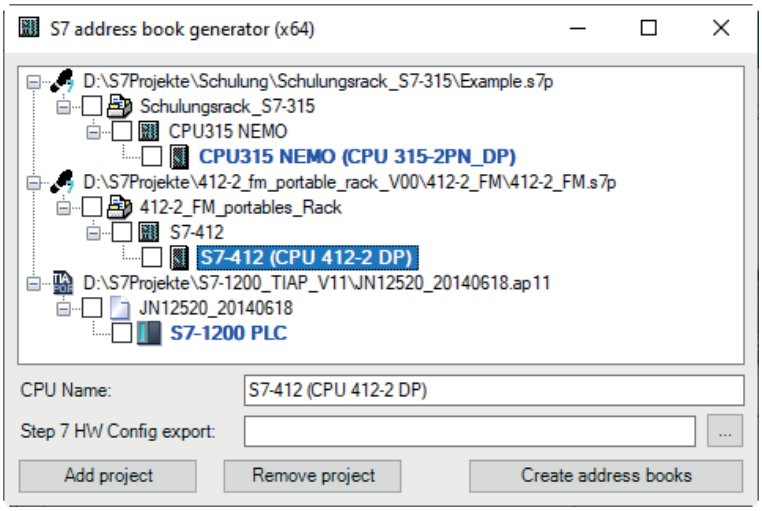
1. Open the S7 address book generator by one of the following ways:
   - On the *General* tab *Address books* node via the <Create address books> button
   - In the module configuration on the *General* tab under *S7 – CPU Name*: Select *Create address book* in the drop-down menu.



2. Click on the <Add project>.
3. Select a project file in the file browser

→ Now, the STEP 7 project with all configured CPUs is displayed.

4. Mark the CPUs you want to create the address books from and click on <Create address books>.



Note

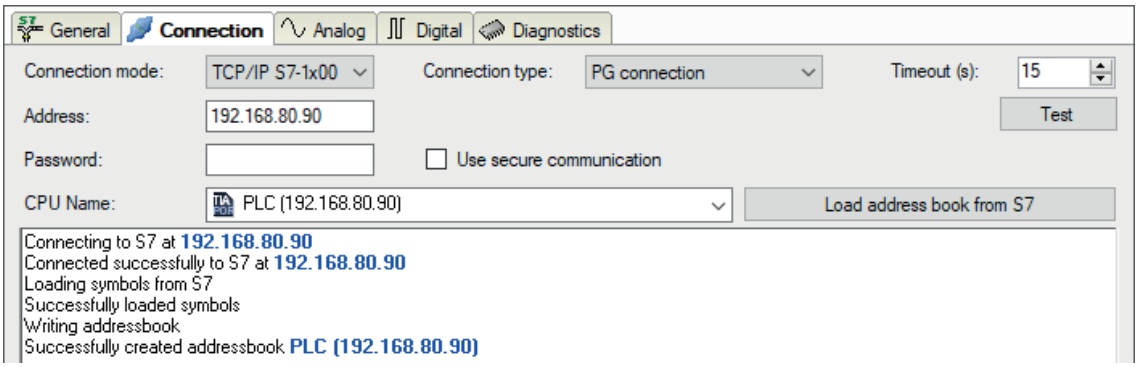
When address books of these projects are generated, TIA Portal projects must be compiled, saved and closed.

3.3.15.2 Creating address books online from S7-1200/1500 CPU

You can create online address books out of an S7-1200 or S7-1500 CPU if connection mode *TCP/IP S7-1x00* has been selected. The address data are read directly from the CPU. Accessing the S7 project is not necessary.

Click on <Load address book from S7> to load the address book.

The CPU name of the address book is given automatically.



Address books that have been created online also contain operand type address information and hence can be used in connection mode TCP/IP as well. Just change the connection mode after creating the address book.

3.3.16 Connection with SIMATIC WinAC (WinLC RTX)

For using *ibaPDA-Interface-S7-Xplorer* with a Soft-PLC SIMATIC WinAC RTX, the following items have to be observed:

- Operating *ibaPDA* and WinAC RTX on the same computer is generally permitted.
- Not permitted is using iba interface boards (e.g. *ibaFOB* or *ibaCom-L2B*) on a WinLC RTX system. Here, high loads of the PCI bus from both sides might lead to instabilities.
- When using the module *S7-Xplorer* with the connection mode TCP/IP, always enter the value 0 in the *Rack* and *Slot* field in the I/O Manager.
- When using the module *S7-Xplorer* with the connection mode PC/CP, the MPI or PROFIBUS address in the *Address* field, always enter the value 0 in the *Rack* and *Slot* field in the I/O Manager.

3.3.17 Connection with S7-PLCSIM via ibaPDA-S7-Xplorer Proxy

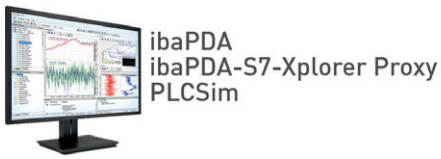
You can use the *ibaPDA-Interface-S7-Xplorer* also with a CPU simulated by SIMATIC S7-PLCSIM. To do so, you must use *ibaPDA-S7-Xplorer Proxy*. This proxy acts as data gateway between *ibaPDA* and SIMATIC S7-PLCSIM.

Prerequisites

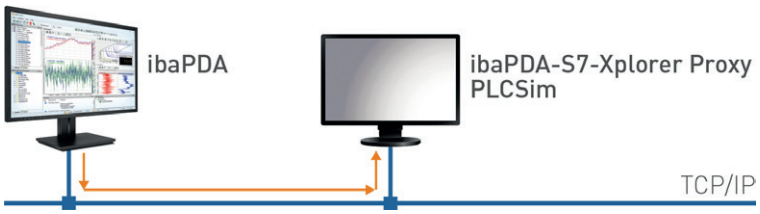
- S7-PLCSIM version 5.4.3.0 or higher
- *ibaPDA-S7-Xplorer Proxy*

System topologies

- Single user
  All components run running on one computer.



- Distributed installation
  ibaPDA is not installed on the computer with S7-PLCSIM and *ibaPDA-S7-Xplorer Proxy*.



  Access of *ibaPDA* to the proxy takes place via TCP/IP.

If you want to connect a CPU simulated with S7-PLCSIM, you have to install and run *ibaPDA-S7-Xplorer Proxy* on the same computer. You can install *ibaPDA-S7-Xplorer Proxy* as an option during the installation of *ibaPDA*.

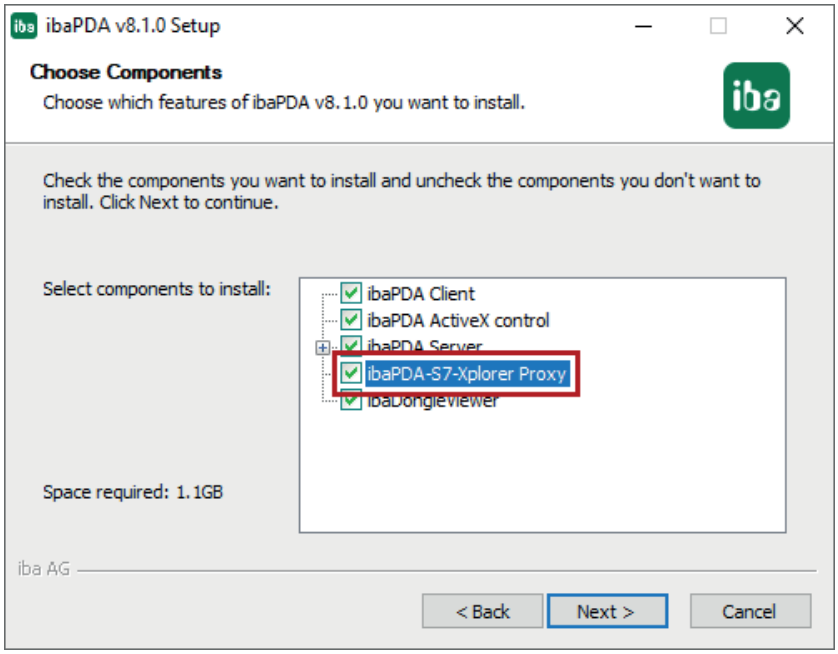
3.3.17.1 Installation and application of ibaPDA-S7-Xplorer Proxy

ibaPDA-S7-Xplorer Proxy is a standalone application that you can install with *ibaPDA*. After that, you can open the application and configure *ibaPDA-S7-Xplorer Proxy*.

Installation with ibaPDA

To use the *ibaPDA-S7-Xplorer Proxy*, observe the following points when installing *ibaPDA*:

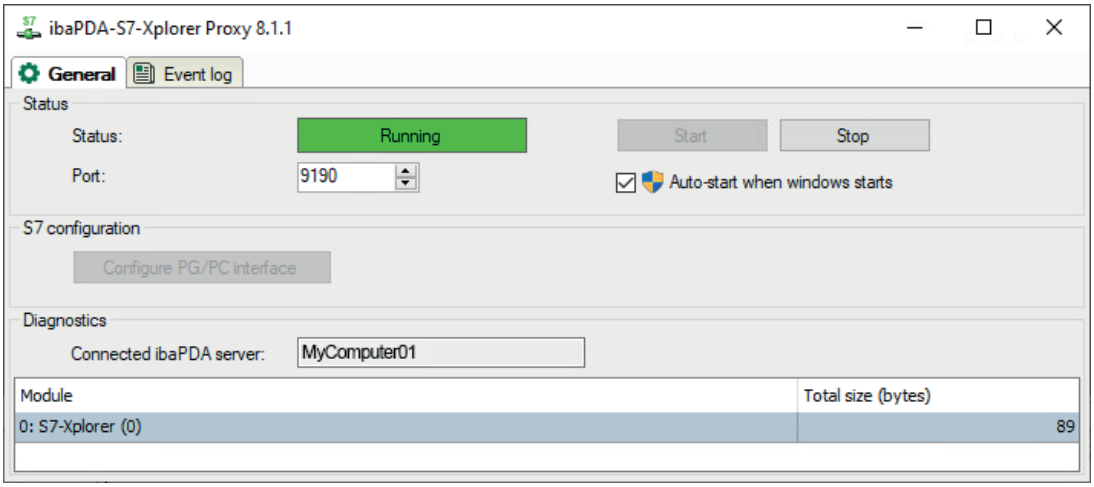
- Install *ibaPDA* as usual. However, for the components, also select the *ibaPDA-S7-Xplorer Proxy*.



- Select the default setting *Local system account*.

→ The *ibaPDA-S7-Xplorer Proxy* runs as stand-alone application.

ibaPDA-S7-Xplorer Proxy application



Status

Current operating state

Port
Used port

<Start>, Stop>
Manually starting and stopping the *ibaPDA-S7-Xplorer Proxy*

Auto-start when Windows starts
ibaPDA-S7-Xplorer Proxy is automatically started when starting the system and runs in the background.

S7 configuration
Starts the configuration dialog to set the SIMATIC PG/PC interface.

Diagnostics
Shows the *ibaPDA* server currently connected.

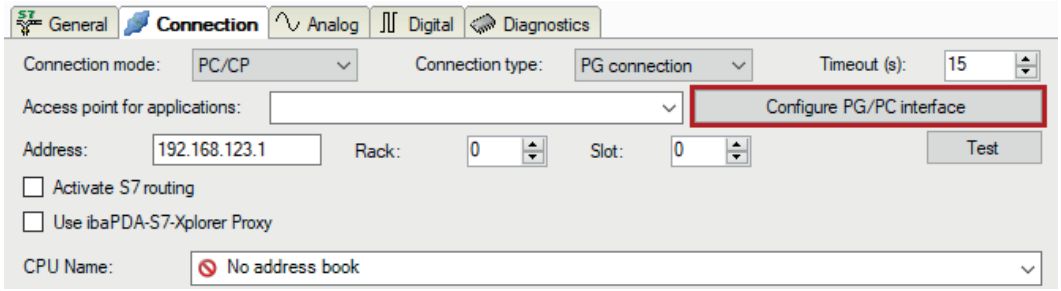
Module
Shows the modules currently active.

3.3.17.2 Configuring ibaPDA-S7-Xplorer Proxy in ibaPDA

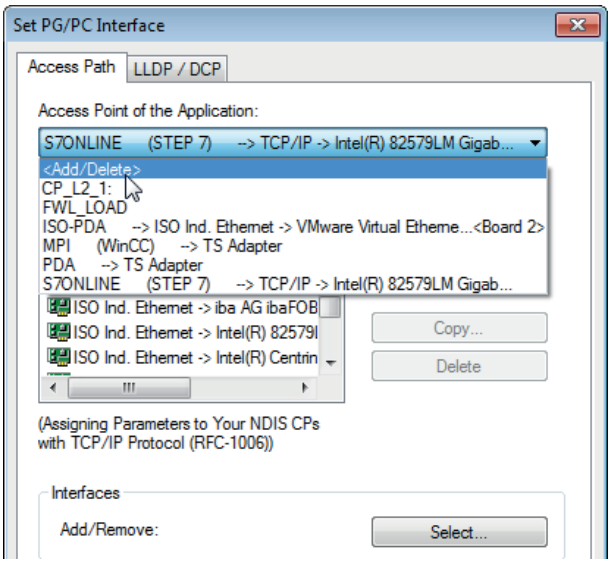
1. Add an S7-Xplorer module in the I/O Manager of *ibaPDA*.

   Set the connection mode *PC/CP* and the connection type *PG connection*.

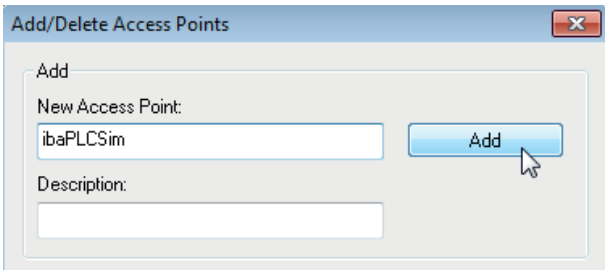
2. With the <Configure PG/PC interface> button, open the dialog for setting the PG/PC interface to configure an access point of the application.



