ibaPDA-Interface-MELSEC-Xplorer

PLC-Xplorer Data Interface to Mitsubishi MELSEC Systems

Manual

Issue 1.4

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

Version	Date	Revision	Author	Version SW
1.4	02-2023	GUI new, changes in GX Works ethernet module	RM, mm	8.2.0

Content

1 About this Manual

This document describes the function and application of the software interface

ibaPDA-Interface-MELSEC-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 documentation addresses qualified professionals, who are familiar with handling electrical and electronic modules as well as communication and measurement technology. A person is regarded as a professional if he/she is capable of assessing the work assigned to him/her and recognizing possible 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-MELSEC-Xplorer* the following basic knowledge is required and/or useful:

- Windows operating system
- 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 MELSEC-Xplorer

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

- *ibaPDA* v7.1.7 or more recent
- Base license for *ibaPDA*+ license for *ibaPDA-Interface-PLC-Xplorer* or *ibaPDA-Interface-MELSEC-Xplorer*
- With more than 16 connections you need additional *one-step-up-Interface-MELSEC-Xplorer* licenses for each additional 16 connections.

Note

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

ibaPDA supports the following MELSEC controller families:

- MELSEC-A
- MELSEC FX3U
- MELSEC-Q
- MELSEC-L
- MELSEC iQ-R
- MELSEC iQ-F

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

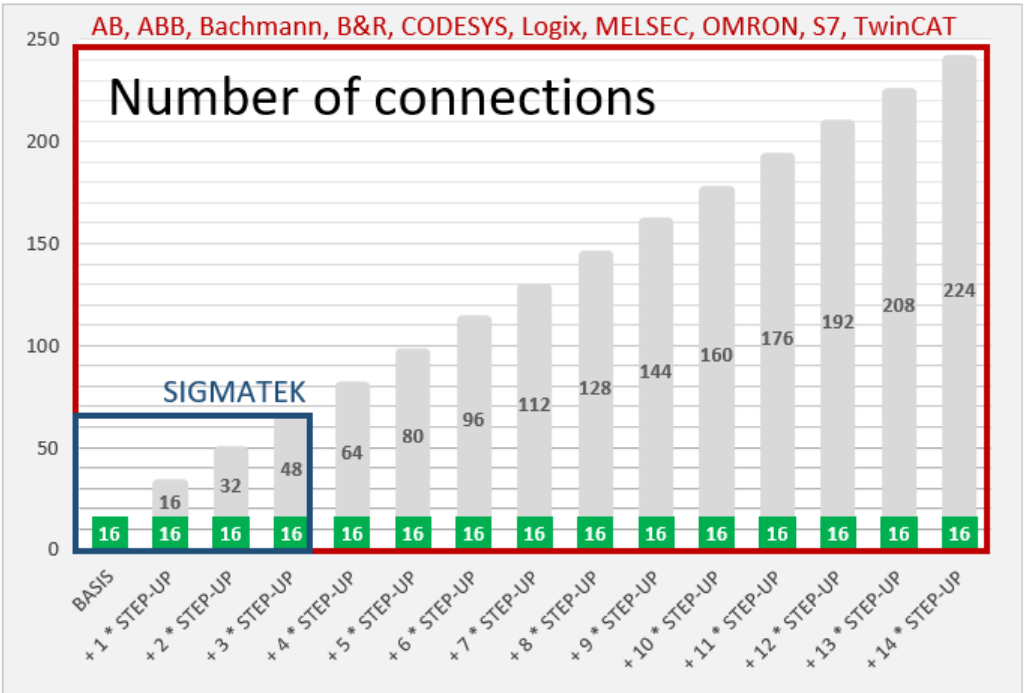
License information

Order no.	Product name	Description
31.001042	ibaPDA-Interface-PLC-Xplorer	Extension license for an *ibaPDA* system adding all available Xplorer data interfaces. (Complete specification under www.iba-ag.com)
31.000008	ibaPDA-Interface-MELSEC-Xplorer	Extension license for an *ibaPDA* system adding the data interface: + MELSEC-Xplorer (16 connections)
31.100008	One-step-up-Interface-MELSEC-Xplorer	Extension license for 16 additional MELSEC-Xplorer connections (max. 14 extension licenses)

Table 1: Available MELSEC-Xplorer interface 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 interface to MELSEC controllers

3.1 General information

The MELSEC-Xplorer interface is suitable for measurement data acquisition with *ibaPDA* from Mitsubishi MELSEC PLCs. The data is cyclically read by *ibaPDA* instead of being sent by the PLC.

3.2 System topologies

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

No further software is necessary for operation.

Note

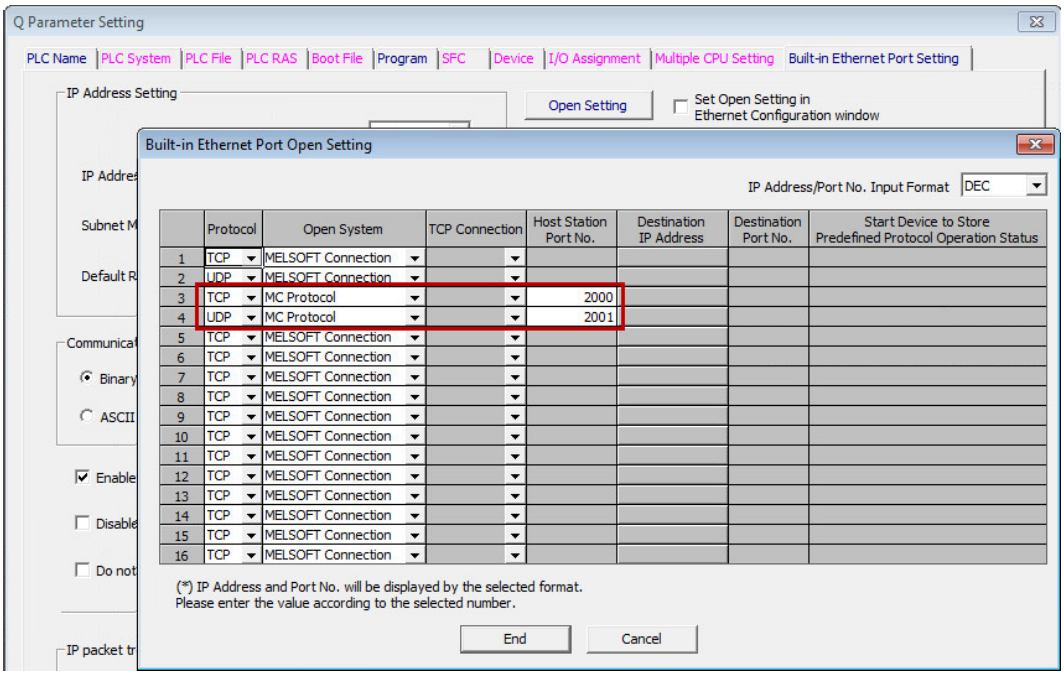
It is recommended carrying out the TCP/IP communication on a separate network segment to exclude a mutual influence by other network components.

