



ibaPDA-Interface-OMRON-Xplorer

PLC-Xplorer data interface to OMRON systems

Manual
Issue 1.1

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
Fax +49 911 97282-33
Support +49 911 97282-14
Engineering +49 911 97282-13
E-mail iba@iba-ag.com
Web www.iba-ag.com

Unless explicitly stated to the contrary, it is not permitted to pass on or copy this document, nor to make use of its contents or disclose its contents. Infringements are liable for compensation.

© iba AG 2023, All rights reserved.

The content of this publication has been checked for compliance with the described hardware and software. Nevertheless, discrepancies cannot be ruled out, and we do not provide guarantee for complete conformity. However, the information furnished in this publication is updated regularly. Required corrections are contained in the following regulations or can be downloaded on the Internet.

The current version is available for download on our web site www.iba-ag.com.

Version	Date	Revision	Author	Version SW
1.1	06-2023	Note for UDP connection	mm	8.1.0

Windows® is a brand and registered trademark of Microsoft Corporation. Other product and company names mentioned in this manual can be labels or registered trademarks of the corresponding owners.

Contents

1	About this Manual	4
1.1	Target group and previous knowledge	4
1.2	Notations	4
1.3	Used symbols.....	5
2	System requirements OMRON-Xplorer	6
3	PLC-Xplorer data interface to OMRON controllers	8
3.1	General information	8
3.2	System topology	8
3.3	OMRON configuration and engineering	8
3.4	Configuration and engineering ibaPDA.....	11
3.4.1	Interface settings	11
3.4.2	Adding a module.....	12
3.4.3	General module settings.....	12
3.4.4	Connection settings	14
3.4.5	Signal configuration	16
3.4.6	Module diagnostics.....	17
3.4.7	Output module	17
4	Diagnostics.....	19
4.1	License	19
4.2	Visibility of the interface.....	19
4.3	Log files.....	20
4.4	Connection diagnostics with PING.....	21
4.5	Connection table	22
4.6	Diagnostic modules	23
5	Support and contact.....	28

1 About this Manual

This document describes the function and application of the software interface

ibaPDA-Interface-OMRON-Xplorer.

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 manual 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-OMRON-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 <i>Logic diagram</i>
Calling the menu command	<i>Step 1 – Step 2 – Step 3 – Step x</i> Example: Select the menu <i>Logic diagram – Add – New function block</i> .
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 OMRON-Xplorer

The following system requirements are necessary when using the OMRON-Xplorer data interface:

- *ibaPDA* v8.1 or higher
- Base license for *ibaPDA* + license for *ibaPDA-Interface-PLC-Xplorer* or *ibaPDA-Interface-OMRON-Xplorer*
- With more than 16 connections you need additional *one-step-up-Interface-OMRON-Xplorer* licenses for each additional 16 connections.

Note



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

ibaPDA supports OMRON controllers that communicate via the FINS protocol, e.g., CV series and NX series.

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

License information

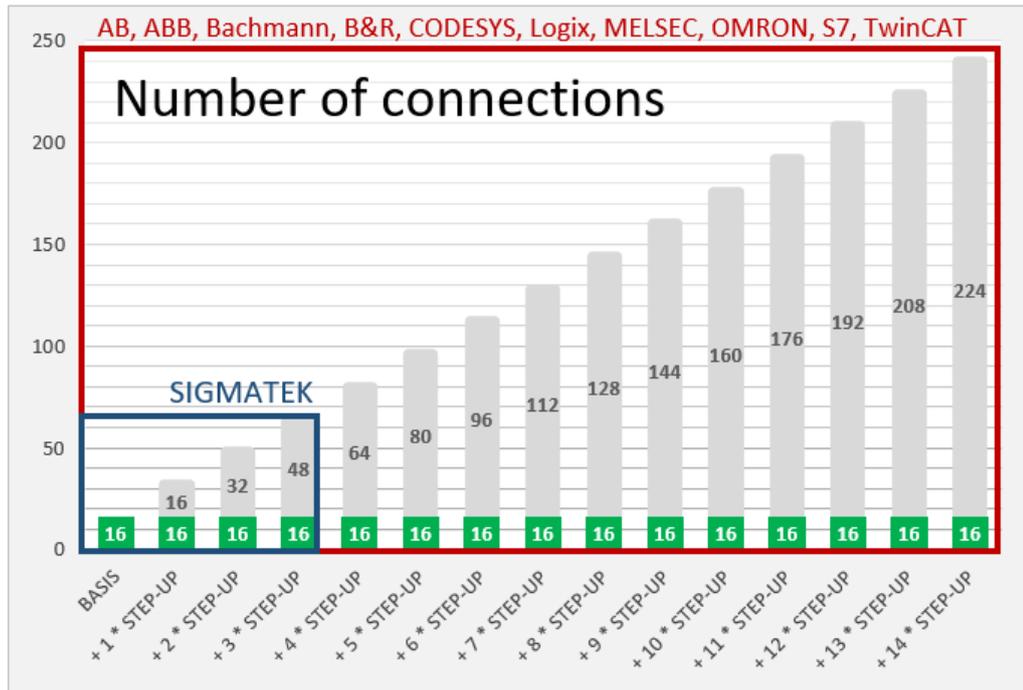
Order no.	Product name	Description
31.001042	ibaPDA-Interface-PLC-Xplorer	Extension license for an <i>ibaPDA</i> system. All additional Xplorer data interfaces are added. (Complete specifications at www.iba-ag.com)
31.000035	ibaPDA-Interface-OMRON-Xplorer	Extension license for an <i>ibaPDA</i> system that adds the data interface: + OMRON-Xplorer (16 connections)
31.100035	one-step-up-Interface-OMRON-Xplorer	Extension license for 16 further OMRON-Xplorer connections (max. 14 extension licenses)

Table 1: Available OMRON-Xplorer 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 OMRON controllers

3.1 General information

The OMRON-Xplorer interface is suitable for measurement data acquisition with *ibaPDA* from an OMRON controller via an Ethernet connection and use of the FINS protocol.