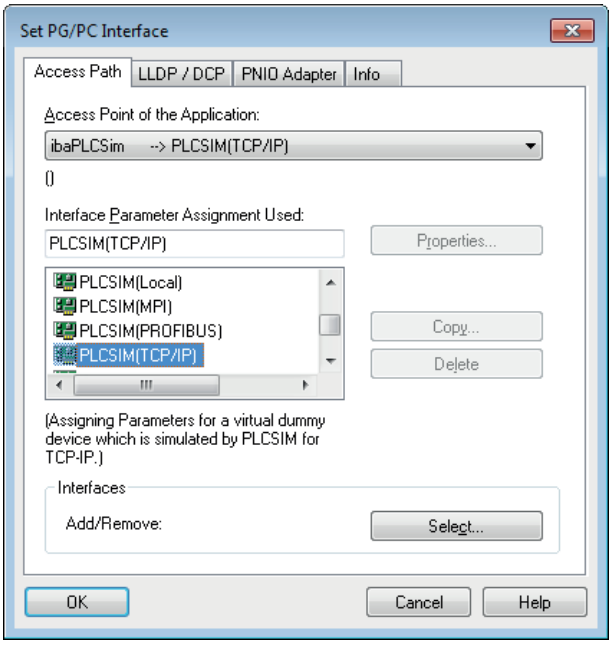
3. In the drop-down menu *Access point of the application*, select the line *<Add/Delete>*.



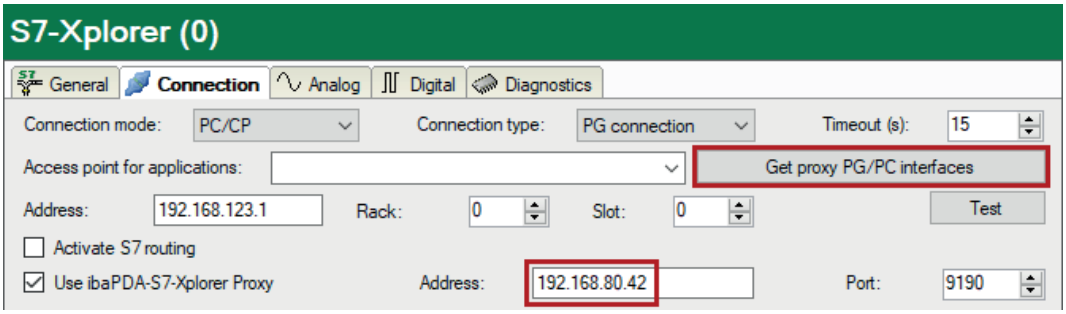
4. Specify the new access point: Enter a name, e.g. "ibaPLCSim", and optionally a description for better understanding.
   Confirm the entries with <Add> and <Close>.



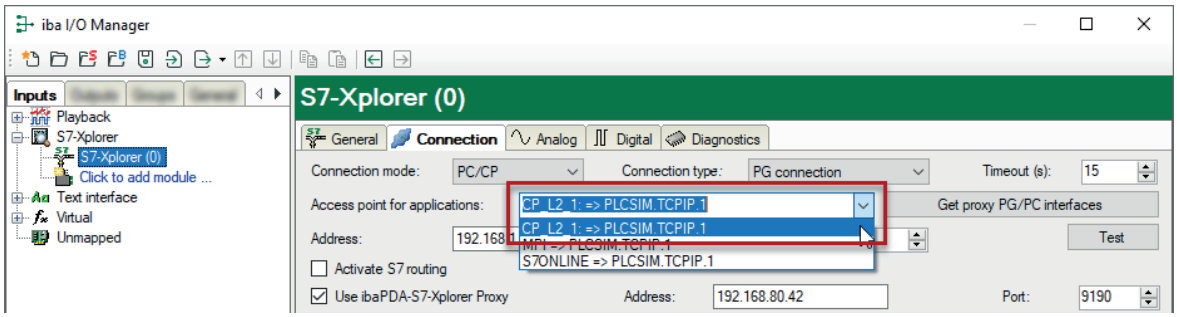
5. Assign an interface parameter assignment to this access point, e.g. "PLCSIM.TCPIP.1".



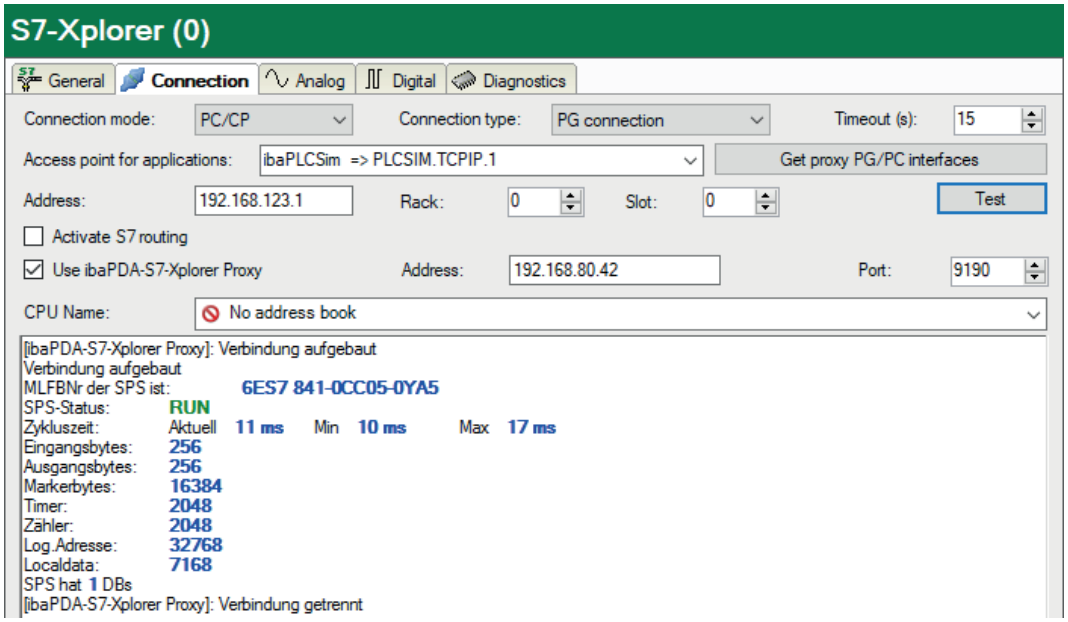
6. Close the configuration with <OK>.
7. In the *ibaPDA* module, enable the *Use ibaPDAS7-Xplorer Proxy* option.
8. If *ibaPDA-S7-Xplorer Proxy* runs on another computer, adjust the address accordingly and call the proxy PG/PC interfaces.



9. Select the previously created access point from the list.



10. Check the connection with the <Test> button.

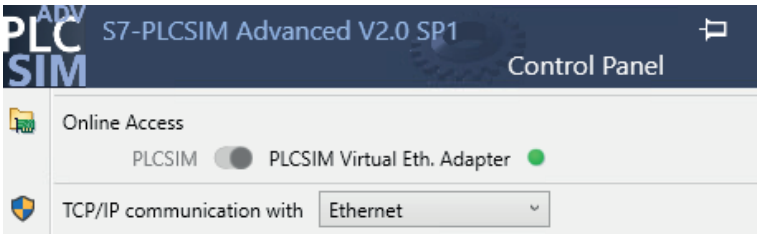


3.3.18 Connection with S7-PLCSIM Advanced

You can also use the interface *ibaPDA-Interface-S7-Xplorer* with a CPU simulated by SIMATIC S7-PLCSIM Advanced. Depending on the selected communication path, it may be necessary to use the *ibaPDA-S7-Xplorer Proxy*. This acts as a data gateway between *ibaPDA* and SIMATIC S7-PLCSIM Advanced.

SIMATICS S7-PLCSIM Advanced offers two different methods of online access:

- PLCSIM
- PLCSIM Virtual Ethernet Adapter



When using "PLCSIM", proceed as described in chapter
↗ *Connection with S7-PLCSIM via ibaPDA-S7-Xplorer Proxy*, page 64.

When using "PLCSIM Virtual Ethernet Adapter", you can also use the connection mode *TCP/IP* or *TCP/IP S7-1x00* in *ibaPDA*. Then the use of the *ibaPDA-S7-Xplorer Proxy* is not necessary.

In particular, this enables access to "optimized blocks" and online address book generation directly from the CPU.

Other documentation

Further information can be found in the documentation for SIMATIC S7-1500 S7-PLCSIM-Advanced.

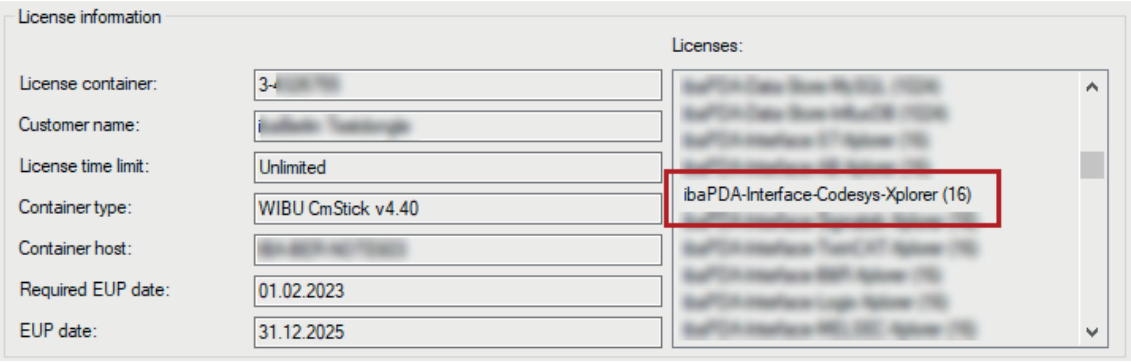
https://support.industry.siemens.com/cs/de/en/view/109760835

4 Diagnostics

4.1 License

If the interface is not displayed in the signal tree, you can either check in *ibaPDA* in the I/O Manager under *General – Settings* or in the *ibaPDA* service status application whether your license for the interface *ibaPDA-Interface-S7-Xplorer* has been properly recognized. The number of licensed connections is shown in brackets.

The figure below shows the license for the *Codesys Xplorer* interface as an example.



4.2 Visibility of the interface

If the interface is not visible despite a valid license, it may be hidden.

Check the settings in the *General* tab in the *Interfaces* node.

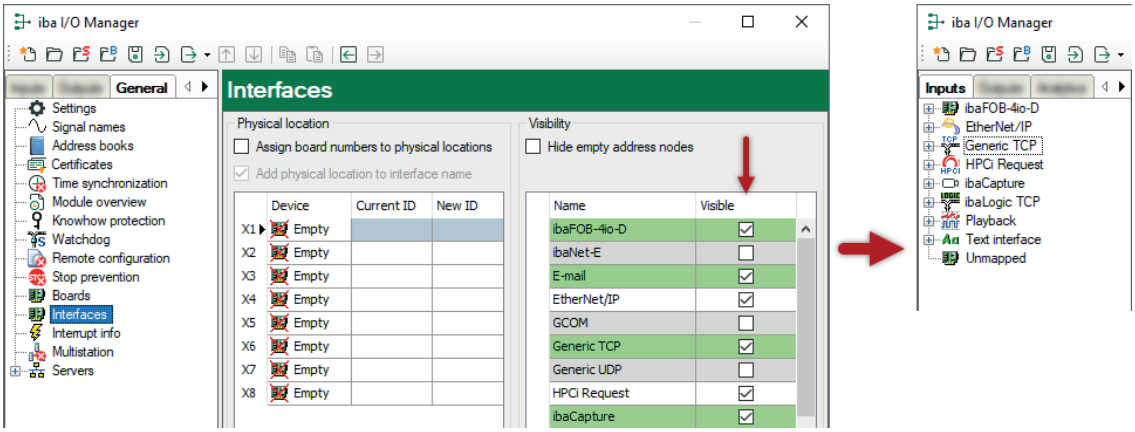
Visibility

The table *Visibility* lists all the interfaces that are available either through licenses or installed cards. These interfaces can also be viewed in the interface tree.

You can hide or display the interfaces not required in the interface tree by using the checkbox in the *Visible* column.

Interfaces with configured modules are highlighted in green and cannot be hidden.

Selected interfaces are visible, the others are hidden:



4.3 Log files

If connections to target platforms or clients have been established, all connection-specific actions are logged in a text file. You can open this (current) file and, e.g., scan it for indications of possible connection problems.

You can open the log file via the button <Open log file>. The button is available in the I/O Manager:

- for many interfaces in the respective interface overview
- for integrated servers (e.g. OPC UA server) in the *Diagnostics* tab.

In the file system on the hard drive, you can find the log files of the *ibaPDA* server (…\ProgramData\iba\ibaPDA\Log). The file names of the log files include the name or abbreviation of the interface type.

Files named `interface.txt` are always the current log files. Files named `Interface_yyyy_mm_dd_hh_mm_ss.txt` are archived log files.

Examples:

- `ethernetipLog.txt` (log of EtherNet/IP connections)
- `AbEthLog.txt` (log of Allen-Bradley Ethernet connections)
- `OpcUAServerLog.txt` (log of OPC UA server connections)

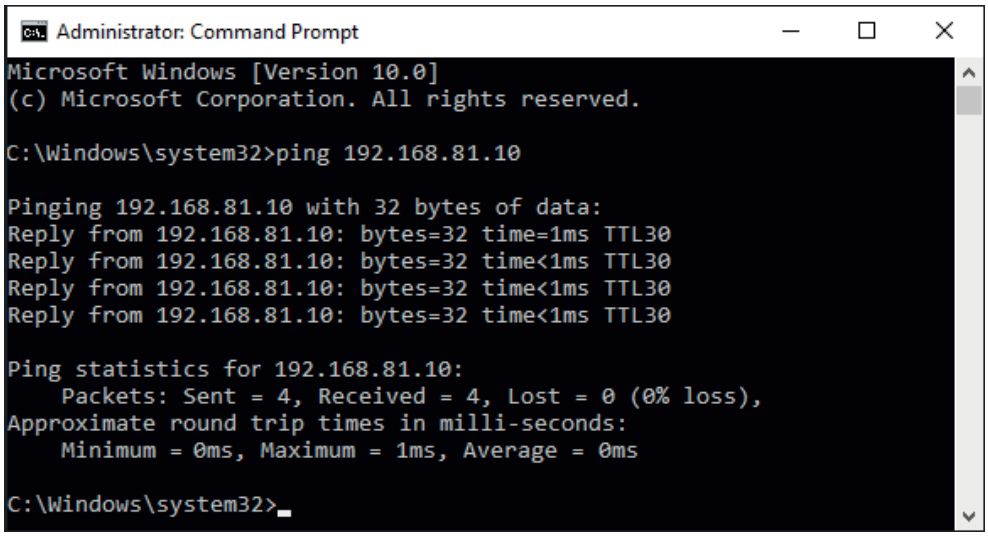
4.4 Connection diagnostics with PING

PING is a system command with which you can check if a certain communication partner can be reached in an IP network.

