



ibaPDA-Interface-ABB-Xplorer

PLC-Xplorer Data Interface for ABB AC800
controllers

Manual

Issue 1.2

Measurement Systems for Industry and Energy

www.iba-ag.com

Manufacturer

iba AG
Koenigswarterstr. 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 2022, 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 - Chapter / Page	Author	Version SW
1.2	08-2022	Direct access to PLC variables, output module	RM/IP	8.0.3

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.

Content

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	6
3	PLC-Xplorer data interface for AC800 controllers	8
3.1	General Information	8
3.2	System topology	8
3.3	Configuration and engineering of the AC800 controller.....	9
3.3.1	Access variables in the ABB Compact Control Builder.....	9
3.4	Configuration and engineering ibaPDA.....	13
3.4.1	General interface settings.....	13
3.4.2	Adding a module.....	14
3.4.3	General module settings.....	15
3.4.4	Connection settings	16
3.4.5	Configuration ibaPDA output module	18
3.4.5.1	General module settings.....	18
3.4.5.2	Use of the output module	19
3.4.6	Signal configuration	20
3.4.7	Module diagnostics.....	23
4	Diagnostics.....	24
4.1	License	24
4.2	Log files.....	24
4.3	Connection diagnostics with PING.....	25
4.4	Connection table	26
4.5	Diagnostic modules	27
5	Support and contact.....	32

1 About this Manual

This document describes the function and application of the software interface

ibaPDA-Interface-ABB-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-ABB-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 <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	<i>Filename, Path</i> Example: <i>Test.docx</i>

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

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

- *ibaPDA* V8.0.0 or higher
- Base license for *ibaPDA* + license for *ibaPDA-Interface-PLC-Xplorer* or *ibaPDA-Interface-ABB-Xplorer*
- With more than 16 connections you need additional *one-step-up-Interface-ABB-Xplorer* licenses for each additional 16 connections