The data is cyclically read by *ibaPDA* instead of being sent by the PLC.

3.2 System topology

The connections to the controllers can be established via the computer's standard Ethernet ports.

No further software is necessary for operation.

Note



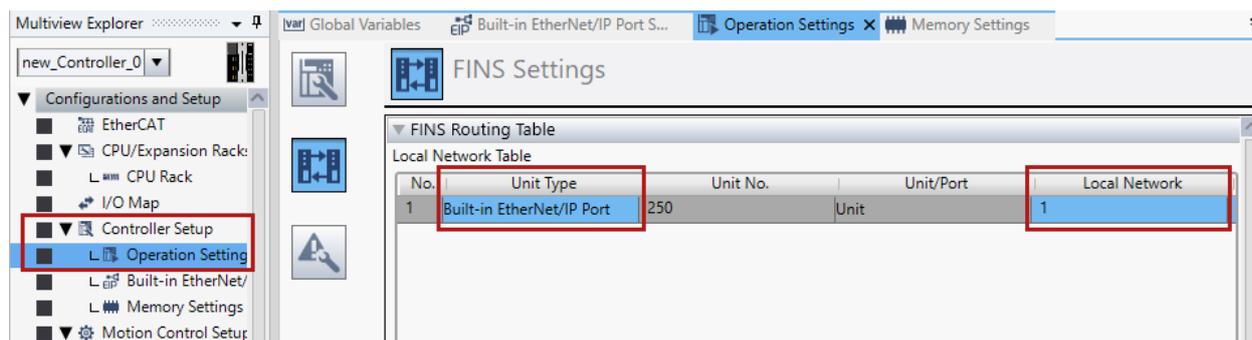
It is recommended carrying out the FINS communication on a separate network segment to exclude a mutual influence by other network components.

3.3 OMRON configuration and engineering

Configuration in Sysmac Studio

Before you can connect *ibaPDA* and the OMRON controller, you need to configure settings in Sysmac Studio under *Configuration and Setup – Controller Setup*.

1. Under *Operation Settings – FINS Settings* set the *Unit Type* to *Built-in EtherNet/IP Port* and the *Local Network* to "1".

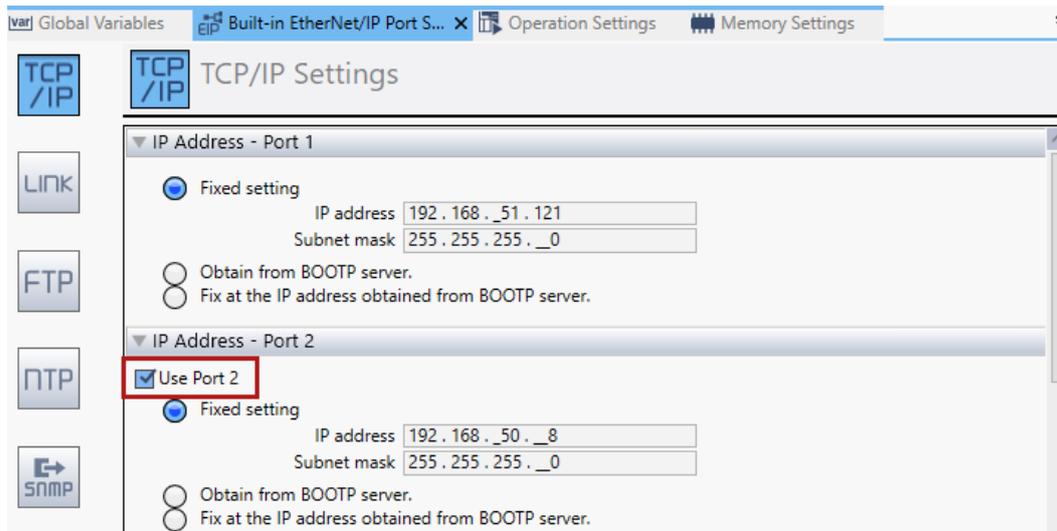


- Under *Built-in EtherNet/IP Port Setting – TCP/IP Settings* activate Port 2 via the *Use Port 2* selection field.
You cannot use the FINS protocol via Port 1.