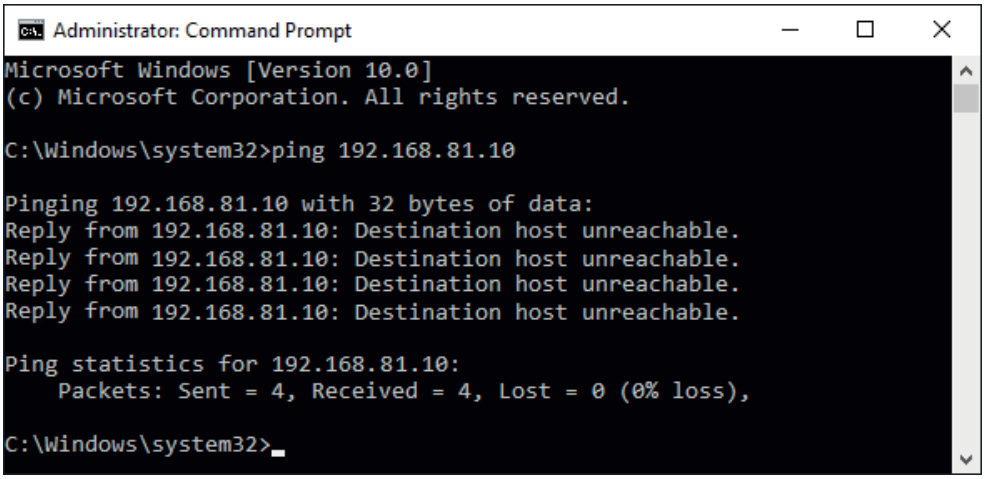
1. Open a Windows command prompt.

2. Enter the command "ping" followed by the IP address of the communication partner and press <ENTER>.

→ With an existing connection you receive several replies.

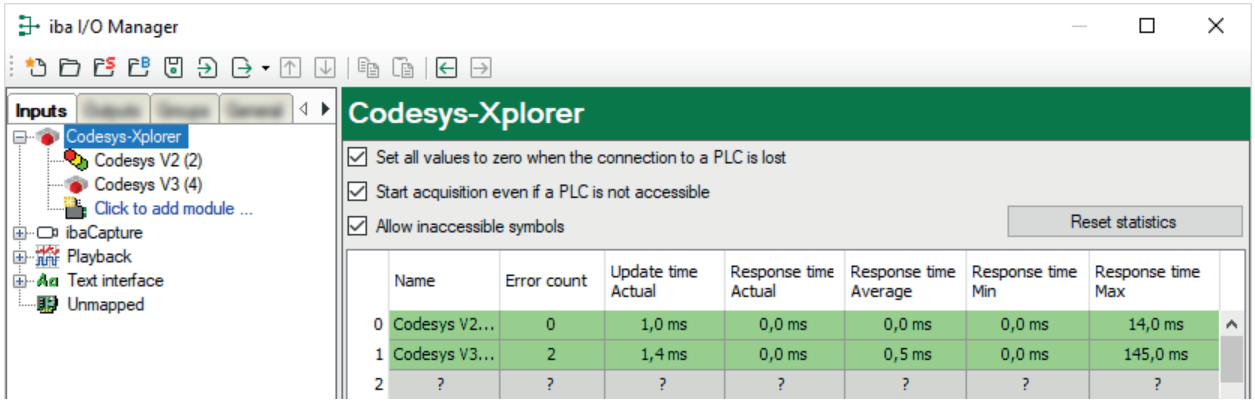


→ With no existing connection you receive error messages.



4.5 Connection table

For every Ethernet-based interface, there is a table available in the I/O Manager which shows the status of each connection. Each line represents one connection. The following figure shows, as an example, the connection table of the Codesys-Xplorer interface:



The connected target systems (controllers) are identified by their name or IP address in the first (left) column.

Depending on the interface type the table shows error counters, read counters and/or data sizes, as well as the cycle times, refresh times and/or update times of the different connections during the data acquisition.

Click the <Reset statistics> button to reset the error counters and the calculation of the response times.

Additional information is provided by the background color of the table rows:

Color	Meaning
Green	The connection is OK and the data are read.
Yellow	The connection is OK, however the data update is slower than the configured update time.
Red	The connection has failed.
Gray	No connection configured.

4.6 Diagnostic modules

Diagnostic modules are available for most Ethernet based interfaces and Xplorer interfaces. Using a diagnostic module, information from the diagnostic displays (e.g. diagnostic tabs and connection tables of an interface) can be acquired as signals.

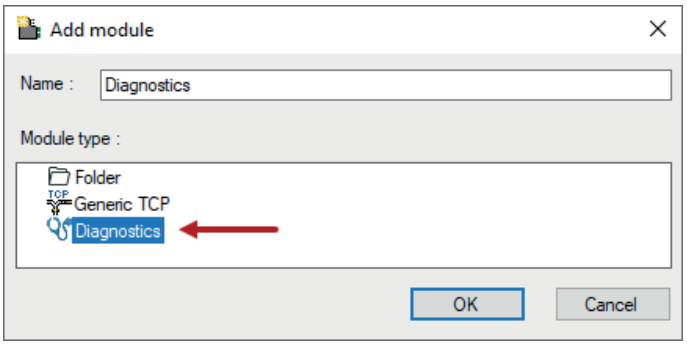
A diagnostic module is always assigned to a data acquisition module of the same interface and supplies its connection information. By using a diagnostic module, you can record and analyze the diagnostic information continuously in the *ibaPDA* system.

Diagnostic modules do not consume any license connections because they do not establish their own connection but refer to another module.

Example for the use of diagnostic modules:

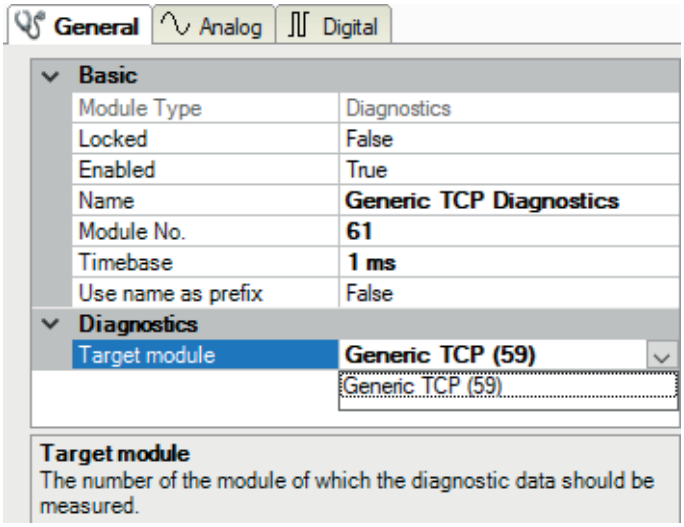
- A notification can be generated, whenever the error counter of a communication connection exceeds a certain value or the connection gets lost.
- In case of a disturbance, the current response times in the telegram traffic may be documented in an incident report.
- The connection status can be visualized in *ibaQPanel*.
- You can forward diagnostic information via the SNMP server integrated in *ibaPDA* or via OPC DA/UA server to superordinate monitoring systems like network management tools.

In case the diagnostic module is available for an interface, a "Diagnostics" module type is shown in the *Add module* dialog (example: Generic TCP).



Module settings diagnostic module

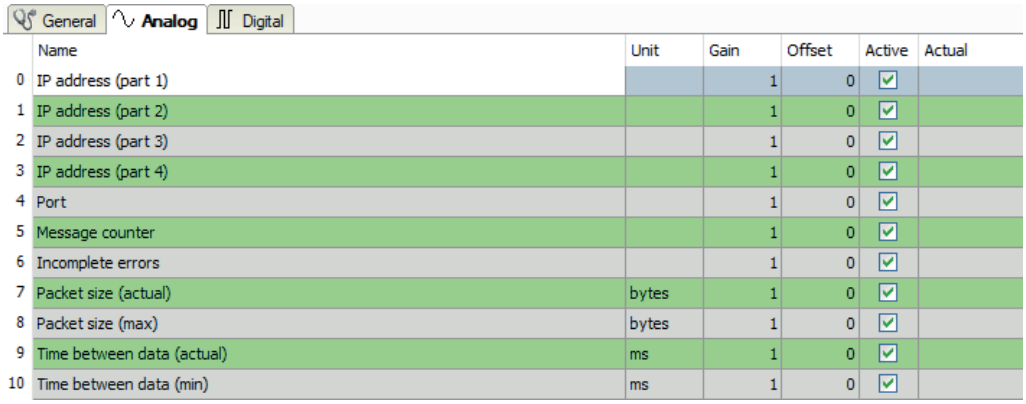
For a diagnostic module, you can make the following settings (example: Generic TCP):



The basic settings of a diagnostic module equal those of other modules.

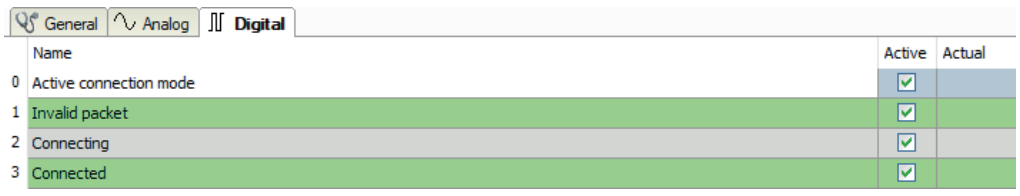
There is only one setting which is specific for the diagnostic module: the target module.

By selecting the target module, you assign the diagnostic module to the module on which you want to acquire information about the connection. You can select the supported modules of this interface in the drop-down list of the setting. You can assign exactly one data acquisition module to each diagnostic module. When having selected a module, the available diagnostic signals are immediately added to the *Analog* and *Digital* tabs. It depends on the type of interface, which signals exactly are added. The following example lists the analog values of a diagnostic module for a Generic TCP module.



	Name	Unit	Gain	Offset	Active	Actual
0	IP address (part 1)		1	0	☑	
1	IP address (part 2)		1	0	☑	
2	IP address (part 3)		1	0	☑	
3	IP address (part 4)		1	0	☑	
4	Port		1	0	☑	
5	Message counter		1	0	☑	
6	Incomplete errors		1	0	☑	
7	Packet size (actual)	bytes	1	0	☑	
8	Packet size (max)	bytes	1	0	☑	
9	Time between data (actual)	ms	1	0	☑	
10	Time between data (min)	ms	1	0	☑	

For example, the IP (v4) address of a Generic TCP module (see fig. above) will always be split into 4 parts derived from the dot-decimal notation, for better reading. Also other values are being determined, as there are port number, counters for telegrams and errors, data sizes and telegram cycle times. The following example lists the digital values of a diagnostic module for a Generic TCP module.



	Name	Active	Actual
0	Active connection mode	☑	
1	Invalid packet	☑	
2	Connecting	☑	
3	Connected	☑	

Diagnostic signals

Depending on the interface type, the following signals are available:

Signal name	Description
Active	Only relevant for redundant connections. Active means that the connection is used to measure data, i.e. for redundant standby connections the value is 0. For normal/non-redundant connections, the value is always 1.
Buffer file size (actual/avg/max)	Size of the file for buffering statements
Buffer memory size (actual/avg/max)	Size of the memory used by buffered statements
Buffered statements	Number of unprocessed statements in the buffer
Buffered statements lost	Number of buffered but unprocessed and lost statements
Connected	Connection is established
Connected (in)	A valid data connection for the reception (in) is available
Connected (out)	A valid data connection for sending (out) is available
Connecting	Connection being established
Connection attempts (in)	Number of attempts to establish the receive connection (in)
Connection attempts (out)	Number of attempts to establish the send connection (out)
Connection ID O->T	ID of the connection for output data (from the target system to *ibaPDA*). Corresponds to the assembly instance number
Connection ID T->O	ID of the connection for input data (from *ibaPDA* to target system). Corresponds to the assembly instance number
Connection phase (in)	Status of the ibaNet-E data connection for reception (in)
Connection phase (out)	Status of the ibaNet-E data connection for sending (out)
Connections established (in)	Number of currently valid data connections for reception (in)
Connections established (out)	Number of currently valid data connections for sending (out)
Data length	Length of the data message in bytes
Data length O->T	Size of the output message in byte
Data length T->O	Size of the input message in byte
Destination IP address (part 1-4) O->T	4 octets of the IP address of the target system Output data (from target system to *ibaPDA*)
Destination IP address (part 1-4) T->O	4 octets of the IP address of the target system Input data (from *ibaPDA* to target system)
Disconnects (in)	Number of currently interrupted data connections for reception (in)
Disconnects (out)	Number of currently interrupted data connections for sending (out)
Error counter	Communication error counter
Exchange ID	ID of the data exchange
Incomplete errors	Number of incomplete messages

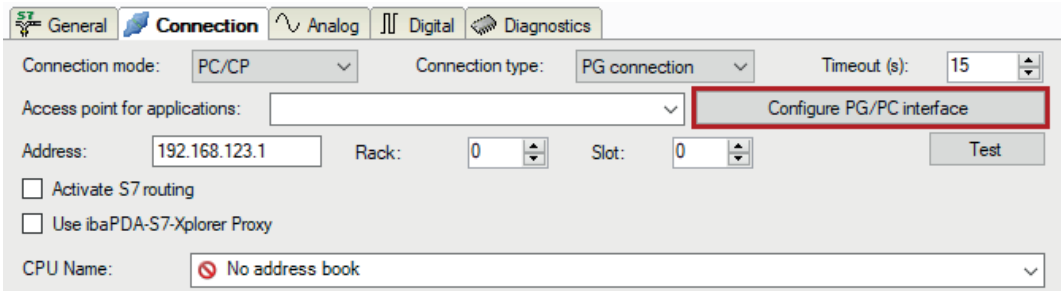
Signal name	Description
Incorrect message type	Number of received messages with wrong message type
Input data length	Length of data messages with input signals in bytes (*ibaPDA* receives)
Invalid data points	Number of received data points with missing configuration
Invalid packet	Invalid data packet detected
IP address (part 1-4)	4 octets of the IP address of the target system
Keepalive counter	Number of Keepalive messages received by the OPC UA Server
Lost images	Number of lost images (in) that were not received even after a retransmission
Lost Profiles	Number of incomplete/incorrect profiles
Message counter	Number of messages received
Messages per cycle	Number of messages in the cycle of the update time
Messages received since configuration	Number of received data telegrams (in) since start of acquisition
Messages received since connection start	Number of received data telegrams (in) since the start of the last connection setup. Reset with each connection loss.
Messages sent since configuration	Number of sent data telegrams (out) since start of acquisition
Messages sent since connection start	Number of sent data telegrams (out) since the start of the last connection setup. Reset with each connection loss.
Multicast join error	Number of multicast login errors
Number of request commands	Counter for request messages from *ibaPDA* to the PLC/CPU
Output data length	Length of the data messages with output signals in bytes (*ibaPDA* sends)
Packet size (actual)	Size of the currently received message
Packet size (max)	Size of the largest received message
Ping time (actual)	Response time for a ping telegram
Port	Port number for communication
Producer ID (part 1-4)	Producer ID as 4-byte unsigned integer
Profile Count	Number of completely recorded profiles
Read counter	Number of read accesses/data requests
Receive counter	Number of messages received
Response time (actual/average/max/min)	Response time is the time between measured value request from *ibaPDA* and response from the PLC or reception of the data. Actual: current value Average/max/min: static values of the update time since the last start of the acquisition or reset of the counters.
Retransmission requests	Number of data messages requested again if lost or delayed