3.3 Configuration and engineering GX Works 2

3.3.1 Configuration CPU

For access to the MELSEC controllers, *ibaPDA* uses the Mitsubishi MC Protocol.

Activate the Mitsubishi MC Protocol in the PLC parameters on the controller side.



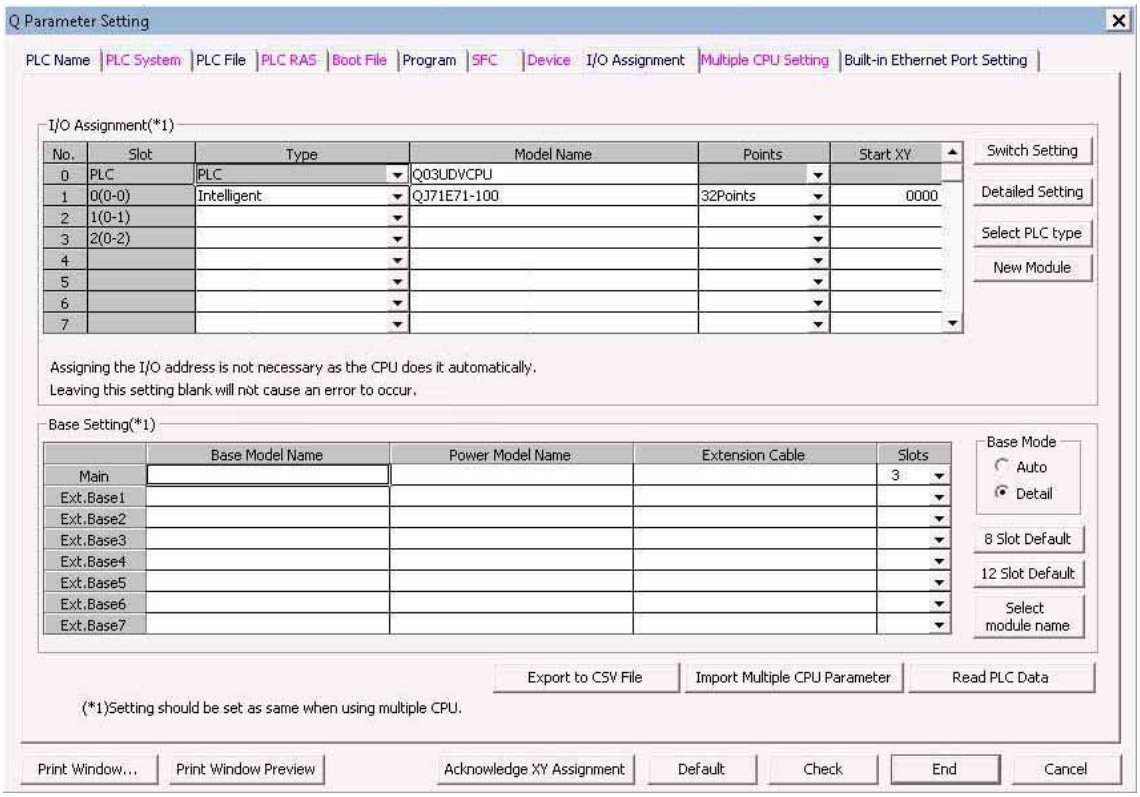
Apart from that no further configuration and programming is required on the controller side. In particular, it is not necessary to call any program modules.

3.3.2 Configuration Ethernet module

The communication with the CPU is also possible via a MELSEC Ethernet module over TCP. First configure the hardware and then set the network parameters. The single steps are described below.