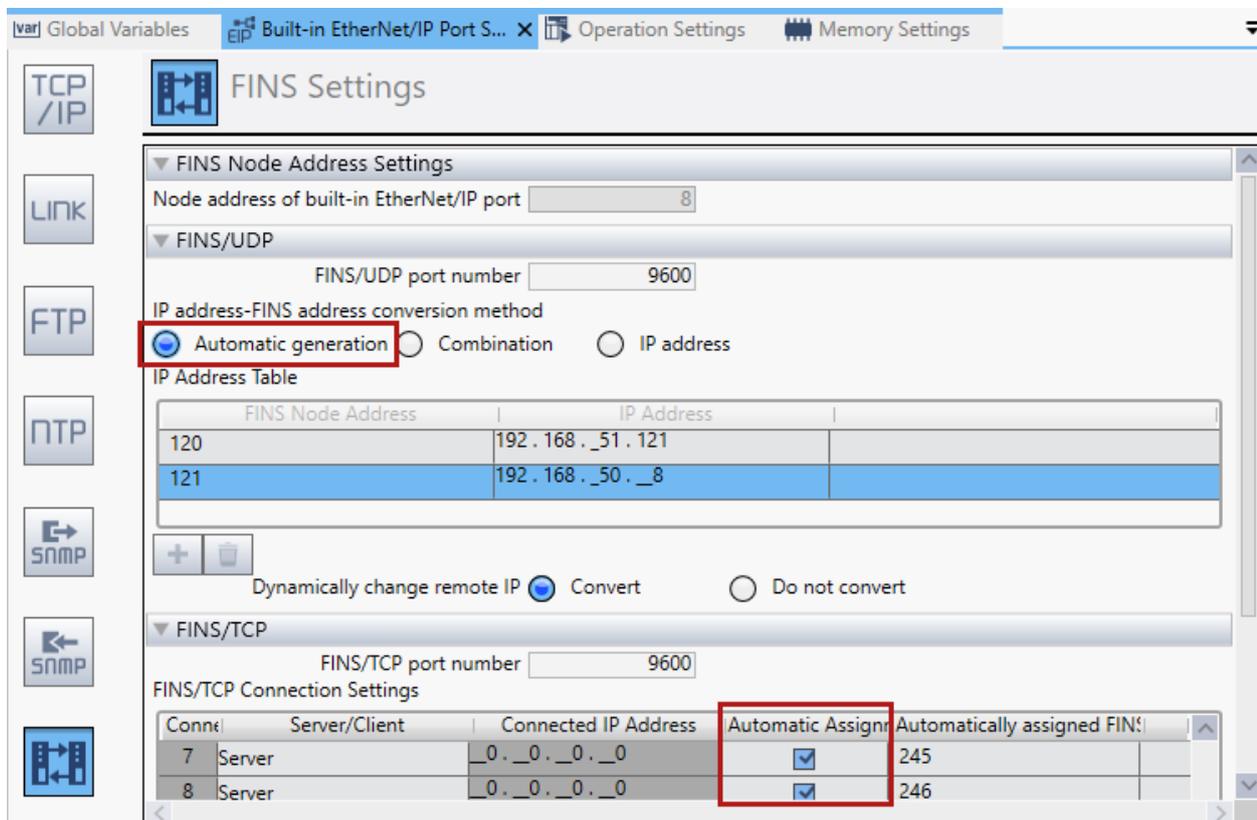
Note



For many devices, you must use an IP address whose last number is below 128. This is especially true if you enable automatic generation of the FINS Node Address.



- Under *Built-in EtherNet/IP Port Setting – FINS Settings* enable *Automatic Assignment* of the IP address for FINS/TCP for all servers.



- In the *Memory Settings* tab, enable the memory regions that can be used with FINS.

Enable	Area Type	Size (Number of Words)	Retain
<input checked="" type="checkbox"/>	CIO	6144	Not retained
<input checked="" type="checkbox"/>	WR	512	Not retained
<input checked="" type="checkbox"/>	HR	512	Retained
<input checked="" type="checkbox"/>	DM	32768	Retained
<input checked="" type="checkbox"/>	EM0	32768	Retained
<input checked="" type="checkbox"/>	EM1	32768	Retained
<input type="checkbox"/>	EM2	32768	Retained
<input type="checkbox"/>	EM3	32768	Retained

- In the *Global Variables* tab, assign the variables to be used to the previously enabled memory regions.

You must then specify the selected addresses in the *AT* column (memory area and address) in the same way in the FINS settings in *ibaPDA*.

You must also specify the access rights: Output – *ibaPDA* has read rights; Input – *ibaPDA* has write rights.

Name	Data Type	Initial Value	AT	Retain	Constant	Network Publish	Comment
SampleOut	ARRAY[0..35] OF BYTE			<input type="checkbox"/>	<input type="checkbox"/>	Output	
SampleIn	ARRAY[0..15] OF BYTE			<input type="checkbox"/>	<input type="checkbox"/>	Input	
Start1	BOOL		%W50.1	<input type="checkbox"/>	<input type="checkbox"/>	Do not publish	
Stop1	BOOL			<input type="checkbox"/>	<input type="checkbox"/>	Do not publish	
Wert1	INT		%H0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Do not publish	
Zaehler100ms	REAL		%W0	<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Zaehler10ms	REAL		%W2	<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Zaehler1s	REAL		%W4	<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Sinus_1	REAL		%W6	<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Cosinus_1	REAL		%E0_1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Byte_tag	BYTE		%D0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Byte_arr5_tag	ARRAY[0..4] OF BYTE			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Byte_matr23_tag	ARRAY[0..1,0..2] OF BY			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Word_tag	WORD		%D20	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Word_arr5_tag	ARRAY[0..4] OF WORD		%E1_0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Word_matr23_tag	ARRAY[0..1,0..2] OF W			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Real_tag	REAL		%D2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Real_arr5_tag	ARRAY[0..4] OF REAL			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Real_matr23_tag	ARRAY[0..1,0..2] OF RE			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
LReal_tag	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
LReal_arr5_tag	ARRAY[0..4] OF LREAL			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
LReal_matr23_tag	ARRAY[0..1,0..2] OF LR			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Int_tag	INT		%0	<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Int_arr5_tag	ARRAY[0..4] OF INT			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	
Int_matr23_tag	ARRAY[0..1,0..2] OF IN			<input type="checkbox"/>	<input type="checkbox"/>	Publish Only	

- When the parameterization is complete, load the changed settings into the corresponding controller.