Signal name	Description
Rows (last)	Number of resulting rows by the last SQL query (within the configured range of result rows)
Rows (maximum)	Maximum number of resulting rows by any SQL query since the last start of acquisition (possible maximum equals the configured number of result rows)
Send counter	Number of send messages
Sequence errors	Number of sequence errors
Source IP address (part 1-4) O->T	4 octets of the IP address of the target system Output data (from target system to *ibaPDA*)
Source IP address (part 1-4) T->O	4 octets of the IP address of the target system Input data (from *ibaPDA* to target system)
Statements processed	Number of executed statements since last start of acquisition
Synchronization	Device is synchronized for isochronous acquisition
Time between data (actual/ max/min)	Time between two correctly received messages Actual: between the last two messages Max/min: statistical values since start of acquisition or reset of counters
Time offset (actual)	Measured time difference of synchronicity between *ibaPDA* and the ibaNet-E device
Topics Defined	Number of defined topics
Topics Updated	Number of updated topics
Unknown sensor	Number of unknown sensors
Update time (actual/average/ configured/max/min)	Specifies the update time in which the data is to be retrieved from the PLC, the CPU or from the server (configured). Default is equal to the parameter "Timebase". During the measurement the real actual update time (actual) can be higher than the set value, if the PLC needs more time to transfer the data. How fast the data is really updated, you can check in the connection table. The minimum achievable update time is influenced by the number of signals. The more signals are acquired, the greater the update time becomes. Average/max/min: static values of the update time since the last start of the acquisition or reset of the counters.
Write counter	Number of successful write accesses
Write lost counter	Number of failed write accesses

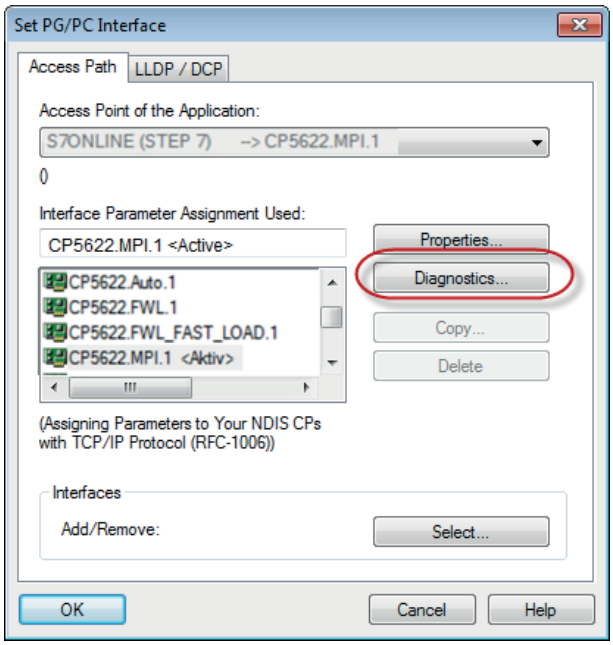
4.7 Connection diagnostics with PG/PC interface

Use the diagnostic function of the PG/PC interface to the functionality and connection configuration.

1. Open the dialog for configuring the PG/PC interface with the <Configure PG/PC interface> button.

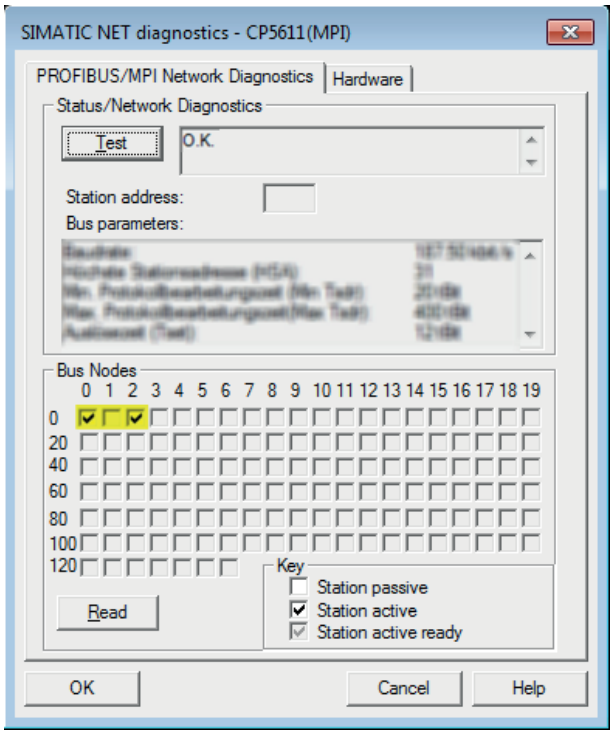


2. Open the diagnostics dialog with the <Diagnostics> button.



3. Start the network diagnostics with the <Test> button.
   Check the availability of the bus devices with the <Read> button.

→ The following figure shows an example of a diagnostics of a SIMATIC Net CP5622 (PROFIBUS).
One active station is on the address 0 and 2 each.



4.8 Error messages

The following tables contain error messages and their possible cause.

The error messages are output when validating the configuration at the beginning of the measurement in the I/O Manager of *ibaPDA*.

Error code	Description	Possible cause
0xFFF00000	Function not implemented	
0xFFF00001	Parameter error when calling a function	False format of IP address
0xFFF00002	Invalid device number	
0xFFF00003	Invalid PLC number	
0xFFF00004	Invalid parameter type	
0xFFF00005	Not enough memory	
0xFFF00006	Device already opened	
0xFFF00007	Device not opened	
0xFFF00008	Order has already been removed	
0xFFF00009	Invalid order number	
0xFFF0000A	Function is not supported	
0xFFF0000B	Adapter not initialized	

Error code	Description	Possible cause
0xFFF0000C	No free connections available	
0xFFF0000D	Device is not configured and not supported	
0xFFF0000E	Hardware was not found	No connection Adapter not connected Interface not available
0xFFF0000F	Error when reading the parameters	
0xFFF00010	Error when writing the parameters	
0xFFF00011	The demo time has elapsed	
0xFFF00012	False character in the text. Conversion not possible.	
0xFFF00013	Buffer is not long enough.	
0xFFF00014	File was not found	
0xFFF00015	Error when calling the configuration program	
0xFFF00016	Error when dynamically loading the DLL	
0xFFF00017	The file could not be generated	
0xFFF50000	No free order buffer available	
0xFFF50001	Invalid packet	
0xFFF50002	No connection to the PLC	All connections are already assigned
0xFFF50003	The connection has been closed	
0xFFF50004	Timeout	Wrong MPI address False rack number False slot number
0xFFF50005	Invalid context	
0xFFF50006	PLC-memory error	
0xFFF50007	Invalid operating state	
0xFFF50008	Invalid address on the PLC	
0xFFF50009	Invalid mode	
0xFFF5000A	No data available, e.g. DB is missing	
0xFFF5000B	Execution level for OB not available	
0xFFF5000C	Empty block list	
0xFFF5000D	Error in the block size	
0xFFF5000E	Invalid block number	
0xFFF5000F	Protection level of the function not sufficient	
0xFFF50010	Unknown SSL ID (e.g. access to CP instead of PLC)	False slot number

Error code	Description	Possible cause
0xFFF50011	Unknown SZL index (e.g. access to CP instead of PLC)	
0xFFF50012	Information cannot be obtained at the moment	
0xFFF50013	Unknown error message from PLC (please report!)	
0xFFF50014	Hardware failure, e.g. nonexistent peripherals	
0xFFF50015	Object access not allowed	
0xFFF50016	Context is not supported	
0xFFF50017	Type (data type) not supported	
0xFFF50018	Internal error, please report	
0xFFF50019	Error when restarting the PLC	
0xFFF5001A	Error at automatic start-up of the PLC	
0xFFF5001B	End of connection received	
0xFFF5001C	PLC not found	False rack number False slot number
0xFFF5001D	Too many data for this ACCON-AGLink version	
0xFFF5001E	The PLC does not support this function	
0xFFF5001F	The password entered is wrong	
0xFFF50020	The connection has already been legitimized	
0xFFF50021	The connection legitimation has already been canceled	
0xFFF50022	The password is not required, since no password has been configured	
0xFFF50023	At least one variable address is invalid	
0xFFF50024	Specified order does not exist	
0xFFF50025	Impermissible order state	
0xFFF50026	Impermissible cycle time (timebase or multiple thereof impermissible)	
0xFFF50027	No further cyclical reading order can be set up	
0xFFF50028	Function for this order not possible (false state)	
0xFFF50029	Termination of the function due to overload (reading cycle takes longer than the cycle time set)	
0xFFF5002A	No data for this part of the reading cycle has been supplied	

Error code	Description	Possible cause
0xFFF5002B	False time format	
0xFFF5002C	Unknown PI name	
0xFFF5002D	Transfer has been interrupted by NCK	
0xFFF5002E	The response telegram is too big for the PDU size	
0xFFF5002F	No H-CPU found	
0xFFF50030	The data have not changed	
0xFFF30000	Error when reading the device information	
0xFFF30001	Error when reading the bus parameters	
0xFFF30002	Error when writing the bus parameters	
0xFFF30003	No resources available on the device	
0xFFF30004	Invalid device	
0xFFF30005	Adapter was not found	
0xFFF30006	Required device driver has not been found	
0xFFF20000	The desired adapter address already exists	
0xFFF20001	The HSA is invalid (shorter than biggest active node)	
0xFFF20002	The adapter is not in the logical ring	
0xFFF20004	Received packet has wrong contents	
0xFFF20006	Unknown error code of the adapter	
0xFFF20007	Unknown error code of the device driver	
0xFFF20008	The communication adapter has been removed	
0xFFF20009	The modem has been removed	
0xFFF2000A	No directly connected PLC found	
0xFFF20313	Invalid speed at the MPI bus	
0xFFF20314	The address is longer than HSA	
0xFFF20315	The desired adapter address already exists	
0xFFF2031A	No other active bus device found	
0xFFF2031C	The bus is disturbed	
0xFFF2031D	The bus is disturbed	
0xFFF2031E	Automatic bus profile identification does not work, bus parameter telegram is missing	
0xFFF20337	Legitimization for accessing the TS adapter is missing	
0xFFF10000	The COM port is already in use	
0xFFF10001	The modem was not found	
0xFFF10002	The modem did not hang up	
0xFFF10003	The modem did not respond	

Error code	Description	Possible cause
0xFFF10004	Modem error when initializing the basis	
0xFFF10005	Error in Initstring 1	
0xFFF10006	Error in Initstring 2	
0xFFF10007	Error in Initstring 3	
0xFFF10008	Error in Initstring 4	
0xFFF10009	Error in the dial mode specification sequence	
0xFFF1000A	Error in the dial tone specification sequence	
0xFFF1000B	Error in the automatic call acceptance sequence	
0xFFF1000C	The modem has been removed	
0xFFF1000D	No connection could be established	
0xFFF1000E	Login has been rejected. The specified user name is not known.	
0xFFF1000F	Login has been rejected. The password entered is not correct.	
0xFFF10010	Login has been rejected. A callback number has already been configured in the TS adapter.	
0xFFF10011	Error when selecting	
0xFFF80000	Function not allowed	
0xFFF80001	Invalid project handle	
0xFFF80002	Error when opening a project	
0xFFF80003	Error when creating the program instance of the project	
0xFFF80004	Error when closing a project	
0xFFF80005	No or no further program found	
0xFFF80006	Specified program has not been found	
0xFFF80007	One of the parameters is outside the valid range	
0xFFF80008	Scope of functions or runtime has elapsed (demo version)	
0xFFF8000A	No or invalid data indicated	
0xFFF8000B	The selected program does not contain a symbol table	
0xFFF8000C	Symbol table already opened	
0xFFF8000D	No or no further symbol entry found	
0xFFF8000E	Error when reading the symbol dataset	
0xFFF8000F	Symbol not found	
0xFFF80010	Absolute operand not found	

Error code	Description	Possible cause
0xFFF80011	Invalid symbol	
0xFFF80012	Invalid absolute operand	
0xFFF80013	Invalid filter string	
0xFFF80014	The indicated DB is not available	
0xFFF80015	Error when reading the block data of the DB	
0xFFF80016	Error when relocating the block data of the DB	
0xFFF80017	A DB for searching components is already opened	
0xFFF80018	No DB for searching components is opened yet	
0xFFF80019	No or no further component available	
0xFFF8001A	DB component not found	
0xFFF8001B	Invalid DB component	
0xFFF8001C	DB component not suitable	
0xFFF8001E	Invalid constant specification	
0xFFF8001F	Invalid constant size or formatting	
0xFFF80020	Initializing the object types failed	
0xFFF80021	False version of the message configuration	
0xFFF80022	Error when opening the message configuration	
0xFFF80023	Error when exiting the message configuration	
0xFFF80024	No or no further message entries found	
0xFFF80025	No or no further language found	
0xFFF80026	Error when accessing the database	
0xFFF80027	Invalid signal number	
0xFFF80028	Invalid specification of associated value	
0xFFF80029	Invalid number of the associated value	
0xFFF8002A	Invalid element type	
0xFFF8002B	Invalid length of associated value data	
0xFFF8002C	Format not specified	
0xFFF8002D	Invalid format specification	
0xFFF8002E	No or no further text library available	
0xFFF8002F	No or no further text library entry available	
0xFFFA0195	Wrong read type	

5 Appendix

5.1 Comparison of the cycle times on different access points

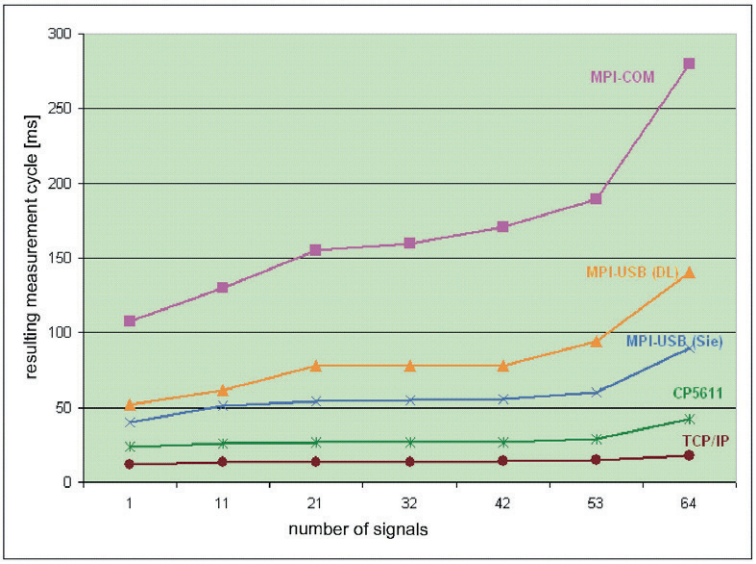
Using an experimental arrangement, the resulting cycle times of the measurement depending on the different access points of the S7 CPU were measured and evaluated.