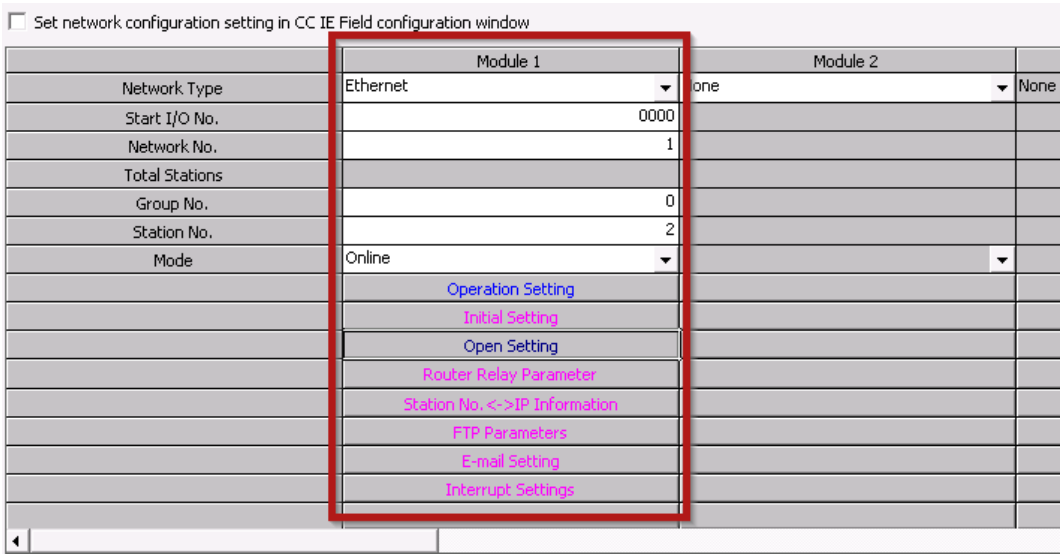
Hardware configuration

1. Open the PLC setting dialog and select the *I/O Assignment* tab.
2. Enter the corresponding communication module.

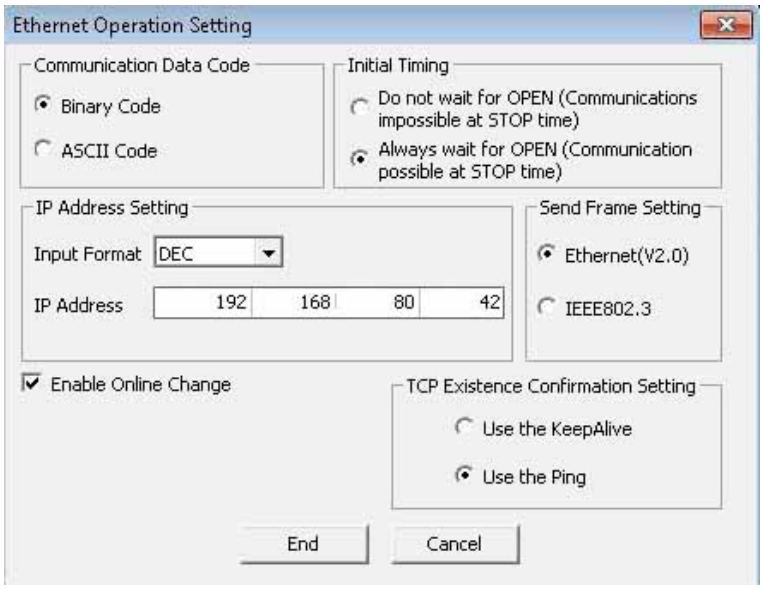


Network parameter configuration

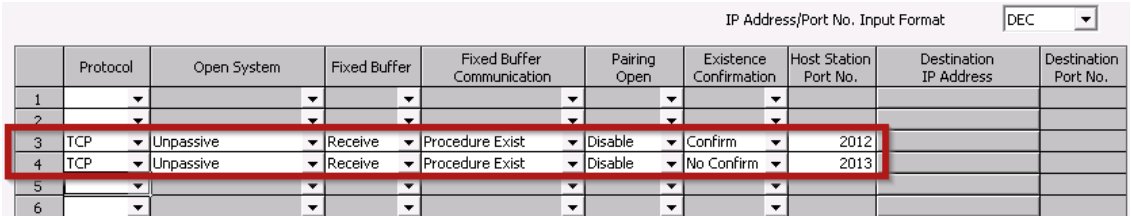
1. Open *Ethernet Settings* and enter the corresponding parameters.



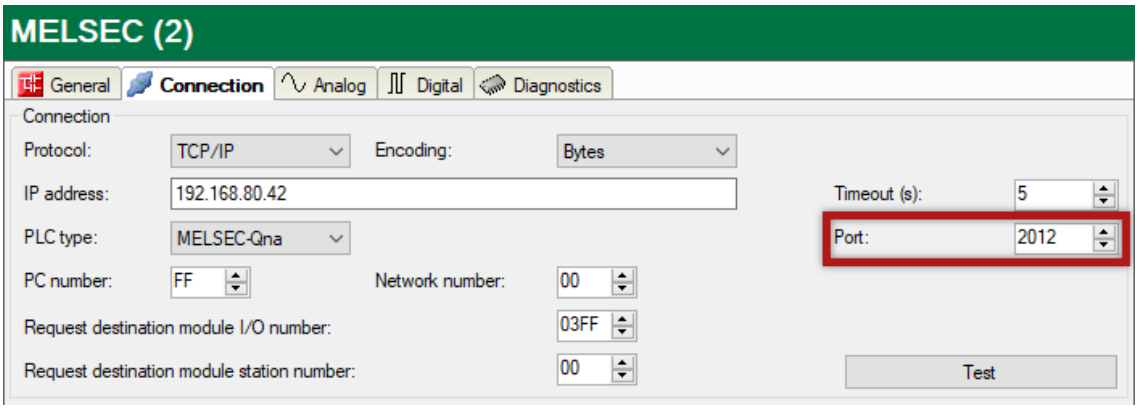
2. Open *Operation Settings* and enter the corresponding parameters.



3. Open *Open Settings*. Here you have to open the ports for the connection to *ibaPDA*. The example shows ports 2012 and 2013. (Both settings are functioning but only one of the two is necessary.)



4. In the MELSEC-Xplorer module in *ibaPDA*, configure the same port as previously configured. The example shows port 2012.



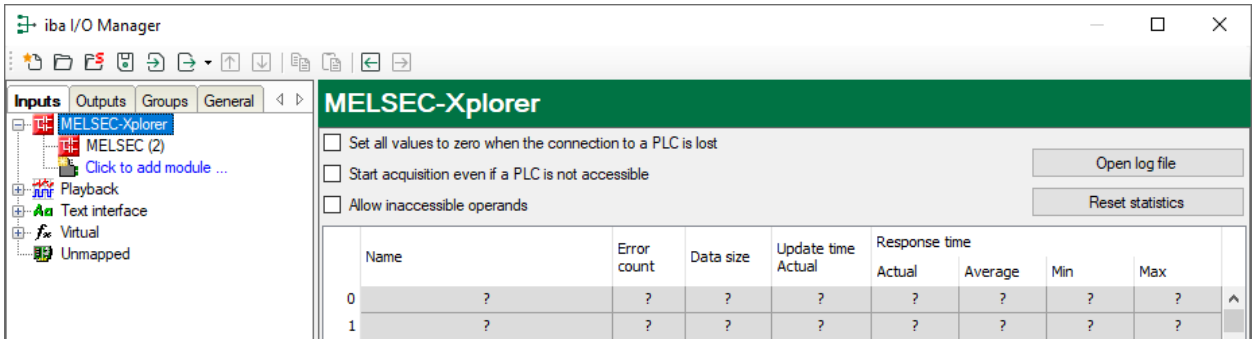
3.4 Configuration and engineering ibaPDA

The engineering for *ibaPDA* is described in the following. If all system requirements are fulfilled, *ibaPDA* displays the *MELSEC-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.