Note



You cannot load the changes into the controller during operation. The controller is stopped for this purpose.

For further explanations of the Sysmac software and the available settings, please read the manual for the software.

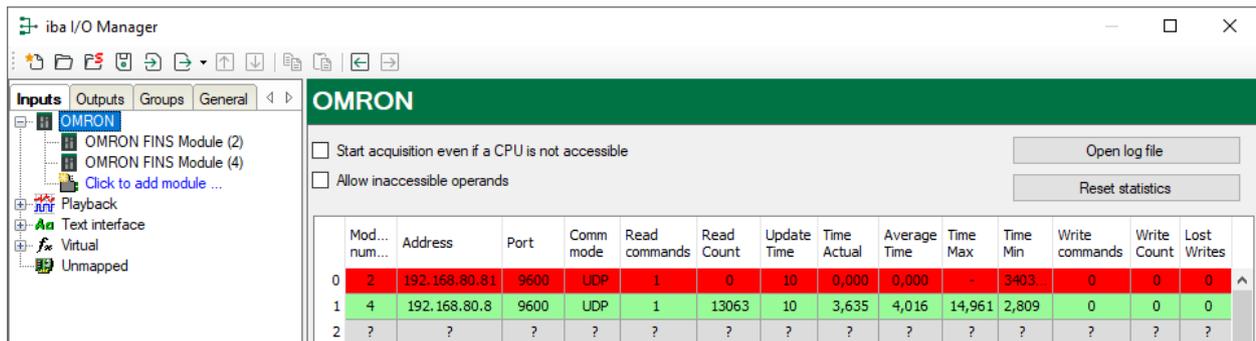
3.4 Configuration and engineering ibaPDA

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

3.4.1 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.



Mod... num...	Address	Port	Comm mode	Read commands	Read Count	Update Time	Time Actual	Average Time	Time Max	Time Min	Write commands	Write Count	Lost Writes	
0	2	192.168.80.81	9600	UDP	1	0	10	0,000	0,000	-	3403...	0	0	0
1	4	192.168.80.8	9600	UDP	1	13063	10	3,635	4,016	14,961	2,809	0	0	0
2	?	?	?	?	?	?	?	?	?	?	?	?	?	?

Start acquisition even if a CPU is not accessible

If this option is enabled, the acquisition starts even if a CPU is not accessible. Instead of an error, a warning is indicated in the validation dialog.

If the system has been started without connection to CPU, *ibaPDA* tries to connect with the CPU at regular intervals. The measured values remain at zero as long as the CPU is disconnected.

Allow inaccessible operands

Enable this option if you wish to start acquisition even if an operand is not accessible during the validation of the I/O configuration. If you do not enable this option, measurement does not start when inaccessible operands are present.

Connection table

For each connection, the table shows the connection status, the current values for the update time (current, real value, average, min. and max.) as well as the data size.

See [↗ Connection table](#), page 22.

<Open log file>

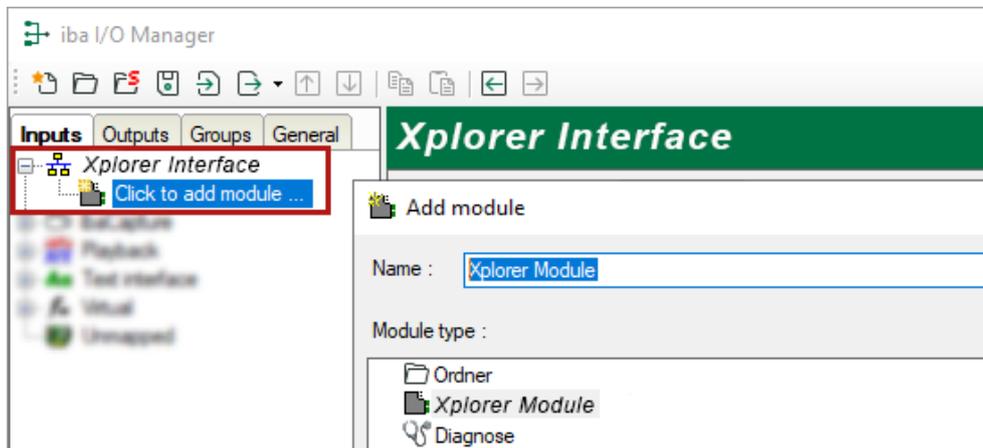
If connections to controllers have been established, all connection specific actions are recorded in a text file. Using this button, you can open and check this file. In the file system on the hard disk, you find the log files of the *ibaPDA* server (... \ProgramData\iba\ibaPDA\Log). The file name of 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.