Configuration of the PLC:

- S7-400 CPU 416 2DP
- Cycle OB1: 6 ms to 10 ms
- *ibaPDA* measurement time base: 1 ms

Number of signals	MPI adapter (COM) 38.4 kBd	MPI adapter (USB, DeltaL) 38.4 kBd	MPI adapter (USB, Siemens)	CP5611 PROFIBUS, 12 Mbit/s	TCP/IP	L2B *)
1	108	52	40	24	12	1
11	130	62	51	26	13	1
21	155	78	54	27	13	1
32	160	78	55	27	13	1
42	171	78	56	27	14	1
53	189	94	60	29	15	1
64	280	140	90	42	18	1

*) 1 ms is the *ibaPDA* time base which is supported by iba cards such as ibaCom-L2B-8-8, for example. But the actual measurement cycle, in fact, is depending on the PROFIBUS cycle time (with 2 slaves: 0.98 ms; with 8 slaves: 1.6 ms, provided 12 Mbit/s) and the program cycle (OB1 or OB35)



5.2 Comparison of the cycle times for different S7-CPUs

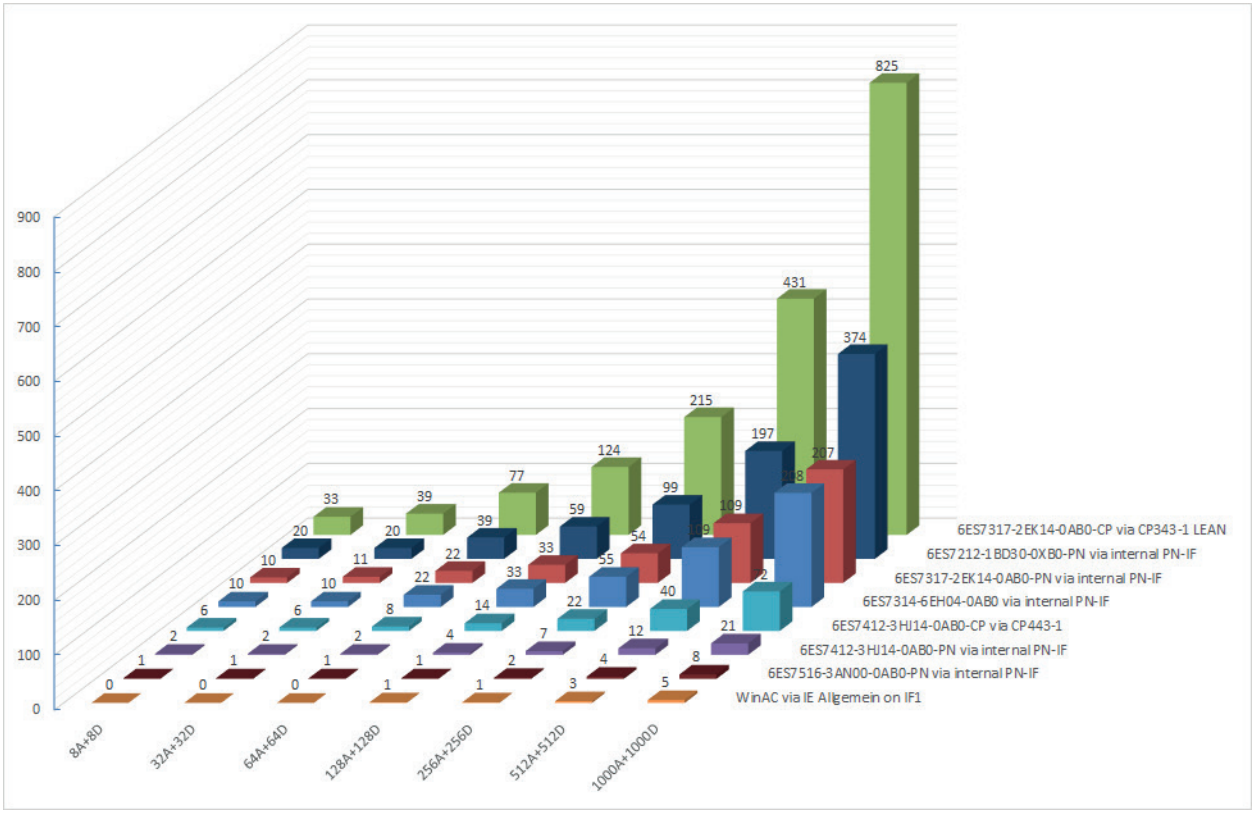
Using an experimental arrangement, the updating times of the signals on different S7-CPUs and access via TCP/IP were measured.

Configuration of the controllers:

- Cycle OB1: 10 ms
- *ibaPDA* base measurement time base: 1 ms
- Connection mode TCP

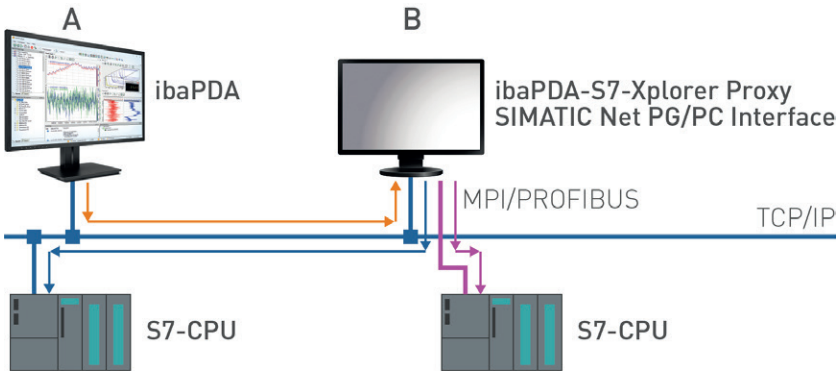
There are many different factors that have influence on the measured updating times. A direct transfer to a specific plant configuration is only feasible to a certain extent.

Quantity structure	**8A +8D**	**32A +32D**	**64A +64D**	**128A +128D**	**256A +256D**	**512A +512D**	**1000A +1000D**
CPU	**Updating time [ms]**						
CPU 314C 6ES7314 6EH04-0AB0 via internal PN-IF	10	10	22	33	55	109	208
CPU 317 6ES7317 2EK14-0AB0 V3.2.10 via internal PN-IF	10	11	22	33	54	109	207
CPU 317 6ES7317 2EK14-0AB0 V3.2.10 via CP343-1 LEAN	33	39	77	124	215	431	825
CPU 412-2PN 6ES7412 3HJ14-0AB0 V6.0.3 via internal PN-IF	2	2	2	4	7	12	21
CPU 412-2PN 6ES7412 3HJ14-0AB0 V6.0.3 via CP443-1	6	6	8	14	22	40	72
WinAC via internal PN-IF	0	0	0	1	1	3	5
CPU1212C 6ES7 212-1BD30-0XB0 V2.0 via internal PN-IF	20	20	39	59	99	197	374
CPU1516 6ES7 516-3AN00-0AB0 V1.0 via internal PN-IF	1	1	1	1	2	4	8


900
800
700
600
500
400
300
200
100
0
8A+8D
32A+32D
64A+64D
128A+128D
256A+256D
512A+512D
1000A+1000D
6ES7317-2EK14-0AB0-CP via CP343-1 LEAN
6ES7212-1BD30-0XB0-PN via internal PN-IF
6ES7317-2EK14-0AB0-PN via internal PN-IF
6ES7314-6EH04-0AB0 via internal PN-IF
6ES7412-3HJ14-0AB0-CP via CP443-1
6ES7412-3HJ14-0AB0-PN via internal PN-IF
6ES7516-3AN00-0AB0-PN via internal PN-IF
WinAC via IE Allgemein on IF1

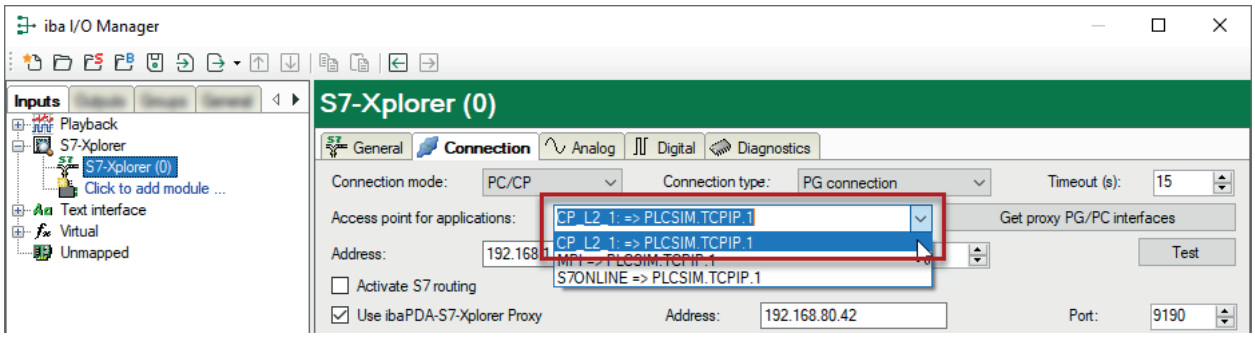
5.3 ibaPDA-S7-Xplorer Proxy for PC/CP interface

You can also use the *ibaPDA-S7-Xplorer Proxy* when STEP 7 or SIMATIC Net is not installed on the *ibaPDA* computer (computer A), but the CPU still must be accessed via an *S7-Xplorer* module with connection mode PC/CP.



For this purpose, install ant start he *ibaPDA-S7-Xplorer Proxy* on the computer with the STEP 7/ SIMATIC NET installation (computer B).

Unlike the configuration with S7-PLCSIM, you must set a real interface (e.g. the network interface card) as access point for applications. Install the access point on computer B.



S7-CPU can be accessed via all installed access points.

5.4 Setting PG/PC interface/defining new access point

ibaPDA-Interface-S7-Xplorer cannot establish a connection to a S7-CPU if the parametrization "AUTO" for an access point (MPI-adapter or CP) has been set in the SIMATIC Manager.

There are 2 possible remedies:

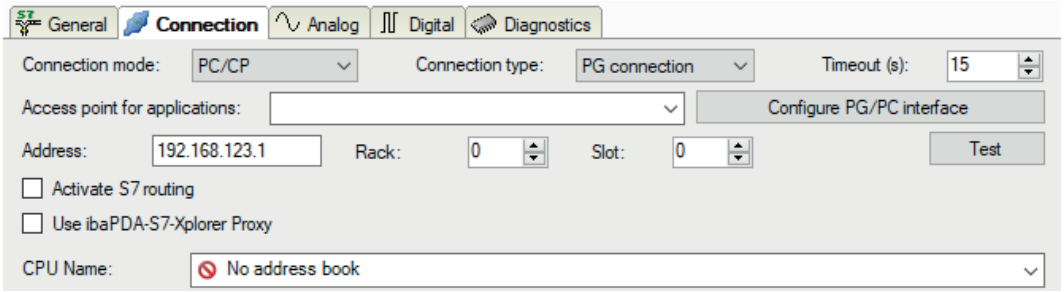
Changing the interface with remaining access point name
Change interface in the SIMATIC Manager e.g. from "CP5622 (AUTO)" to "CP5622 (MPI)" or "CP5622 (PROFIBUS)".

Disadvantage of this method: If the setting of the access point is changed again in the SIMATIC Manager, the measurement does no longer work because *ibaPDA* no longer has any access.

Adding a special access point for ibaPDA
To avoid conflicts with the setting of SIMATIC Manager and *ibaPDA* when both programs run on the same computer, you can define a new access point.

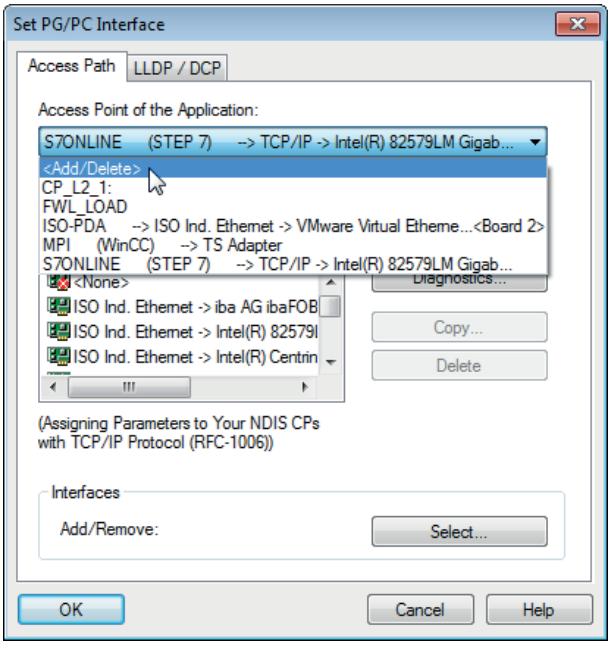
There is the <Configure PG/PC interface> button in the dialog window of the module. With this button, you can open the dialog for configuring the PG/PC interface.

The setting for the SIMATIC Manager is also changed.