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 has been started without a connection to the controller, *ibaPDA* periodically tries to connect to the PLC.

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.

<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.

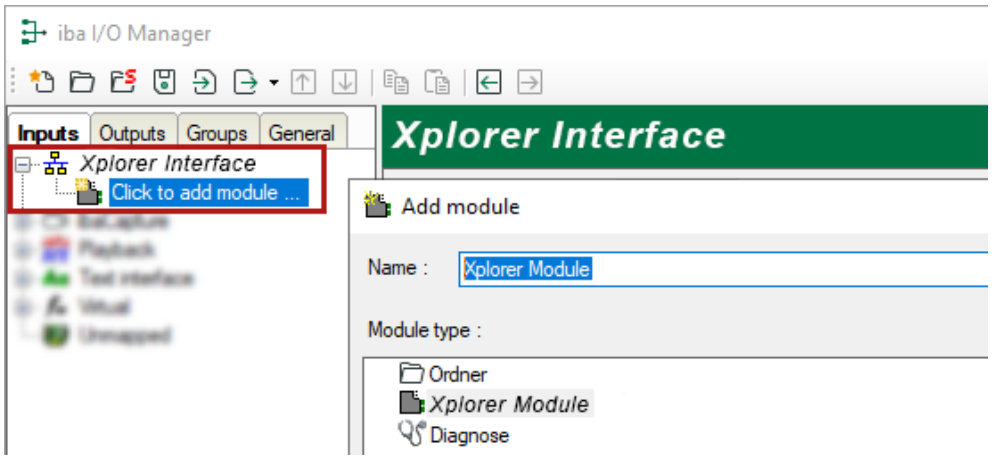
Connection table
For each connection, the table shows the connection status, the current values for the update time (current, real value, average, min. and max.) as well as the data size. In addition, there is an error counter for the individual connections during the acquisition.

See ↗ *Connection table*, page 22.

- Data size: The data size shows how much data is read per read operation; in between brackets, the number of commands used to request the data is displayed.
- Response time: The time it takes to read the data for a connection. The table shows the actual, average, minimum and maximum values of the response time.
- Update time: The update time indicates the time between 2 read operations.

3.4.2 Adding a module

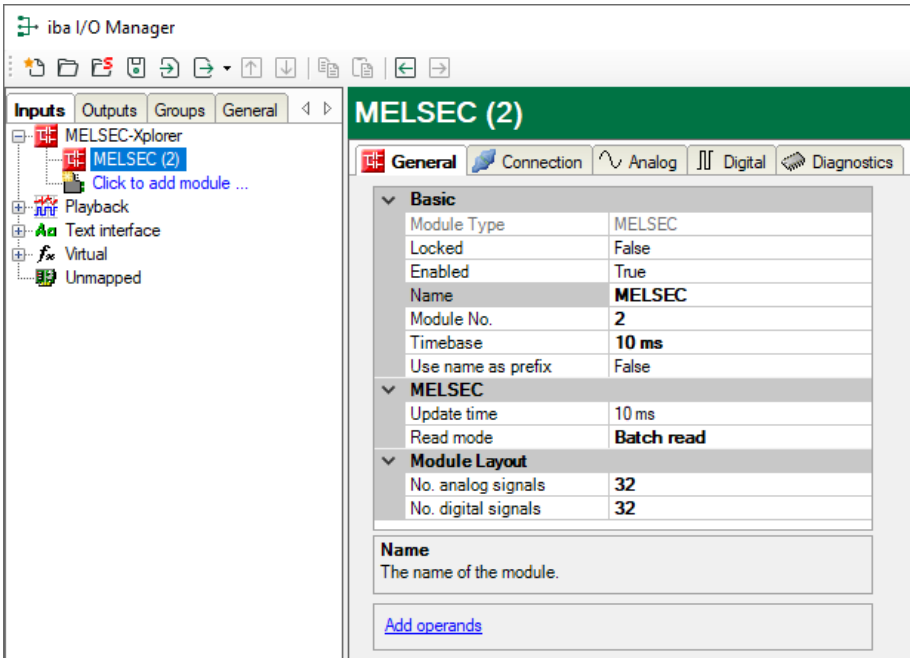
1. Click on the blue command *Click to add module…* located under each data interface in the *Inputs* or *Outputs* tab.
2. Select the desired module type in the dialog box and assign a name via the input field if required.
3. Confirm the selection with <OK>.



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.

MELSEC

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.

Note

It is recommended to check the diagnostic overview (connection table) for measured update rates as overload will result in lost samples.

Read mode

The read mode defines how data is read from the PLC.

- Batch read: Most generic way of reading data. Neighboring operands are grouped into a single request message; otherwise a single message per operand is used. This is the least performant method but supported by most PLC types.
- Random read: Most suited for situations where a lot of non-neighboring operands are requested. Not supported by all PLC types.
- Monitor: Most performant method. The requested operands are registered once and the data is requested periodically. Limited to 192 operands and can only be used in one module per PLC. Not supported by all PLC types.