3.4.2 Adding a module

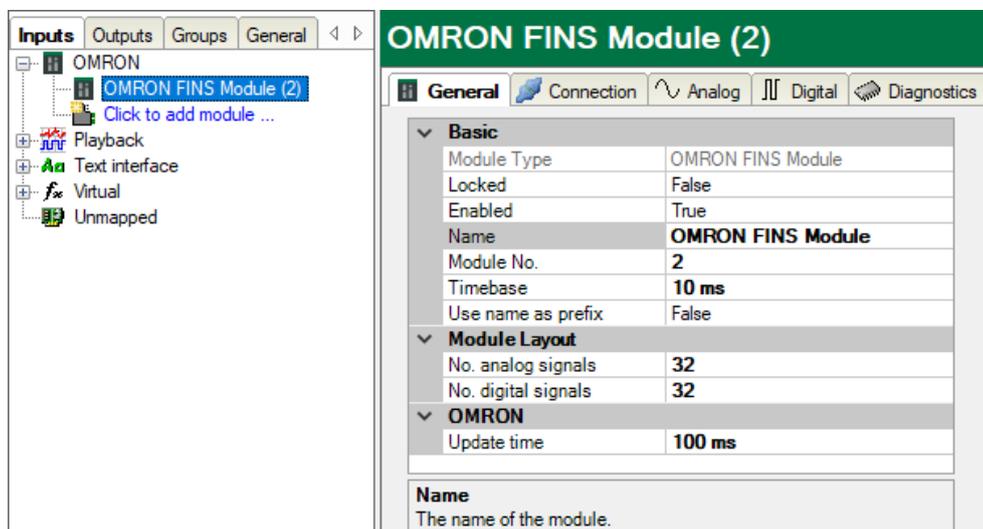
1. Click on the blue command *Click to add module...* located under each data interface in the *Inputs* or *Outputs* tab.
2. Select the desired module type in the dialog box and assign a name via the input field if required.
3. Confirm the selection with <OK>.



3.4.3 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.

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 name as prefix

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

Module Layout**No. of analog signals/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.

OMRON**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.

3.4.4 Connection settings

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

The screenshot shows the 'Connection' tab of a configuration window. It includes the following fields and controls:

- General Tab:** General, **Connection**, Analog, Digital, Diagnostics
- Connection Section:**
 - CPU address: 192.168.0.1
 - Port: 9600
 - Communication timeout (s): 3
 - Communication mode: TCP/IP (dropdown)
 - Nodes configured automatically (checkbox)
 - Memory addressing: CS (dropdown)
 - Buttons: Test connection, Scan available CPUs
- FINS settings Section:**
 - Remote net: 1, Remote node: 1, Remote unit: 0
 - Local net: 0, Local node: 217, Local unit: 0
 - Gateway count: 2, Max reads per message: 167

Connection

CPU Address

Enter the address of the OMRON CPU. The pre-defined default value is just an example.

Port

The default port number is 9600. You can change this port number, but you must specify the same port number in Sysmac Studio.

Communication timeout

Set the time after which *ibaPDA* aborts the connection attempt and declare the CPU as not reachable or non-responsive.

Communication mode

Set the TCP/IP or UDP/IP communication mode here. The communication mode depends on the OMRON controller you are using as well as the network setup.

Note



It is recommended to use the TCP/IP communication mode. Communication via UDP is unstable and can lead to communication failures.

Addressing the memory

The addressing mode CS or CV depends on the OMRON controller you are using. Use CV mode for CV-series controllers. Use CS mode for CV- or NX-series controllers.

<Test connection>

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

<Scan available CPUs>

Via this button you can scan the network to find available OMRON CPUs.

The scan is performed either via broadcast or subnet scan. Broadcast is usually limited to the subnet in which the *ibaPDA* computer is located, whereby no subnet boundaries are crossed. If you need a different subnet, you can perform a special subnet scan. The left tree structure shows all CPUs that were found. On the right side, the transferred data for each CPU is displayed. Here you find the IP address of the controller that you must use for the configured module.

OMRON CPU scan result

Scan subnet: 192 . 168 . 80 . 255

Scan available CPUs (scan subnet)

Scan available CPUs (broadcast)

192.168.80.8

Public IP:
192.168.80.8

Configuration IP:
0.0.0.0

Subnet:
255.255.255.0

HTTP port:
80

MAC address:
00:00:0A:D8:C0:49

Control Type:
NX102-9000

Version:
1.41.02

Control Name:
new_Controller_0

Take public IP

Take configuration IP

Cancel

FINS settings

You can configure certain parameters of the FINS protocol. However, these parameters must match the settings in Sysmac Studio, see [OMRON configuration and engineering](#), page 8.

Remote network

The default value is 1.

Remote node

The default value is the last value of the IP address. You can change this value in Sysmac Studio. In TCP mode, this value is set automatically when the communication is initialized.

Remote unit

The default value is 0.

Local network

The default value is 0.

Local node

You can assign this value freely, unless you have configured a special routing in Sysmac Studio. In TCP mode, this value is set automatically when the communication is initialized.

Local unit

The default value is 0.

Gateway counter

The default values are 2 or 7, depending on the network structure and the values you have set in Sysmac Studio.

Max. reads per message

The default value is 167. DeviceNet connections are an exception because only 89 items are allowed per telegram.

3.4.5 Signal configuration

In the *Analog* or *Digital* tab you configure the signals to be measured. In the *General* tab under *Module Layout* you define the length of the signal tables or the number of signals per table.

Note



Observe the maximum number of signals permitted by your license.

Note



Take into consideration that the number of signals, which are read by a CPU, influences the minimum achievable read cycle. The more signals recorded, the slower the reachable reading cycle.