Note



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

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

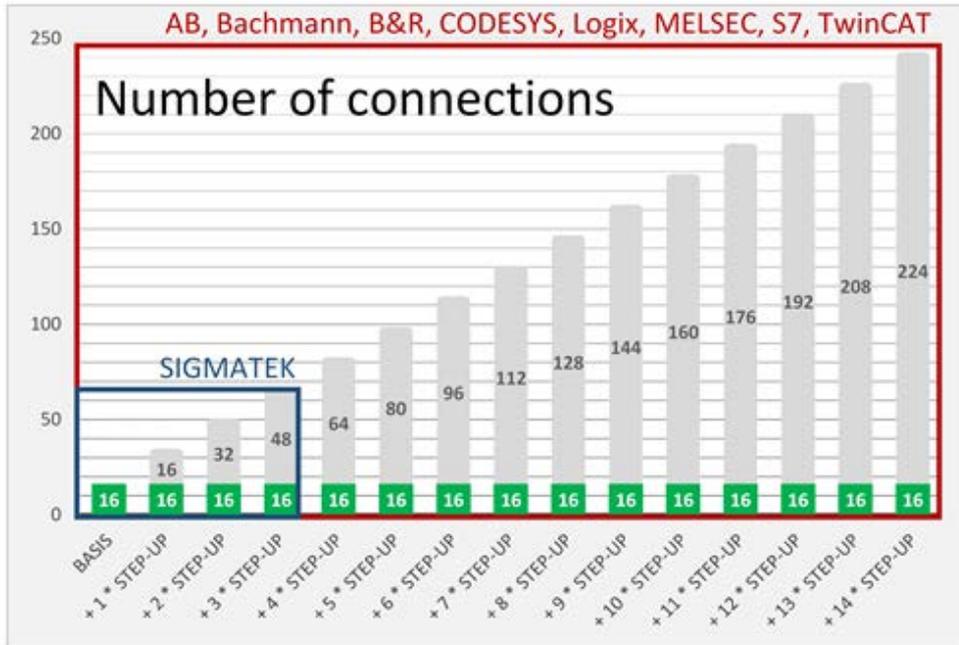
License information

Order no.	Product name	Description
31.001042	ibaPDA-Interface-PLC-Xplorer	Extension license for <i>ibaPDA</i> -System. All additional Xplorer-data interfaces are added. (Complete specifications under www.iba-ag.com)
31.000009	ibaPDA-Interface-ABB-Xplorer	Extension license for an <i>ibaPDA</i> system adding the data interface: + ABB-Xplorer
31.100009	one-step-up-Interface-ABB-Xplorer	Extension license for 16 additional ABB-Xplorer-connections (a maximum of 14 permissible)

Note



If you intend to use more than 16 data links 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.



Please note, that a regular *ibaPDA* license (at least *ibaPDA-V7-64*) is the precondition for the upgrades.

You have to take into consideration the limitation of the number of signals by the *ibaPDA* base license.

3 PLC-Xplorer data interface for AC800 controllers

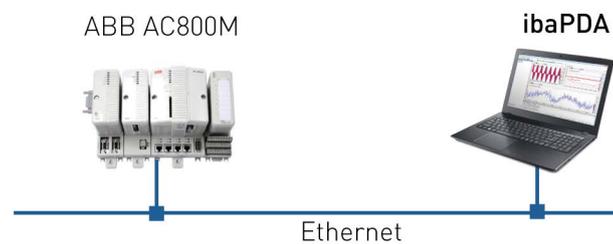
3.1 General Information

The ABB-Xplorer interface is suitable for measurement data acquisition with *ibaPDA* to ABB AC800M and AC800PEC controllers.

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

In the ABB AC800 controller, no programming work is necessary for establishing a connection between *ibaPDA* and a controller with defined IP address and for sending the respective signals. For transmitting measurement data, no additional software from ABB is necessary.

3.2 System topology



The connections to the controllers can be established via standard Ethernet connections of the computer.

No special software is needed.

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 of the AC800 controller

For the configuration of AC800 controllers, variables that are accessible from *ibaPDA* must be defined as MMS access variables in the ABB Compact Control Builder.

Note

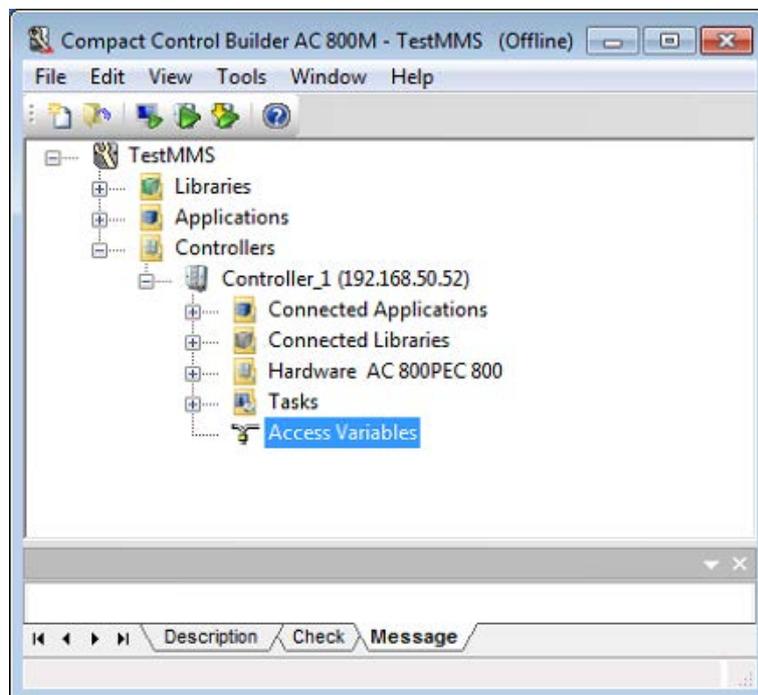


As a standard, the MMS connection to your PLC is enabled as soon as the Ethernet connection is established. If this is not the case, the connection needs to be independently configured. You can find more detailed information in the manufacturer's manual.