Module Layout

No. of analog signals/digital signals

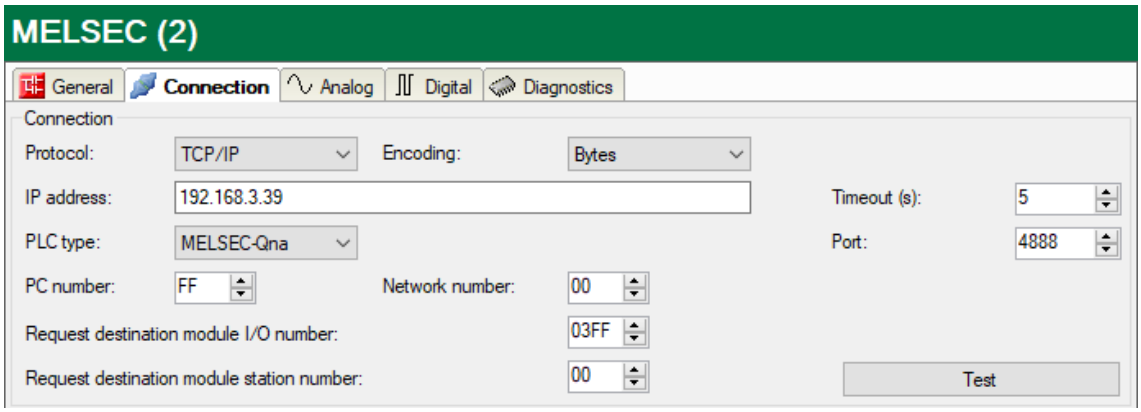
Define the number of configurable analog and digital signals in the signal tables. The default value is 32 for each. The maximum value is 1000. The signal tables are adjusted accordingly.

Link "Add operands"

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

3.4.4 Connection settings

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



Protocol
You can select either "TCP/IP" or "UDP". Note that some PLC types only support TCP/IP.

You have to enable the used protocol within the PLC parameters settings of GX works (see chapter ↗ *Configuration CPU*, page 8).

Other documentation

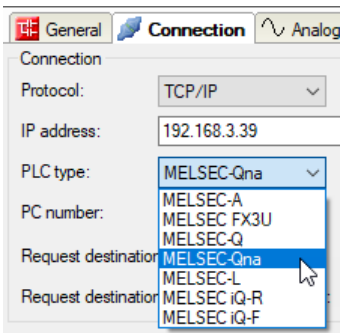
Refer to the particular PLC manual for more information.

Encoding
You can select either "Bytes" or "ASCII", depending on the configuration of your PLC. If possible, it is recommended to use the bytes-encoding method. This way, the size of the data request message is smaller and the processing is more efficient at the PLC side.

IP address
Enter the IP address at which the network interface of the PLC is located.

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

PLC type
Specify to which MELSEC series the PLC is assigned to.



Port
Specify the TCP or UDP port the PLC refers to for incoming requests.

PC number
Specify the network module station number of the access target. The default value is "FF".

Network number
Specify the network number of the access target. The default value is "00".

Request destination module I/O number
When accessing a multidrop connection station, the start input/output number of a multidrop connection source module is to be specified. The default value is "03FF".

Request destination module station number
When accessing a multidrop connection station, specify the station number of an access target module. The default value is "00".

Other documentation

Refer to the Mitsubishi MELSEC documentation for more information on *PC number*, *Network number*, *Request destination module I/O number* or *Request destination module station number*.

<Test>
ibaPDA tests the connection to the CPU and displays available diagnostic data, e.g. the PLC model name and the PLC type. If this option is not supported by the PLC, a warning message will be generated. However, this does not imply that requesting data will not work.

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.

Selection of the signals to be measured

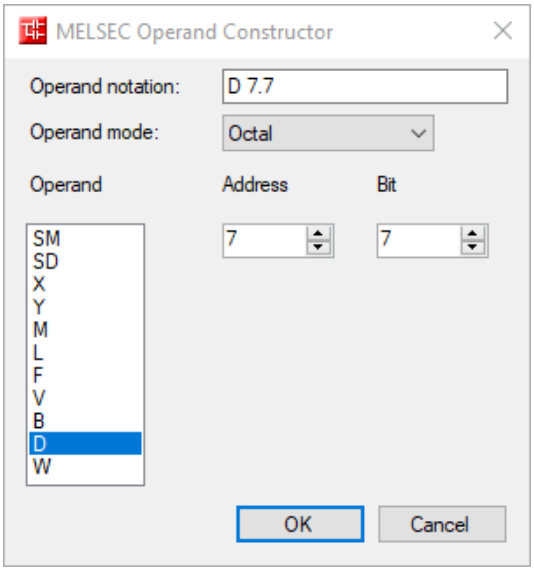
You have two options to select the signals to be measured:

- In the module's *General* tab, click on the *Add operands* hyperlink.

  The MELSEC Operand Constructor opens.

- Click on a field in the *Analog* or *Digital* tab, click in a cell in the *MELSEC Operand* column.

  The button <...> appears. Click the <...> button to open the MELSEC Operand Constructor.



Using the MELSEC Operand Constructor, you can easily add analog and digital signals to the MELSEC-Xplorer module. Depending on which signal tab is selected (*Analog* or *Digital*) and the PLC type configured in the *Connection* tab (see ↗ *Connection settings*, page 15), a list of accessible operands is available. In the digital operand type list, the analog operand types are listed as well.

User interface of the MELSEC Operand Constructor

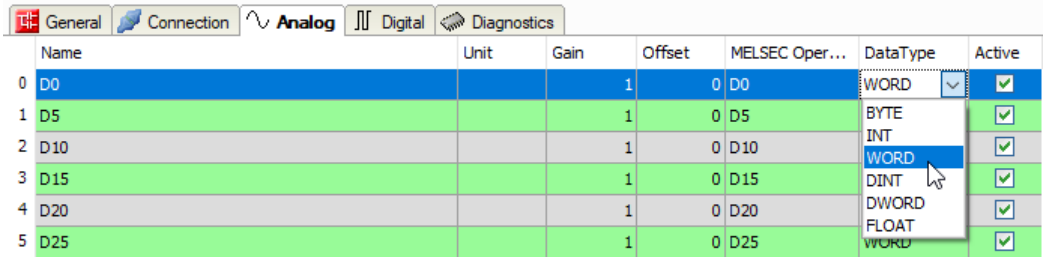
Operand mode
The following number systems are supported: octal, decimal and hexadecimal.