When configuring the analog and digital signals, set the Memory Type, Address and Data Type as they are configured in Sysmac Studio, see [OMRON configuration and engineering](#), page 8.

General Connection Analog Digital Diagnostics							
Name	Unit	Gain	Offset	Memory Type	Address	Data Type	Active
0		1	0	D	0	INT	<input type="checkbox"/>
1		1	0	CIO	1	INT	<input type="checkbox"/>
2		1	0	W	2	INT	<input type="checkbox"/>
3		1	0	H	3	INT	<input type="checkbox"/>
4		1	0	A	4	INT	<input type="checkbox"/>
5		1	0	TIM	5	INT	<input type="checkbox"/>
6		1	0	CNT	6	INT	<input type="checkbox"/>
7		1	0	D	7	INT	<input type="checkbox"/>
8		1	0	E0	8	INT	<input type="checkbox"/>
9		1	0	E1	9	INT	<input type="checkbox"/>
10		1	0	E2	10	INT	<input type="checkbox"/>
11		1	0	E3	11	INT	<input type="checkbox"/>
12		1	0	E4	12	INT	<input type="checkbox"/>
13		1	0	E5	13	INT	<input type="checkbox"/>
14		1	0	E6	14	INT	<input type="checkbox"/>
15		1	0	E7	15	INT	<input type="checkbox"/>

Address information

The address is typically a word address for OMRON controllers. An address cannot be shared with two smaller data types. Therefore, during data transmission and mapping configuration in Sysmac Studio, there is the following usage breakdown:

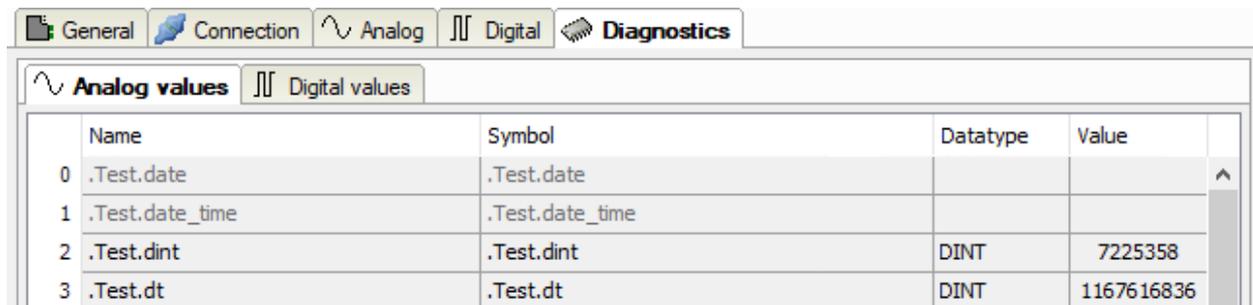
- one word: SINT, BYTE, WORD, INT
- two words: DWORD, FLOAT and DINT
- four words: DOUBLE and LINT

ibaPDA treats the Sysmac data types USINT, UINT and UDINT as BYTE, WORD and DWORD. ULINT is currently not supported.

For digital values, you must also configure the corresponding number of bits.

3.4.6 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.



The screenshot shows the 'Diagnostics' tab selected in the software interface. Below it, the 'Analog values' sub-tab is active, displaying a table with the following data:

	Name	Symbol	Datatype	Value
0	.Test.date	.Test.date		
1	.Test.date_time	.Test.date_time		
2	.Test.dint	.Test.dint	DINT	7225358
3	.Test.dt	.Test.dt	DINT	1167616836

Inactive signals are grayed out.

3.4.7 Output module

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

Note



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

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 *OMRON FINS* module in the *Inputs* register.

The settings correspond to those in the *Inputs* tab, apart from the module-specific settings, and can also be configured there. The connection settings also correspond to those in the *Inputs* tab.

For details concerning the module settings see [↗ General module settings](#), page 12.

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.

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 this interface 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.

License information		Licenses:
License container:	3-4	ibaPDA-Interface-Codesys-Xplorer (16)
Customer name:	ibaflex Technology	
License time limit:	Unlimited	
Container type:	WIBU CmStick v4.40	
Container host:		
Required EUP date:	01.02.2023	
EUP date:	31.12.2025	

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.