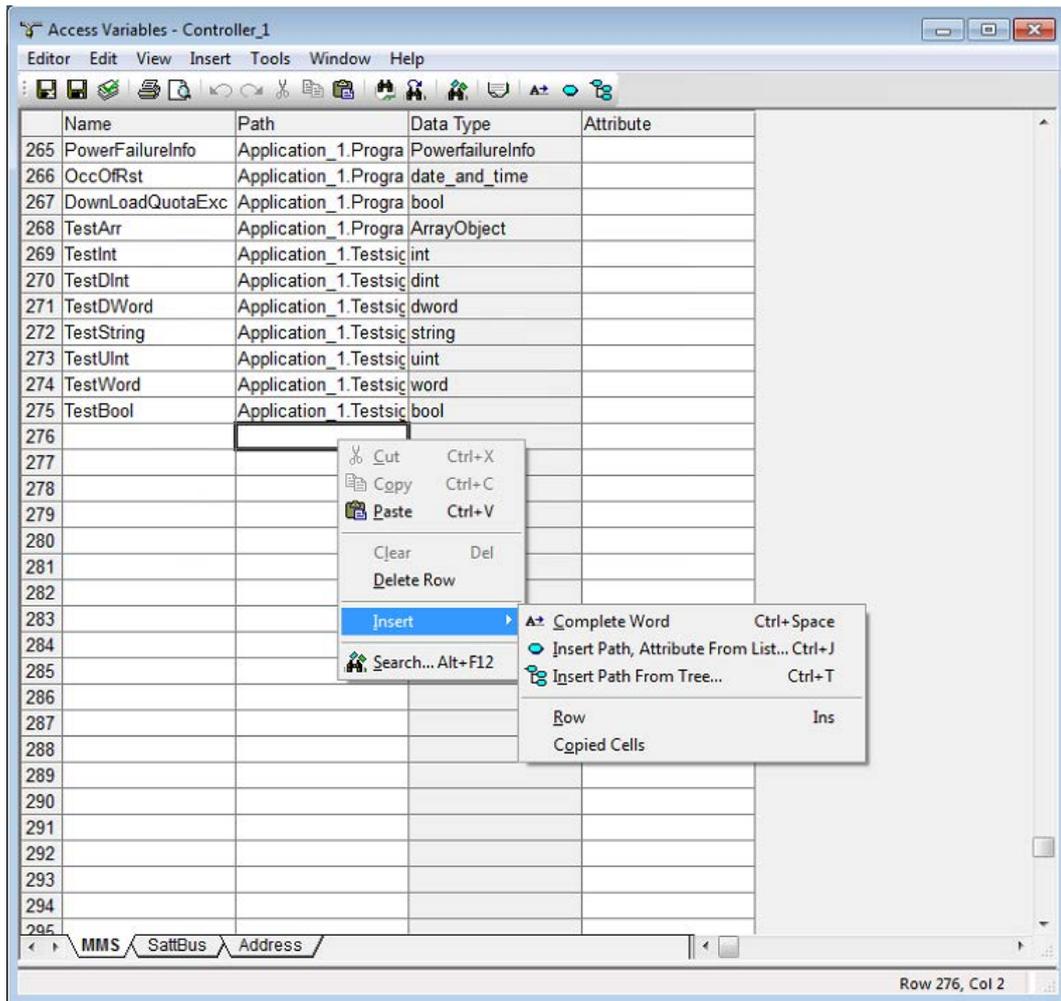
3.3.1 Access variables in the ABB Compact Control Builder

The following steps are required to edit the access variables in the ABB Compact Control Builder:

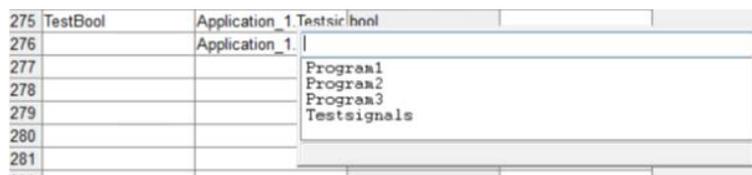
1. Open the controller in use and select the access variables.