Address
Apart from the operand type, you also have to enter the address.

Bit
Since the digital operands are still analog data types in the PLC, you also have to specify the bit number you want to use for the digital signal.

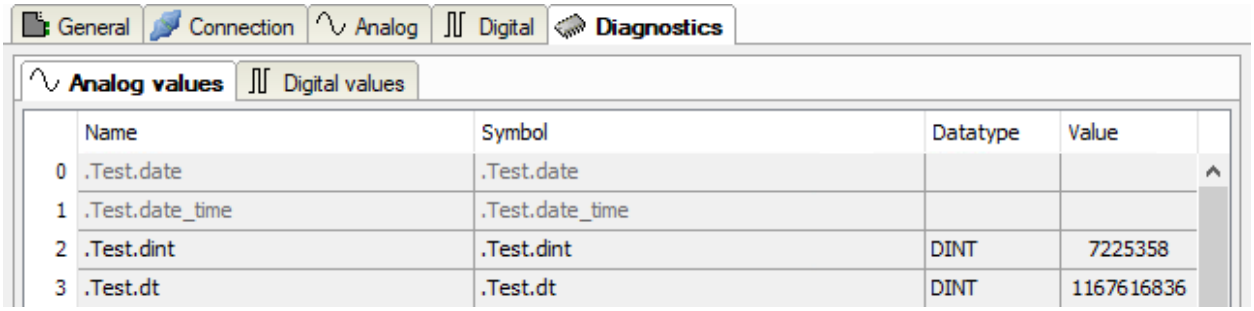
The figure below shows an example of an *Analog* tab of a MELSEC-Xplorer module. Apart from the standard properties of an analog signal, there is a column for editing the MELSEC Operand. You can also configure the data type of the requested operand here. *ibaPDA* supports the following data types are supported: BYTE, INT, WORD, DINT, DWORD and FLOAT.

The *Digital* tab shows a similar grid for the digital signals.



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.



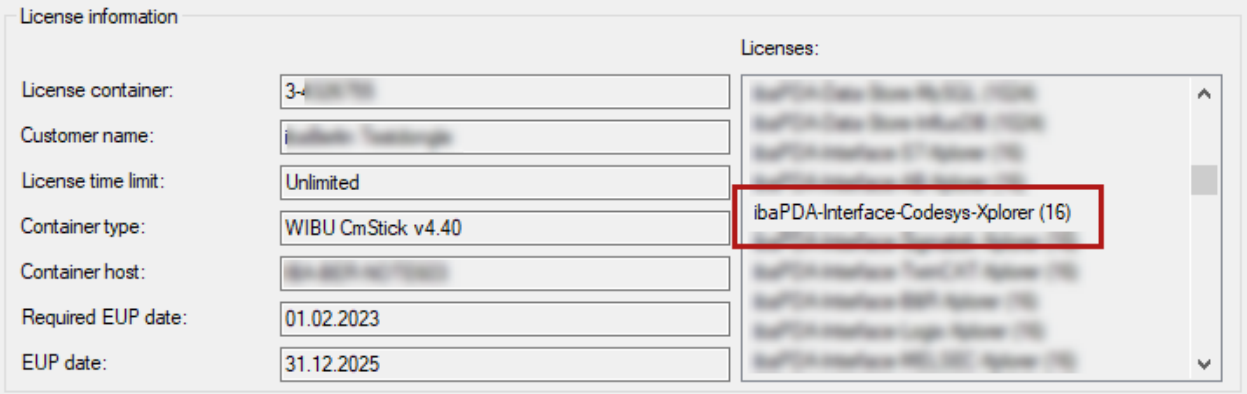
Inactive signals are grayed out.

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.



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.