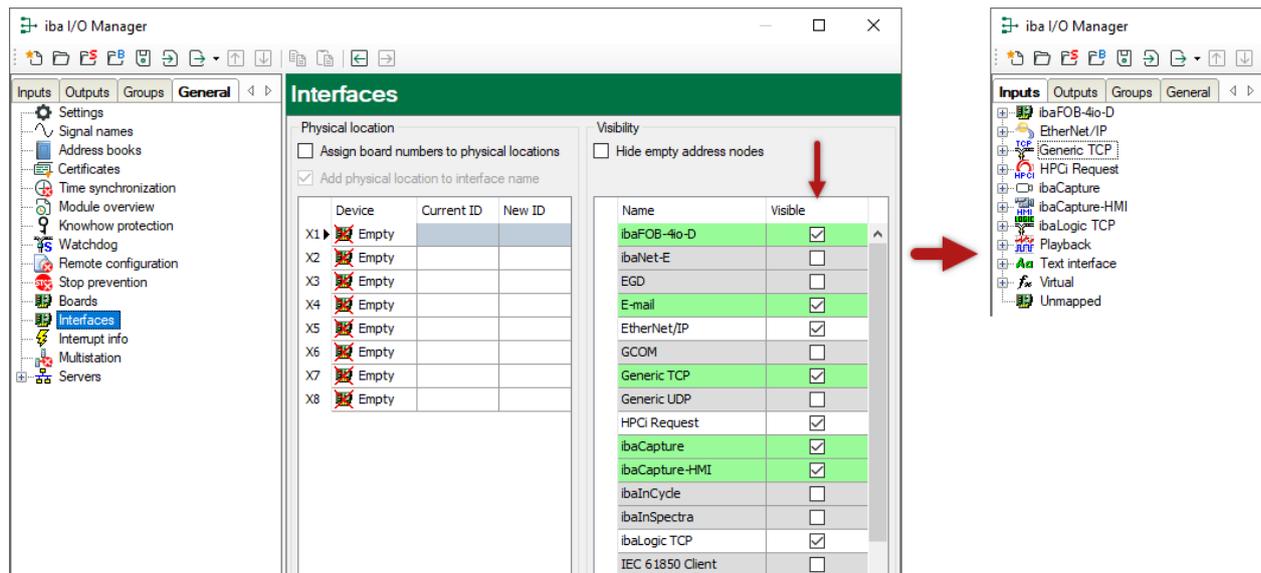
Visibility

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.

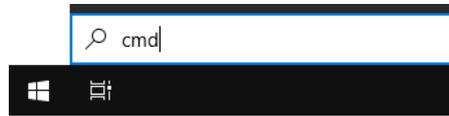
Examples:

- `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.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:
Reply from 192.168.1.10: bytes=32 time=1ms TTL30
Reply from 192.168.1.10: bytes=32 time<1ms TTL30
Reply from 192.168.1.10: bytes=32 time<1ms TTL30
Reply from 192.168.1.10: bytes=32 time<1ms TTL30

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Windows\system32>
```

→ With no existing connection you receive error messages.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping 192.168.1.10

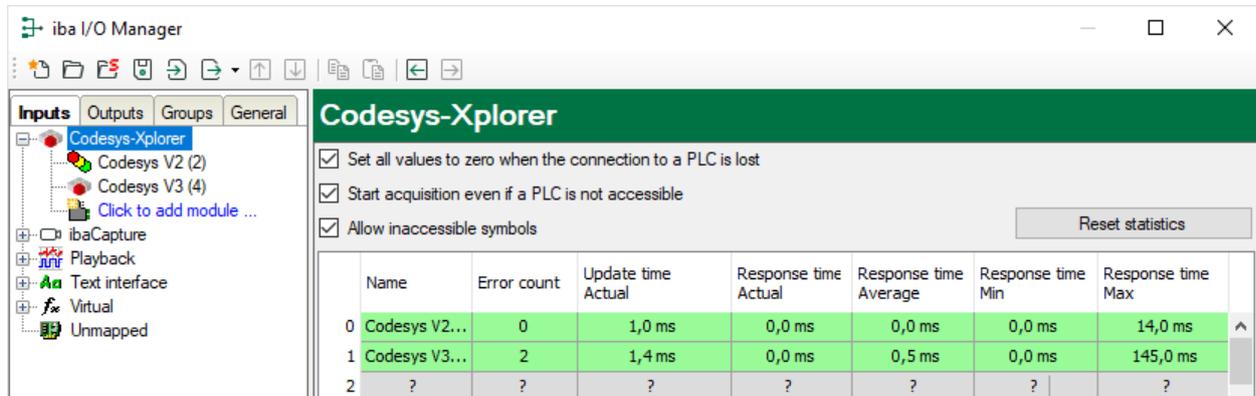
Pinging 192.168.1.10 with 32 bytes of data:
Reply from 192.168.1.10: Destination host unreachable.

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Windows\system32>
```

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.

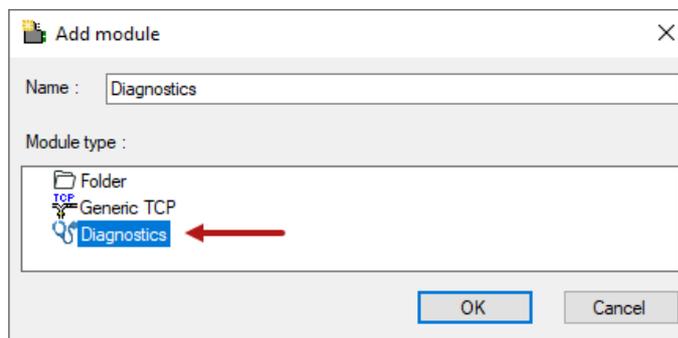
A diagnostic module is always assigned to a data acquisition module of the same interface and supplies its connection information. By using a diagnostic module you can record and analyze the diagnostic information continuously in the *ibaPDA* system.