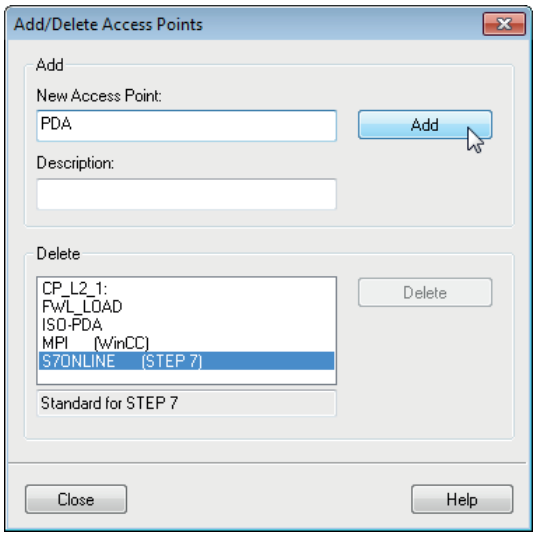


Procedure

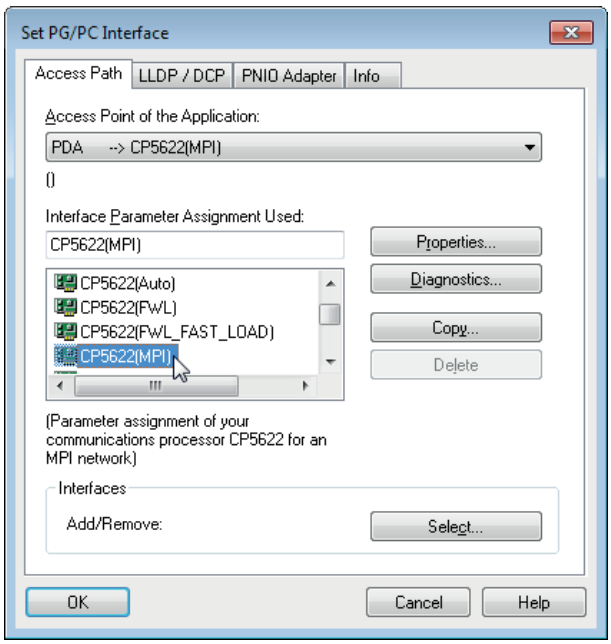
1. Open the dialog for configuring the PG/PC interface with the <Configure PG/PC interface> button.
2. Under *Access Point of the Application* select the row *<Add/Delete>*.



3. Define a new access point: Enter a name, e.g. "PDA", and optionally a description for a better understanding.
   Confirm your entries with <Add> and <Close>.

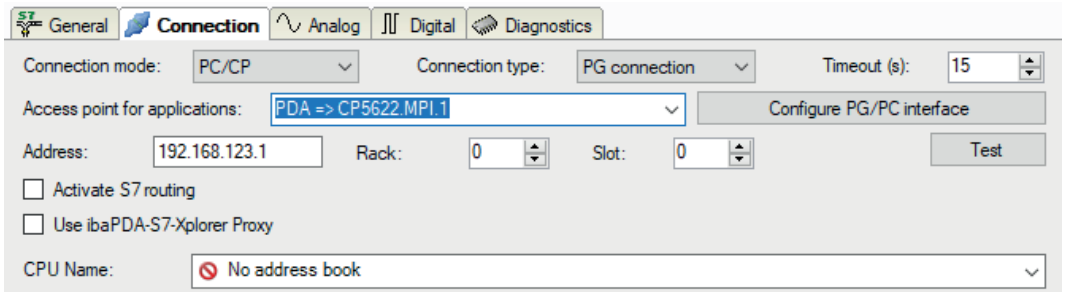


4. Add an interface to the access point, e.g. "CP5622 (MPI)".



5. Exit the configuration with <OK>.

→ *ibaPDA* subsequently displays the newly defined access (e.g. "PDA --> CP5622.MPI.1") in the connection dialog under *Access points for applications*.



Notes on the different access points

Depending on which access points have been configured in the Engineering computer, there are different access points available for selection in the *ibaPDA* system.

Basically, there are 3 types of access points:

- TCP/IP
- ISO
- Bus system PROFIBUS or MPI

TCP/IP

If you select an access point using TCP/IP, you need to enter the IP address, rack number and slot number of the CP in the module configuration dialog. If you do not know the rack number and/or slot number, enter "0" for slot and click on the <Test> button.

ISO

If you select an access point using an ISO interface, you need to enter the MAC address, rack number and slot number. If you do not know the rack number and/or slot number, enter "0" for slot and click on the <Test> button.

Bus system PROFIBUS or MPI

If you select an access point using a bus interface, like e.g. PROFIBUS or MPI, you need to enter the bus address, the rack number and slot number. You can also use the <Test> button and then click on one of the CPU links found to test the connection.

5.5 S7 routing

S7 routing is defined as the possibility to use S7 controls as router to access secondary target systems, i.e. controls or drives, which are in different subnets. This also includes changing the bus system (Ethernet, PROFIBUS, MPI).

Reference

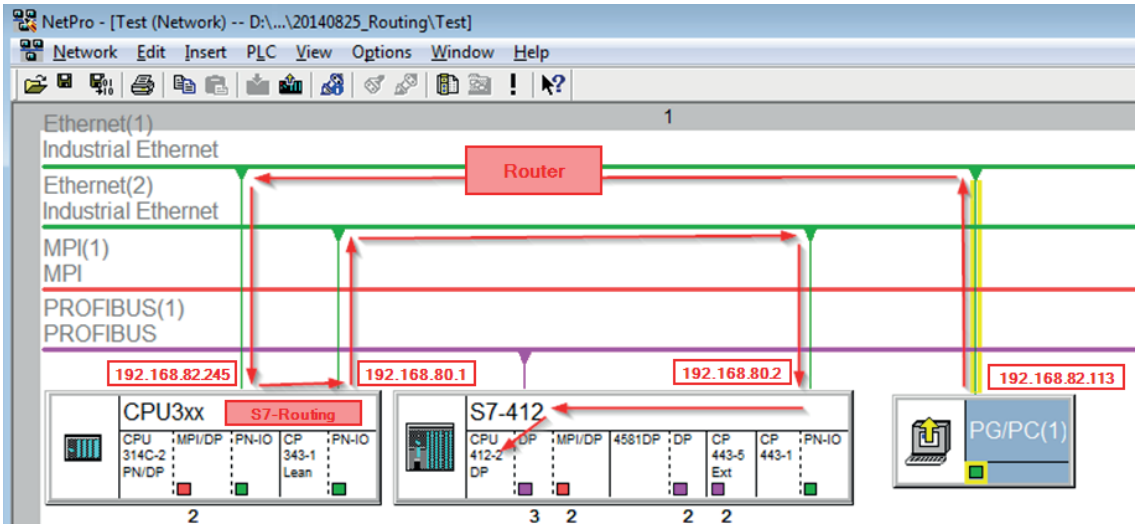
For more information about the S7 Routing, refer to:

- Which modules support the "S7 Routing" function in S7 subnets?
  https://support.automation.siemens.com/ww/view/en/584459
- Which requirements must be fulfilled and what do I have to observe if I want to execute routing?
  https://support.industry.siemens.com/cs/ww/en/view/2383206
- How do you enable cross-project S7 Routing in the TIA Portal and in STEP 7 V5.x?
  https://support.industry.siemens.com/cs/ww/en/view/109474569

5.5.1 Routing from Ethernet to Ethernet

Do not mix up the *S7 Routing* function with IP routing.

The example shows how to implement the following way of access via S7 Routing in NetPro.



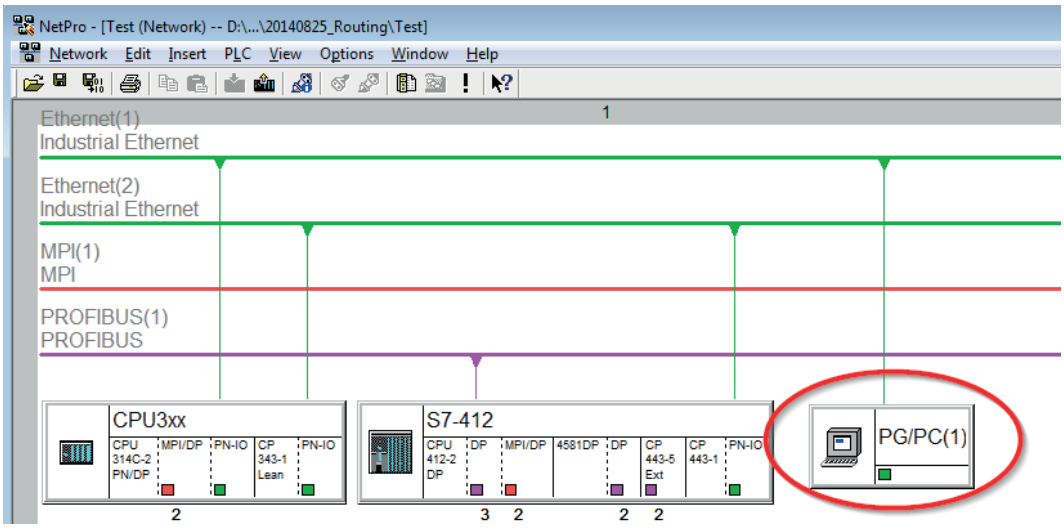
The engineering computer (also with *ibaPDA*) is to access the CPU412 controller. The computer and the controller are not directly connected via a common network/bus. The connection has to run over the CPU314C controller.
"Passing" the communication through this controller is called *S7 Routing*.

In our example, engineering computer and CPU314C are also located in two different (logic) subnets. You need an (IP) router for establishing a communication connection. This is completely independent of the S7 Routing function and should not be mixed up with it.

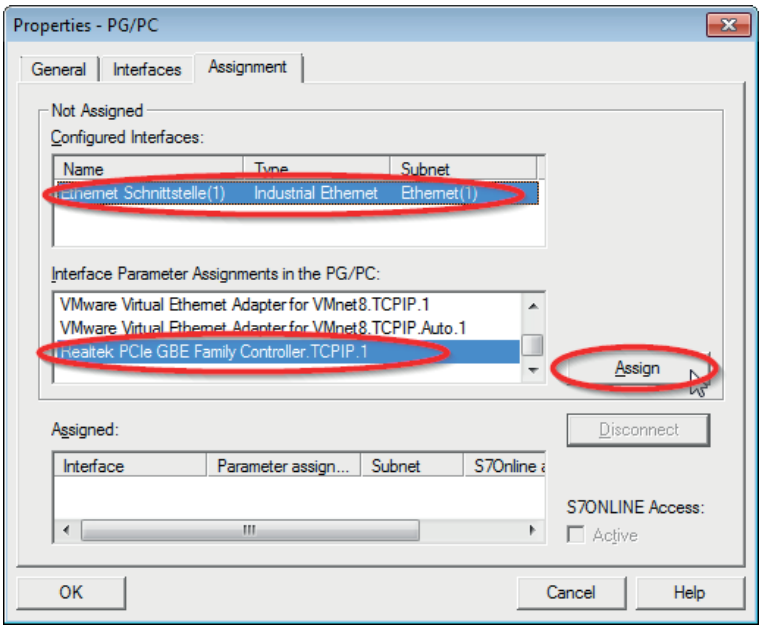
5.5.1.1 Configuration of STEP 7/NetPro

The following configuration steps are required to be able to access the secondary CPU412 control with the SIMATIC STEP 7 programming software. For SINUMERIK, SINAMICS, or SIMOTION, you can apply similar steps. For using *ibaPDA*, these configuration steps are not required.

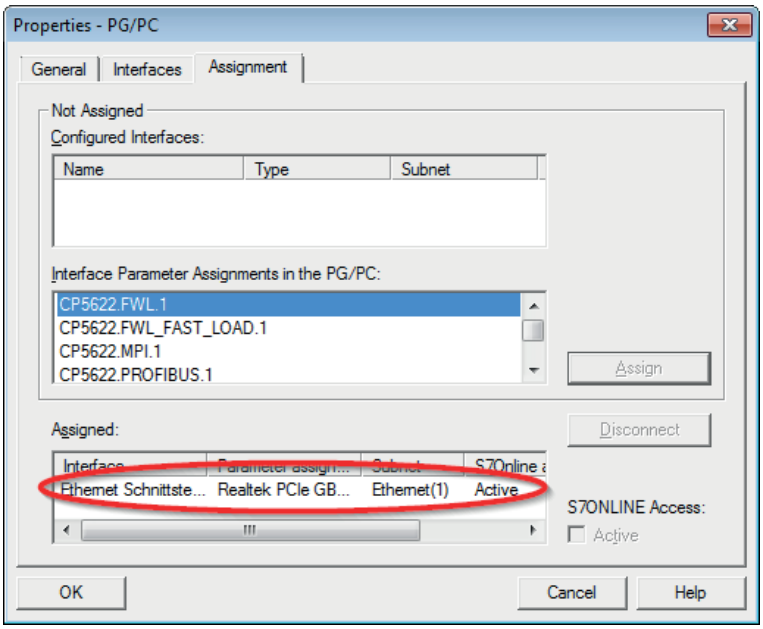
1. Add a PG/PC station and configure it.



2. Assign an interface (network card).

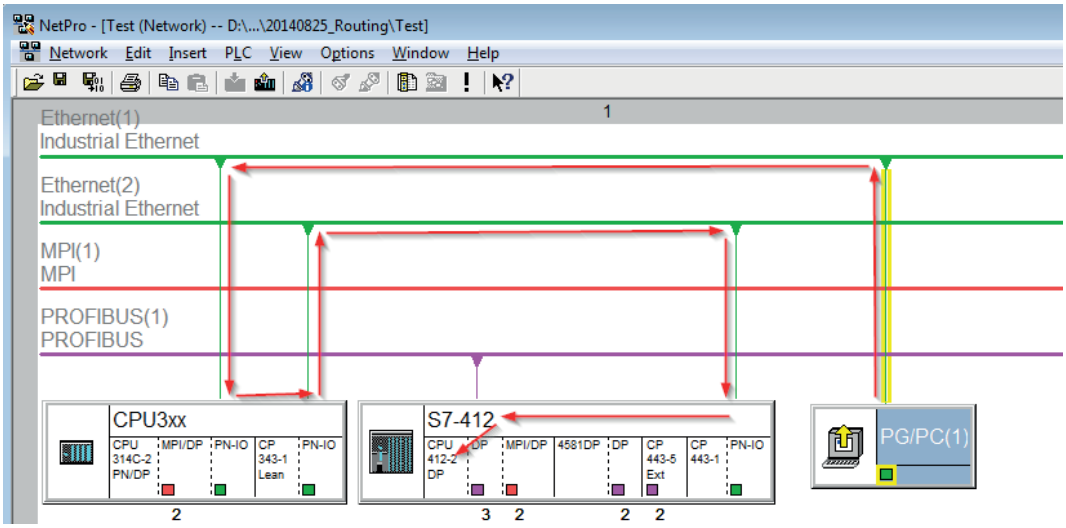


→ Result:



Now, the connection line from PG/PC to the network has to be marked in yellow.

In the following figure, the communication path is shown using arrows (these are not displayed in SIMATIC NetPro).