2. Now select the MMS tab.



3. Define a name for the access variable and select the corresponding path to the variable within the program. To do this, either type the path name and enable the auto-fill function (Complete Word, Insert Path from List) or use the variable browser (Insert Path From Tree).



The screenshot shows the 'Access Variables - Controller_1*' window. It features a menu bar (Editor, Edit, View, Insert, Tools, Window, Help) and a toolbar with various icons. The main area is divided into a table and a tree view.

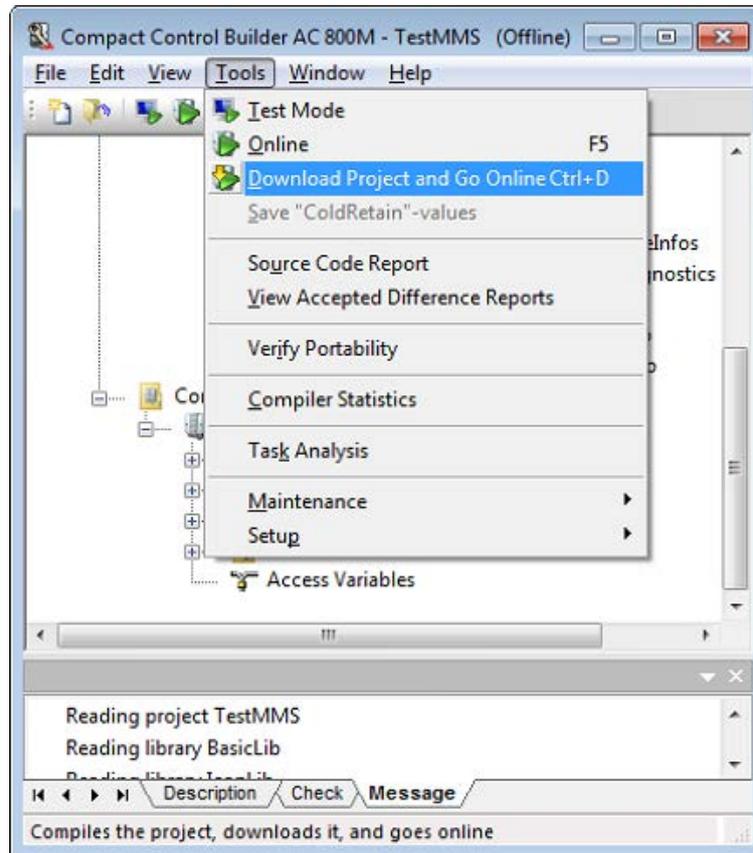
Name	Path	Data Type	Attribute
265	PowerFailureInfo	Application_1.Progra	PowerfailureInfo
266	OccOfRst	Application_1.Progra	date_and_time
267	DownLoadQuotaExc	Application_1.Progra	bool
268	TestArr	Application_1.Progra	ArrayObject
269	TestInt	Application_1.Testsig	int
270	TestDInt	Application_1.Testsig	dint
271	TestDWord	Application_1.Testsig	dword
272	TestString	Application_1.Testsig	string
273	TestUInt	Application_1.Testsig	uint
274	TestWord	Application_1.Testsig	word
275	TestBool	Application_1.Testsig	bool
276		Application_1.	not found
277			
278			
279			
280			
281			
282			
283			
284			
285			
286			
287			
288			
289			
290			
291			
292			
293			
294			
295			

The tree view on the right shows the following structure:

- Application_1
 - Testsignals
 - CTU_1
 - SinGen_1
 - Bool000
 - Bool001
 - Bool002
 - Bool003
 - Bool004
 - Bool005
 - Bool006
 - Bool007
 - Real000
 - Real001
 - Real002
 - Real003
 - Real004
 - Real005
 - Real006
 - Real007
 - Real008

At the bottom, there are tabs for 'MMS', 'SattBus', and 'dint'. The status bar at the bottom right indicates 'Row 276, Col 2'.