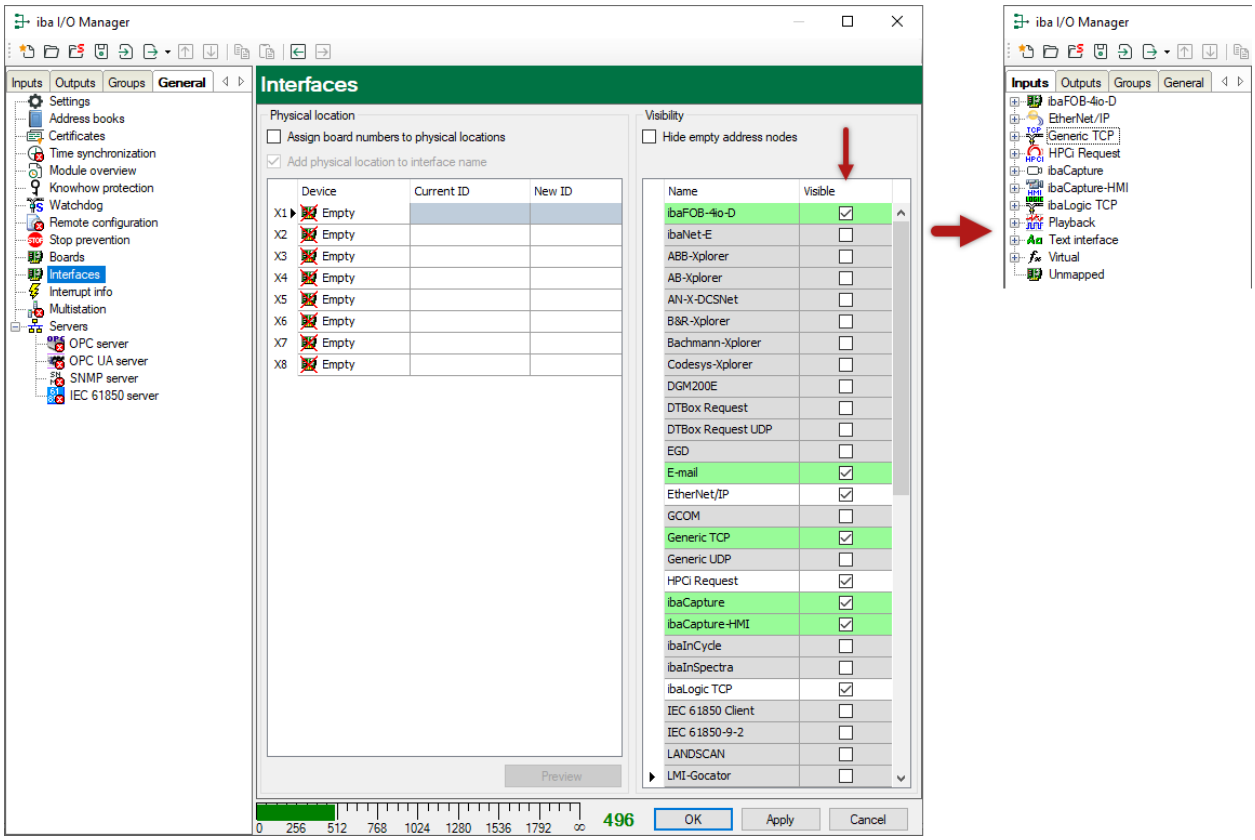
Visibility

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.

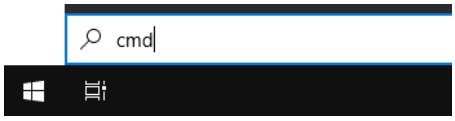
Examples:

- `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

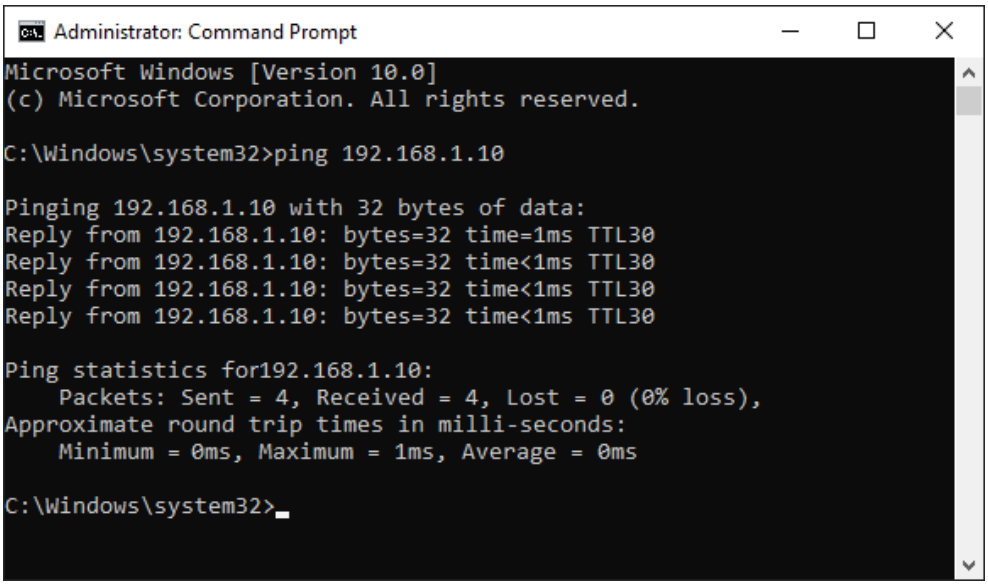
PING is a system command with which you can check if a certain communication partner can be reached in an IP network.

1. Open a Windows command prompt.

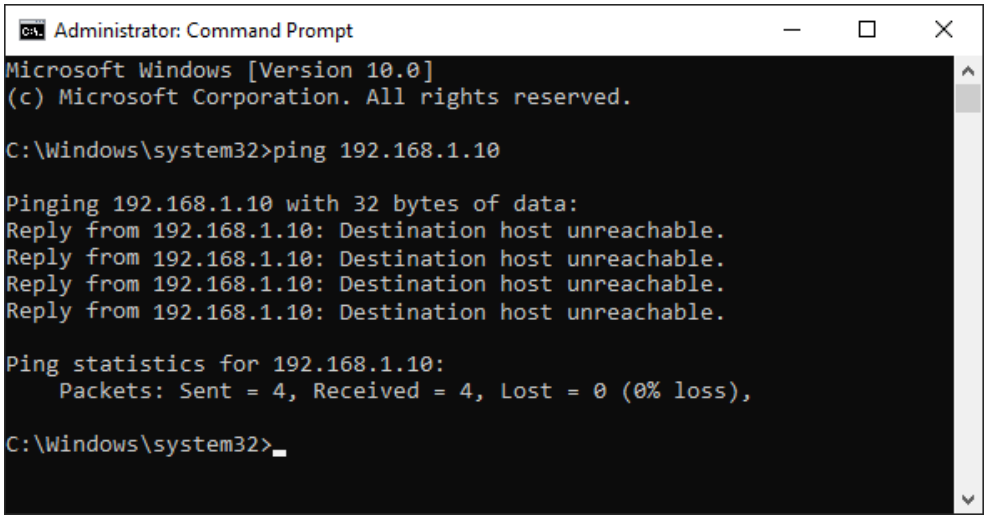


2. Enter the command "ping" followed by the IP address of the communication partner and press <ENTER>.

→ With an existing connection you receive several replies.