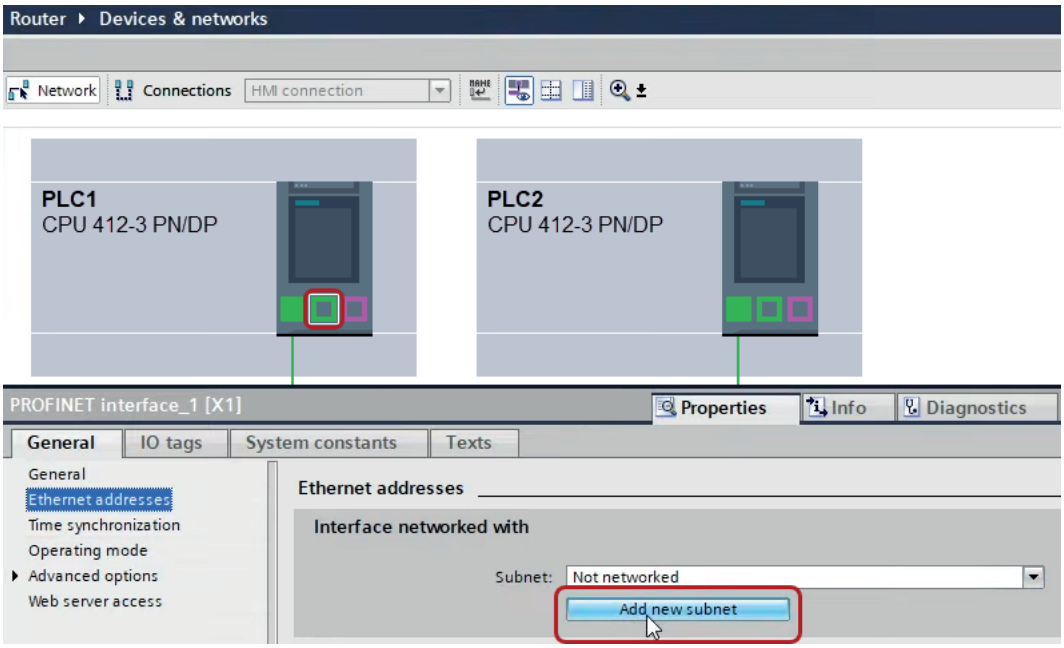


3. Finally, load all hardware configurations and connection data from NetPro.

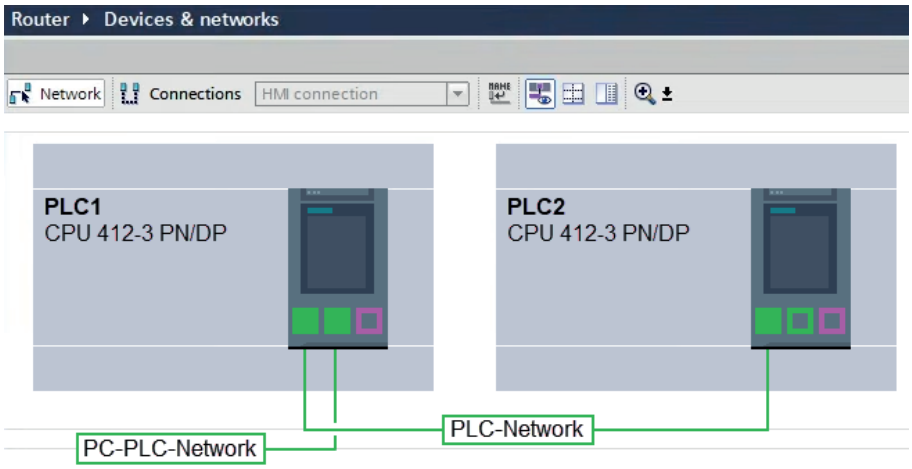
5.5.1.2 Configuration of TIA Portal

The following configuration steps are required exclusively for accessing the subordinate control "PLC2" by the programming software TIA 7. For SINUMERIK, SINAMICS, or SIMOTION, you can apply similar steps. For using *ibaPDA*, these steps are not required.

1. Connect both controllers in TIA Portal via the Ethernet ports.
2. Establish a connection with your computer and the first controller "PLC1" by adding a subnet.

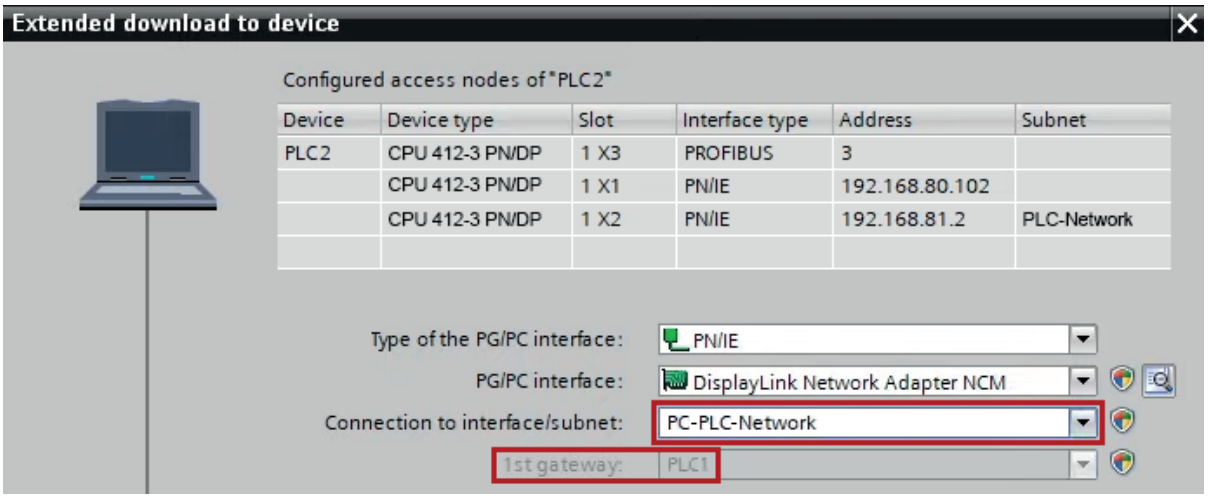


→ The connections between the controllers and to the computer are shown in TIA Portal.



3. Download the programming of the controller "PLC1" and then the programming of the controller "PLC2".

4. In the controller "PLC2", set the connection from "PLC1" to the computer as the connection to the interface/subnet.

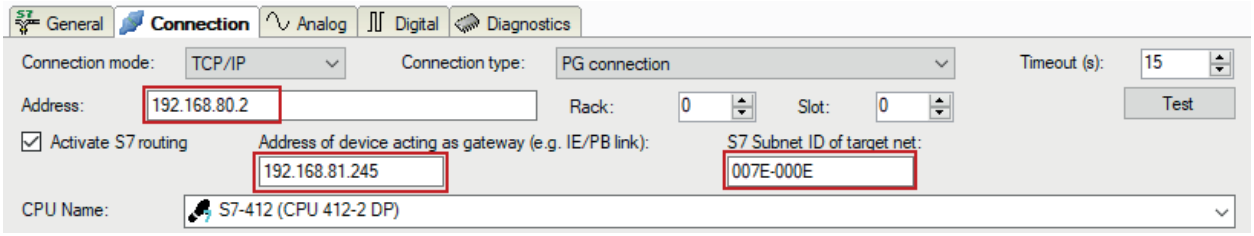


→ "PLC1" appears as the first gateway.

→ "PLC2" is now connected to the computer via S7 routing via "PLC1".

5.5.1.3 Configuration of ibaPDA

Configure the following settings.



Activate S7 routing
Enable this option to use S7 routing.

Address
Enter the address of the target control (here CPU412)

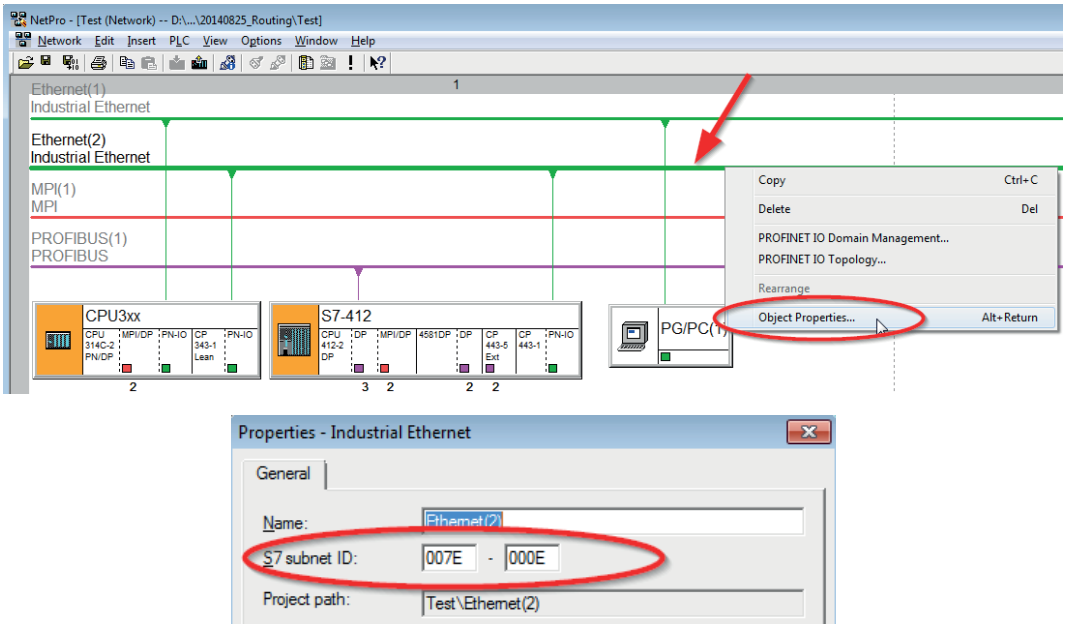
Address of device acting as gateway
Enter the address of the gateway (here CPU314C).

S7 subnet ID of target net
Enter the subnet ID from STEP 7 NetPro or TIA Portal.

Identifying the S7 subnet ID in NetPro

You can identify the S7 subnet ID in NetPro.

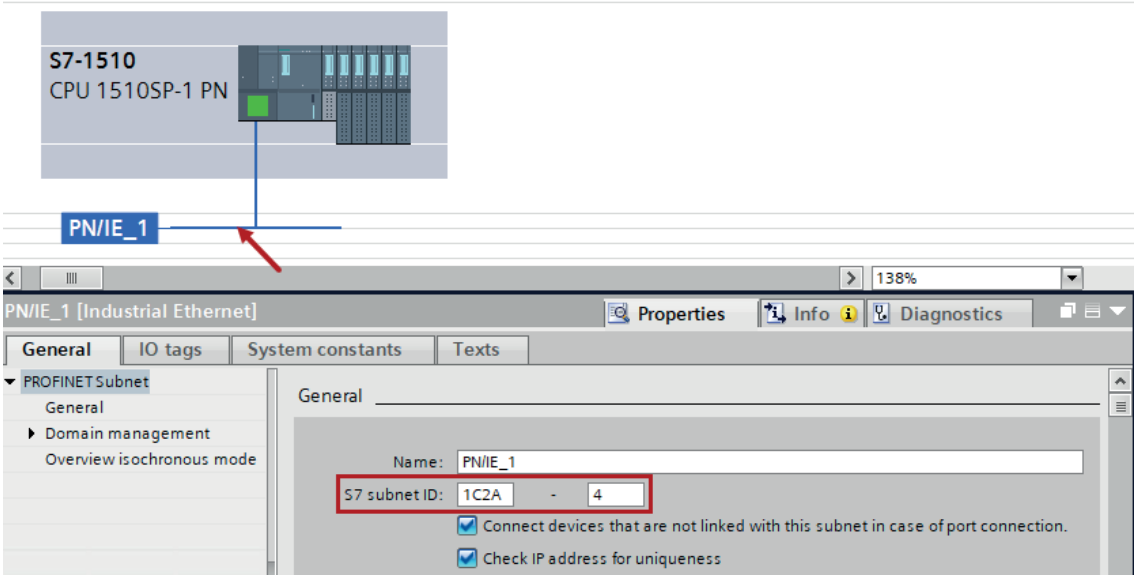
Right-click on the secondary bus system and open the *Object Properties*.



Identifying the S7 subnet ID in TIA Portal

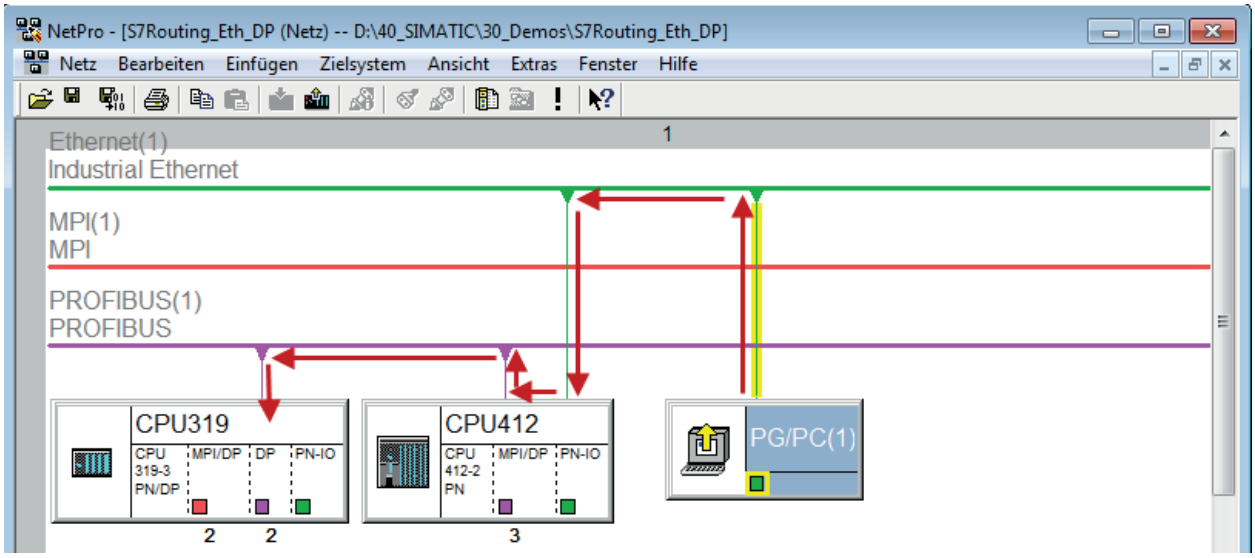
You can identify the S7 subnet ID in TIA Portal.

Click on the bus system and go to *Properties – General – General*.



5.5.2 Routing from Ethernet to PROFIBUS

The example shows how to implement the following way of access via S7 Routing and an example system topology for Ethernet PROFIBUS in NetPro.