4. Once all variables are defined, select “Download Project and Go Online” in the main menu to refresh the program in the PLC.



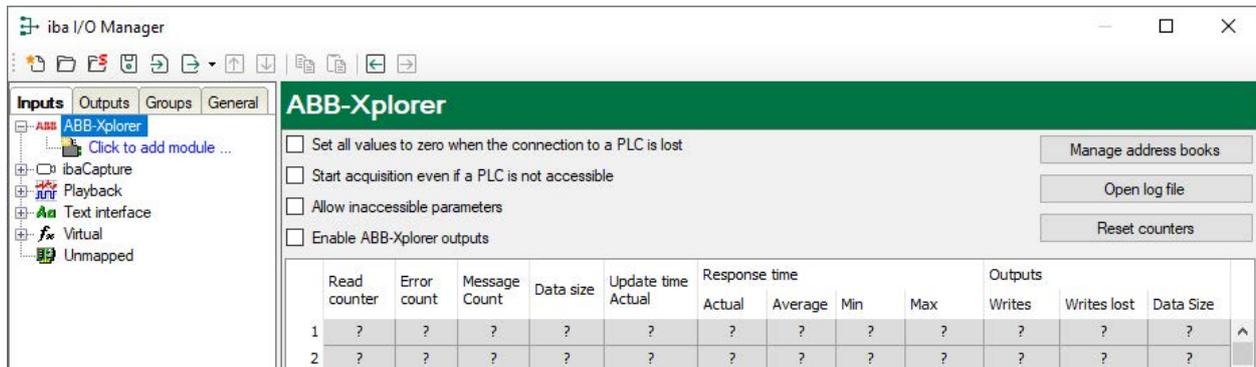
5. Now use the <Test> button in the *Connection* tab in *ibaPDA* to refresh the address book. Select the new variables in the ABB MMS symbol browser and insert them into your *ibaPDA* project.

3.4 Configuration and engineering ibaPDA

Subsequently, the engineering for *ibaPDA* is described. If all system requirements are met, the interface „ABB-Xplorer“ is displayed in the signal tree.

3.4.1 General interface settings

The interface has the following features and configuration options:



Set all values to zero when the connection to a PLC is lost

If enabled, all measured values of the PLC are set to zero as soon as the connection is lost. If this option is disabled, *ibaPDA* will keep the last valid measured value in memory at the time the connection was lost.

Start acquisition even if a PLC is not accessible

If this option is enabled, the acquisition will start even if an AC800 controller is not accessible. Instead of an error, a warning is indicated in the validation dialog. If the system has been started without a connection to the AC800 controller, *ibaPDA* will try to connect to the PLC at regular intervals. The measured values stay at zero as long as the PLC is disconnected.

Allow inaccessible parameters

Enable this option to start the acquisition even if no parameters are accessible. The inaccessible parameters are output as warnings in the validation dialog instead of errors. This can only occur if the address book is not up-to-date. If this option is not enabled and inaccessible parameters are present, then the acquisition will not start.

Enable ABB-Xplorer outputs

See [↗ Use of the output module, page 19](#)

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, you will find an error counter here for the individual connections during the acquisition.

See here [↗ Connection table, page 26](#).

<Manage address books>

Clicking on the <Manage address books> button takes you to the address book management of *ibaPDA*.

The table shows a list of all of the address books currently present in the system with IP address of the PLC from which the address book was created, as well as the date of creation, size and modules that were configured for the respective CPU. Use the <Delete selected address books> button to delete selected address books.