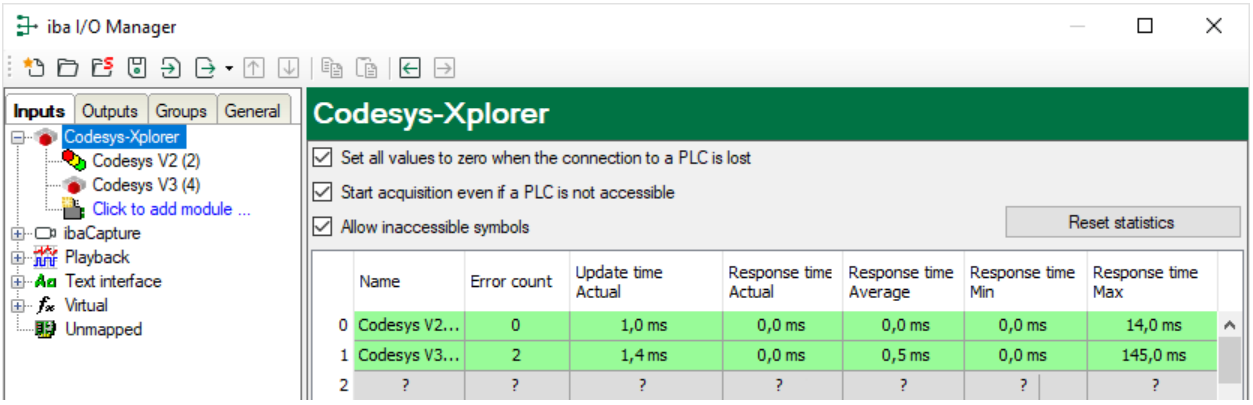


→ With no existing connection you receive error messages.



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.

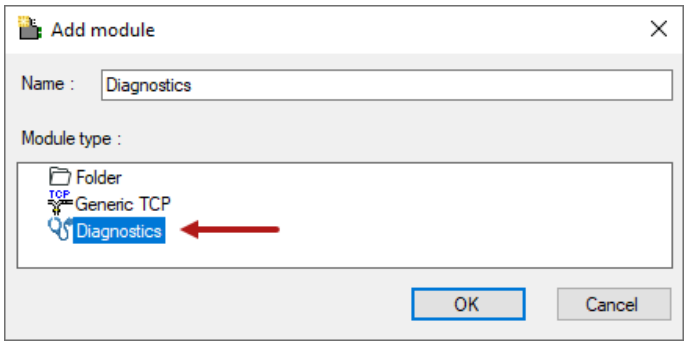
A diagnostic module is always assigned to a data acquisition module of the same interface and supplies its connection information. By using a diagnostic module you can record and analyze the diagnostic information continuously in the *ibaPDA* system.

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

Example for the use of diagnostic modules:

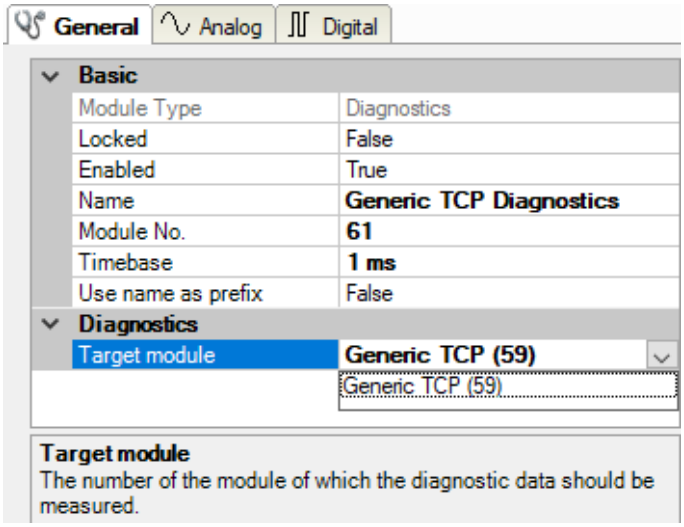
- A notification can be generated, whenever the error counter of a communication connection exceeds a certain value or the connection gets lost.
- In case of a disturbance, the current response times in the telegram traffic may be documented in an incident report.
- The connection status can be visualized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

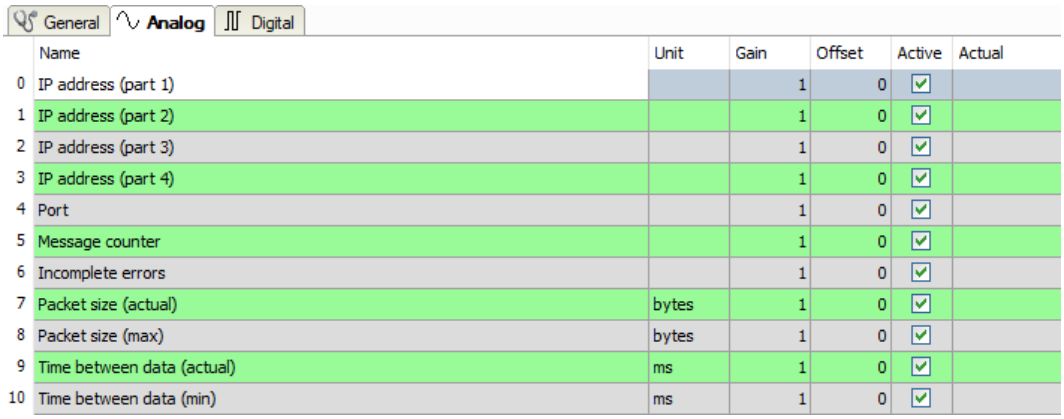
For a diagnostic module, you can make the following settings (example: Generic TCP):



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

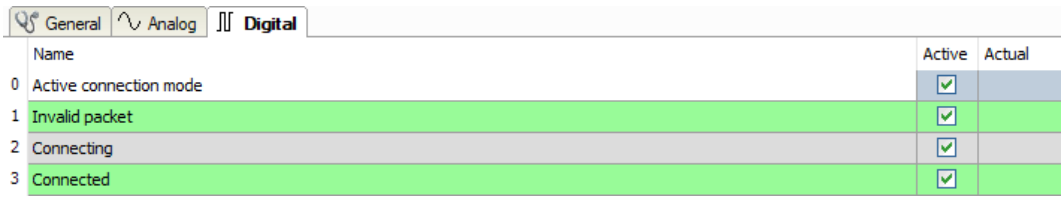
There is only one setting which is specific for the diagnostic module: the target module.

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
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
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 packet	Invalid data packet detected
IP address (part 1-4)	4 octets of the IP address of the target system

Signal name	Description
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
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

Signal name	Description
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

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.