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

Example for the use of diagnostic modules:

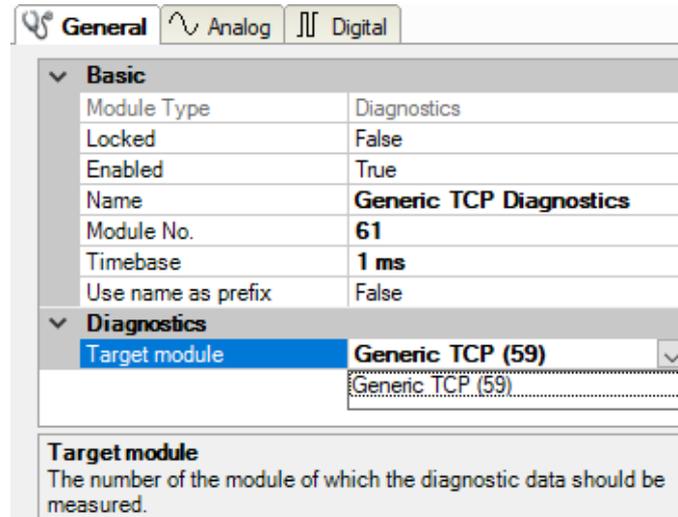
- A notification can be generated, whenever the error counter of a communication connection exceeds a certain value or the connection gets lost.
- In case of a disturbance, the current response times in the telegram traffic may be documented in an incident report.
- The connection status can be visualized 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):



Basic	
Module Type	Diagnostics
Locked	False
Enabled	True
Name	Generic TCP Diagnostics
Module No.	61
Timebase	1 ms
Use name as prefix	False

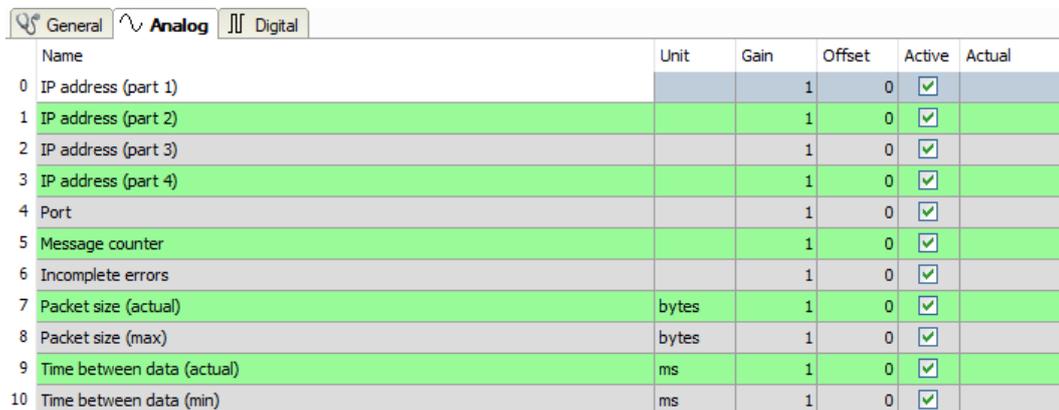
Diagnostics	
Target module	Generic TCP (59)

Target module
The number of the module of which the diagnostic data should be measured.

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

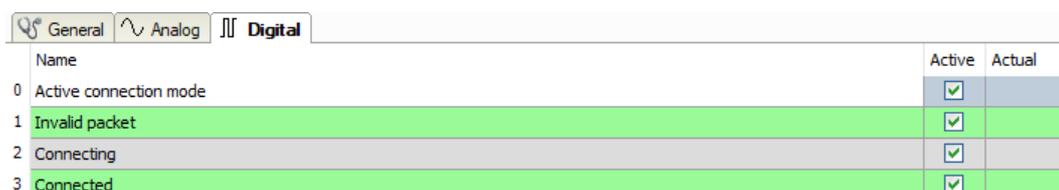
There is only one setting which is specific for the diagnostic module: the target module.

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	<input checked="" type="checkbox"/>	
1 IP address (part 2)		1	0	<input checked="" type="checkbox"/>	
2 IP address (part 3)		1	0	<input checked="" type="checkbox"/>	
3 IP address (part 4)		1	0	<input checked="" type="checkbox"/>	
4 Port		1	0	<input checked="" type="checkbox"/>	
5 Message counter		1	0	<input checked="" type="checkbox"/>	
6 Incomplete errors		1	0	<input checked="" type="checkbox"/>	
7 Packet size (actual)	bytes	1	0	<input checked="" type="checkbox"/>	
8 Packet size (max)	bytes	1	0	<input checked="" type="checkbox"/>	
9 Time between data (actual)	ms	1	0	<input checked="" type="checkbox"/>	
10 Time between data (min)	ms	1	0	<input checked="" type="checkbox"/>	

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	<input checked="" type="checkbox"/>	
1 Invalid packet	<input checked="" type="checkbox"/>	
2 Connecting	<input checked="" type="checkbox"/>	
3 Connected	<input checked="" type="checkbox"/>	

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 <i>ibaPDA</i>). Corresponds to the assembly instance number
Connection ID T->O	ID of the connection for input data (from <i>ibaPDA</i> to target system). Corresponds to the assembly instance number
Connection phase (in)	Status of the <i>ibaNet-E</i> data connection for reception (in)
Connection phase (out)	Status of the <i>ibaNet-E</i> 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 <i>ibaPDA</i>)
Destination IP address (part 1-4) T->O	4 octets of the IP address of the target system Input data (from <i>ibaPDA</i> 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 (<i>ibaPDA</i> receives)
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 <i>ibaPDA</i> to the PLC/CPU
Output data length	Length of the data messages with output signals in bytes (<i>ibaPDA</i> 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 <i>ibaPDA</i> 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 <i>ibaPDA</i>)
Source IP address (part 1-4) T->O	4 octets of the IP address of the target system Input data (from <i>ibaPDA</i> 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 <i>ibaPDA</i> and the <i>ibaNet-E</i> 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

5 Support and contact

Support

Phone: +49 911 97282-14
Fax: +49 911 97282-33
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
Fax: +49 911 97282-33
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.