<Open log file>

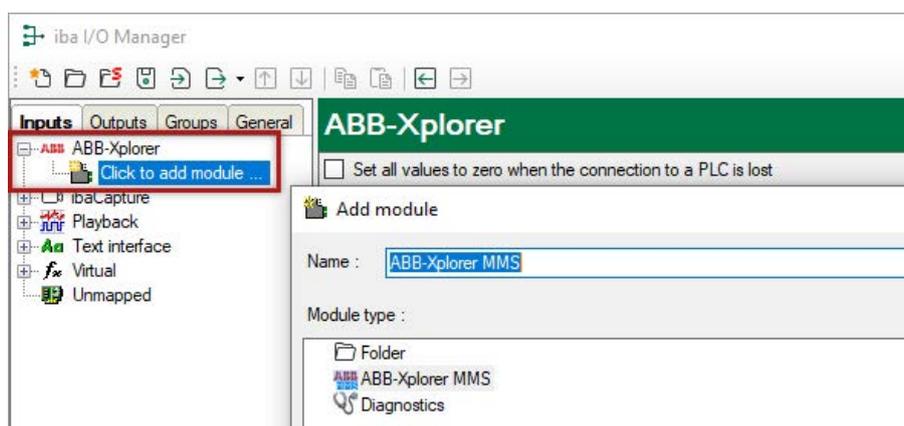
If connections to AC800 controllers have been established, all connection-specific actions are logged in a text file. With this button, you can open and have a look at this file. In the file system on the hard disc, you will find the log file in the program path of the *ibaPDA* server (... \Programs\iba\ibaPDA\Server\Log\). The file name of the current log file is [AbbXplorerLog.txt](#); the name of the archived log files is [AbbXplorerLog_yyy_mm_dd_hh_mm_ss.txt](#).

<Reset counters>

To reset the calculated times and error counters in the table to zero, click on the <Reset counters> button.

3.4.2 Adding a module

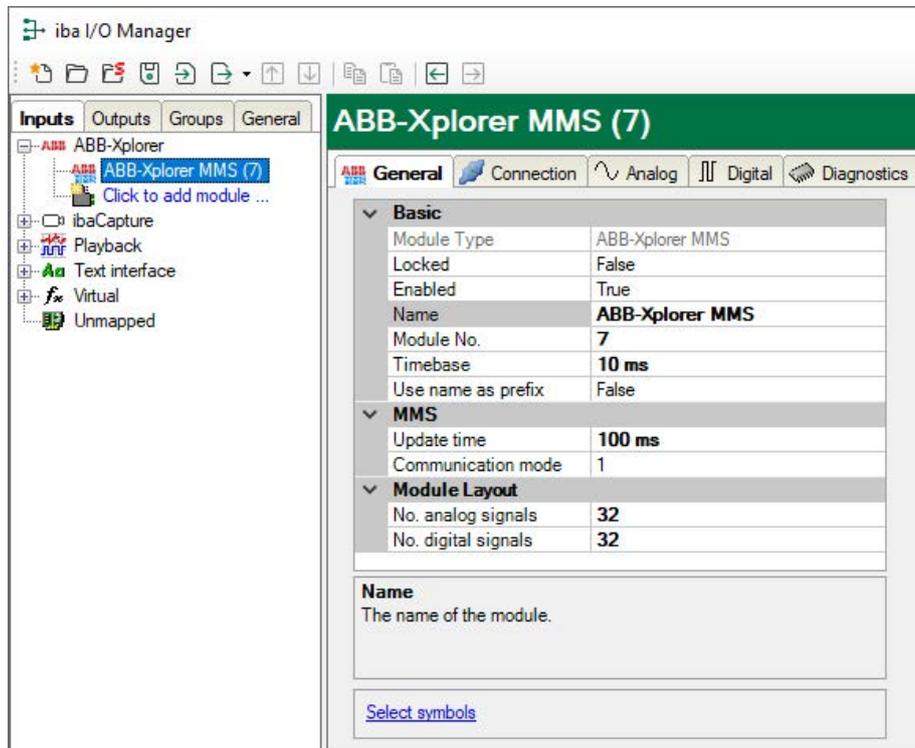
Add a module by clicking below the interface.