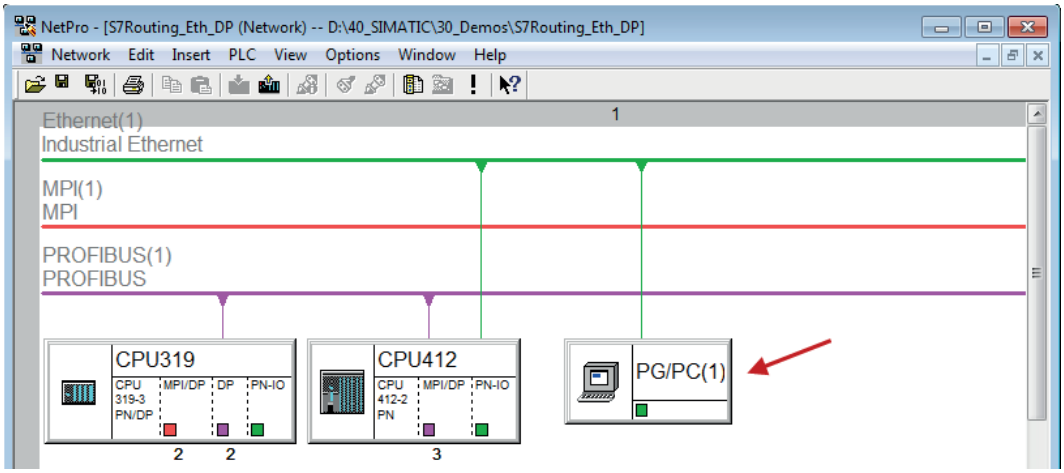


The engineering computer (also with *ibaPDA*) is to access the CPU319 controller. The computer and the controller are not directly connected via a common network/bus. The connection has to run over the CPU412 controller.
"Passing" the communication through this controller is called *S7 Routing*.

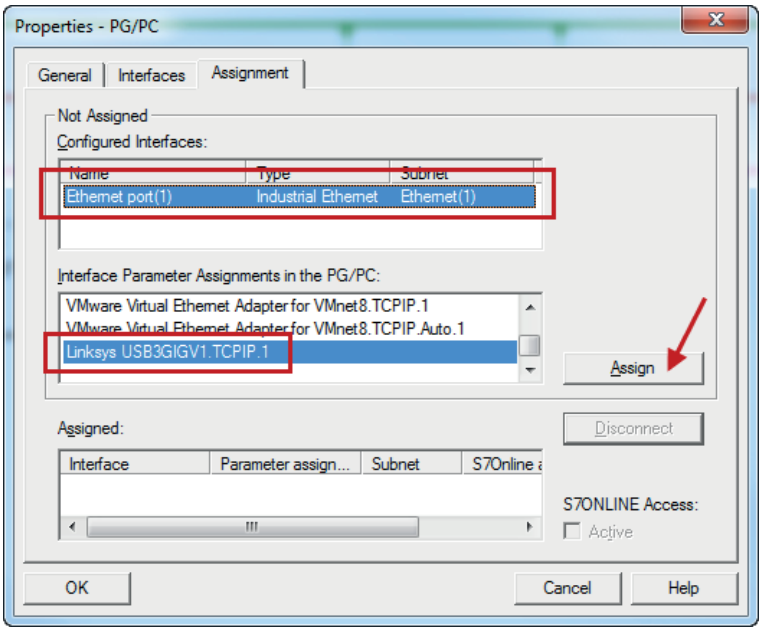
5.5.2.1 Configuration of STEP 7/NetPro

The following configuration steps are exclusively required for accessing the subordinate controller CPU319 via the SIMATIC STEP 7 programming software. For SINUMERIK, SINAMICS, or SIMOTION, you can apply similar steps. For using *ibaPDA*, these configuration steps are not required.

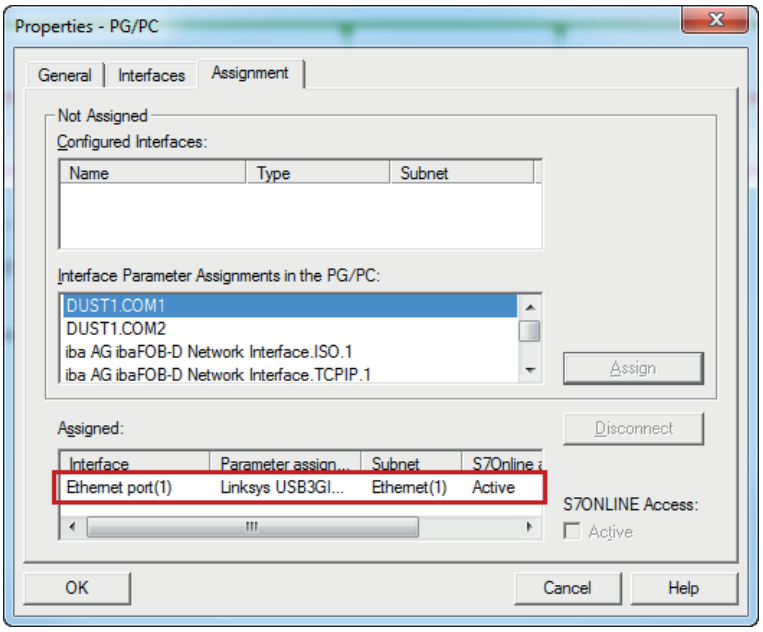
1. Add a PG/PC station and configure it.



2. Assign an interface (network card).

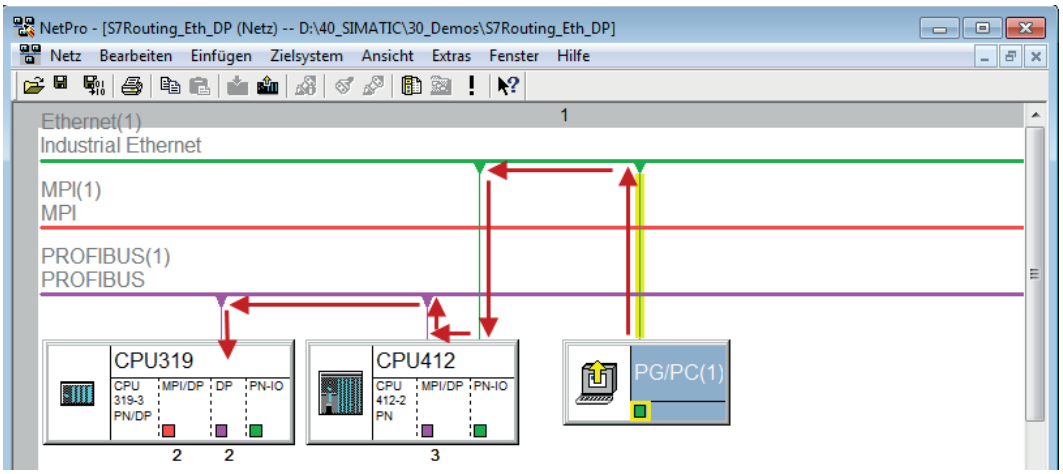


→ Result:



Now, the connection line from PG/PC to the network has to be marked in yellow.

In the following figure, the communication path is shown using arrows (these are not displayed in SIMATIC NetPro).



3. Finally, load all hardware configurations and connection data from NetPro.

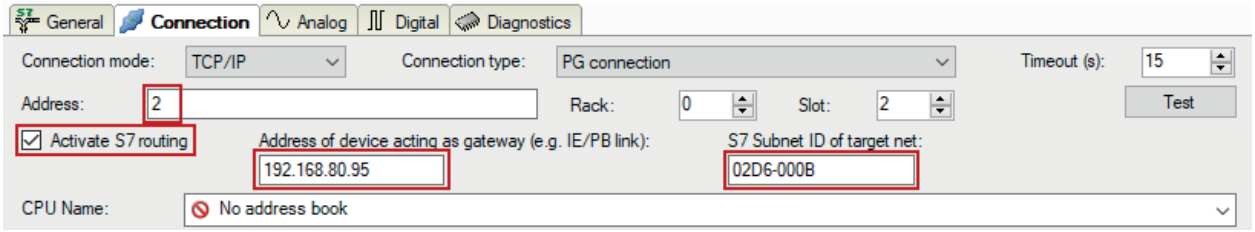
5.5.2.2 Configuration of TIA Portal

The configuration steps are required exclusively for accessing the subordinate control "PLC2" by the programming software TIA Portal. For SINUMERIK, SINAMICS, or SIMOTION, you can apply similar steps.

For the configuration of PROFIBUS proceed as described for Ethernet, see ↗ *Configuration of TIA Portal*, page 96.

5.5.2.3 Configuration of ibaPDA

Configure the following settings.



Activate S7 routing
Enable this option to use S7 routing.

Address
Enter the address of the target control (here CPU319).

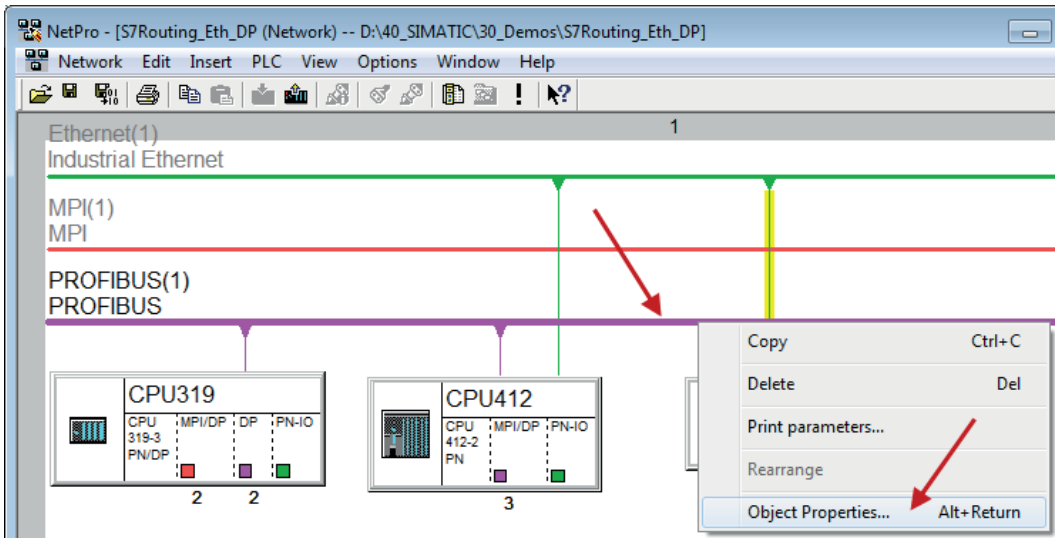
Address of device acting as gateway
Enter the address of the gateway (here CPU412).

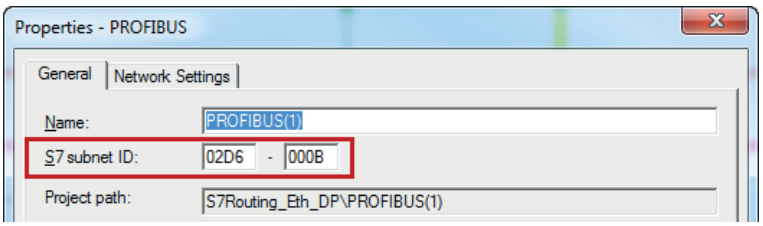
S7 subnet ID of target net
Enter the subnet ID from STEP 7 NetPro.

Identifying the S7 subnet ID in NetPro

You can identify the S7 subnet ID in NetPro.

Right-click on the secondary bus system and open the *Object Properties*.

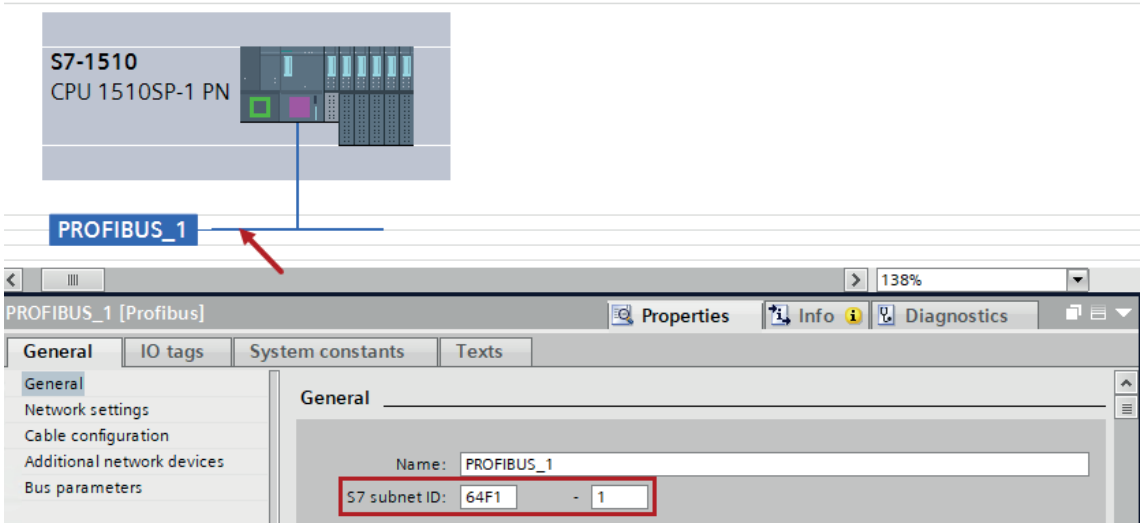




Identifying the S7 subnet ID in TIA Portal

You can identify the S7 subnet ID in TIA Portal.

Click on the bus system and go to *Properties – General – General*.



5.6 Usage of MPI/DP-TCP-adapters

You can connect Siemens S7 CPUs that do not have an Ethernet interface via MPI/DP-TCP-adapters to their MPI interface.

Different manufacturers offer adapters that convert the S7 communication from TCP/IP to MPI/PROFIBUS DP.

Basically, you can use these adapters together with *ibaPDA*. Therefore, you must configure the connection on the *ibaPDA*-side as a TCP-connection.

The access on the adapter is possible over its IP-address, which can be set by means of the associated software provided by the manufacturer.

In the connection configuration enter "0" for frame and the MPI address or DP address of the S7 CPU as slot.

For more information, refer to the corresponding device documentation.

6 Support and contact

Support

Phone: +49 911 97282-14

Email: support@iba-ag.com

Note

If you need support for software products, please state the number of the license container. For hardware products, please have the serial number of the device ready.

Contact

Headquarters

iba AG
Koenigswarterstrasse 44
90762 Fuerth
Germany

Phone: +49 911 97282-0

Email: iba@iba-ag.com

Mailing address

iba AG
Postbox 1828
D-90708 Fuerth, Germany

Delivery address

iba AG
Gebhardtstrasse 10
90762 Fuerth, Germany

Regional and Worldwide

For contact data of your regional iba office or representative
please refer to our web site:

www.iba-ag.com