Select the desired module type and click <OK>.

3.4.3 General module settings

To configure a module, select it in the tree structure and configure the settings in the following dialog:



Basic settings

Module Type (information only)

Indicates the type of the current module.

Locked

A module can be locked to avoid unintentional or unauthorized changing of the module settings.

Enabled

Disabled modules are excluded from signal acquisition.

Name

The plain text name should be entered here as the module designation.

Module No.

Internal reference number of the module. This number determines the order of the modules in the signal tree of *ibaPDA* client and *ibaAnalyzer*.

Timebase

All signals of the module will be sampled on this time base.

Use name as prefix

Puts the module name in front of the signal names.

MMS

Update time

The update time is the time in ms between two reading operations. You can set the value. The update time specifies how quickly the data is called up from the controller.

Communication mode

The communication mode in use is displayed here: Mode 1 for older systems; Mode 2 skips the checksum for newer systems.

Module Layout

Number of analog and digital signals

Presets the number of configurable analog and digital signals in the signal tables. The preset value is 32 in each case. You can change the number. The maximum value is 1000.

<Select symbols>

Clicking on the <Select symbols> link opens the ABB MMS symbol browser. With the symbol browser, you can easily add analog or digital signals to the ABB MMS module by double clicking on any variable or selecting several variables and then clicking <Add>.

A connection to the target PLC must have been configured and tested first so that variables are visible in the symbol browser.

See chapter [➤ Signal configuration](#), page 20

3.4.4 Connection settings

In the *Connection* tab of the module settings, you can configure the connection to the PLC, generate address books and test the connection.

The screenshot shows the 'Connection' tab of the ABB-Xplorer MMS (7) settings window. The window has a green title bar and a tabbed interface with 'General', 'Connection', 'Analog', 'Digital', and 'Diagnostics' tabs. The 'Connection' tab is active. The settings include:

- Address:** 127.0.0.1
- Timeout (s):** 5
- Maximum number of objects to read in a single command:** 64
- Enable direct access** (Up to date Control Builder project files are needed for this.)
- Base path to Control Builder projects:** C:\ABB Industrial IT Data\Engineer IT Data\Compact Control Builder AC 800M\Projects
- Username:** (empty field)
- Password:** (empty field)

Buttons on the right side include 'Test', 'Create address book', 'Create address book offline', and 'Test path'.

Connection

Address

IP address of the PLC

Timeout (s)

Time span after which a link attempt is aborted.

Maximum number of objects to be read in a single command

ibaPDA tries to group the requested elements in as few requests as possible. If the capacity utilization of the PLC-CPU is too high, this leads to smaller packets that can be processed without a timeout due to the limitation of the parameters.

Enable direct access (Up to date Control Builder project files are needed for this.)

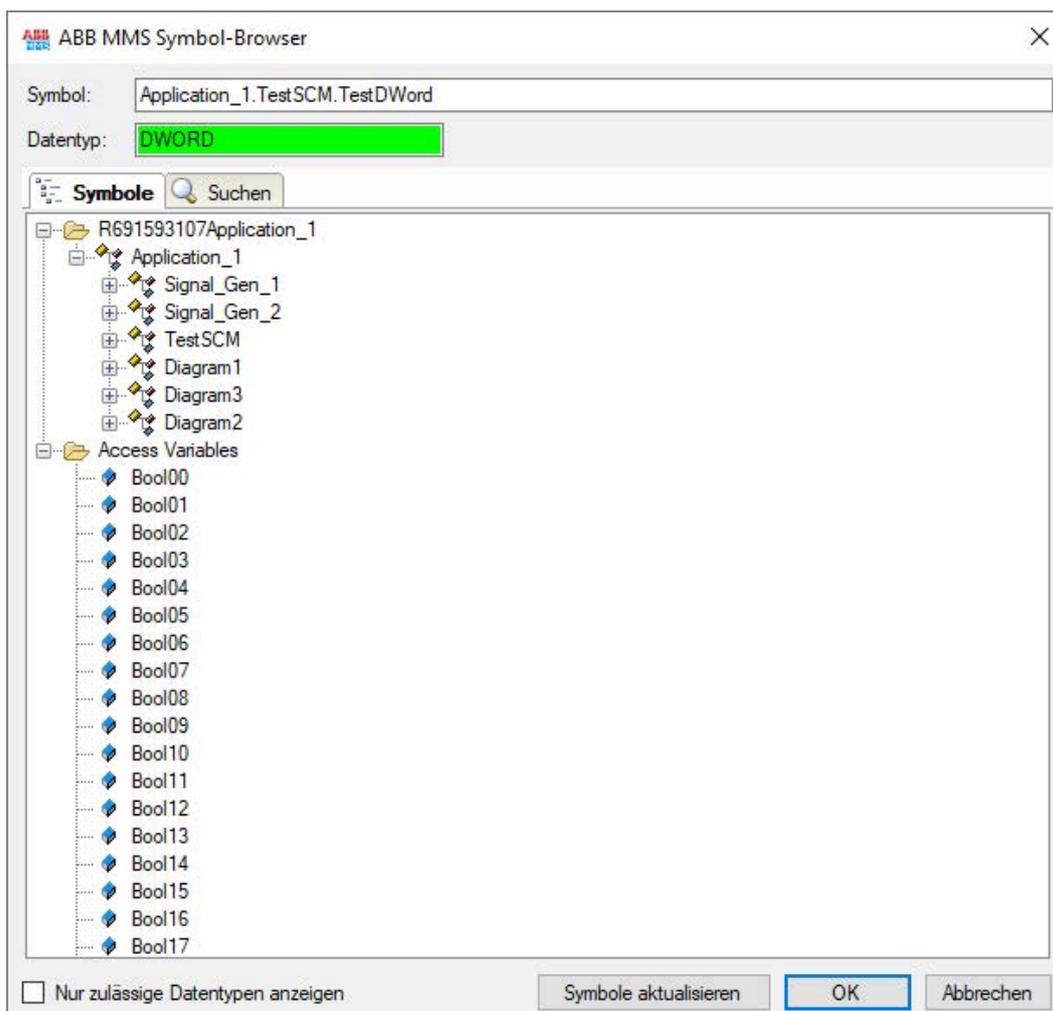
In order to use the ABB Compact Control Builder project files to access internal variables of the PLC, additional parameters must be configured.

On the PC where ABB Compact Control Builder is installed, this path is normally:

"C:\ABB Industrial IT Data\Engineer IT Data\Compact Control Builder AC 800M\Projects".

This project folder can be used either directly when *ibaPDA* is running on the same computer or by sharing the folder over the network. The user name and password for access sharing can also be entered in this dialog. If the folders are copied to another location, the user must ensure that these copies are updated in *ibaPDA* as soon as changes are made in the ABB Compact Control Builder and in the *ibaPDA* address book. Otherwise, access to the variables is denied and a message is displayed that the address book needs to be updated.

The selected folder must be the base path that contains all projects. The corresponding project folder is then automatically selected based on the information from the PLC.



After creating the new address book, an additional branch appears in the symbol browser containing the variables for the imported project, including the named structure elements. The symbol created from this contains, in contrast to the direct name of the access variable, the complete variable path.

Name	Unit	Gain	Offset	Symbol	DataType	Active
0 TestDWord		1	0	R691593107Application_1\Application1.TestSCM_TestDWord	WORD	<input checked="" type="checkbox"/>
1 TestDWordNormal		1	0	TestDWordNormal	DINT	<input checked="" type="checkbox"/>
2		1	0		INT	<input type="checkbox"/>
3		1	0		INT	<input type="checkbox"/>

<Test>

A connection test to the PLC is executed and available diagnostic data is output.

<Create address book>

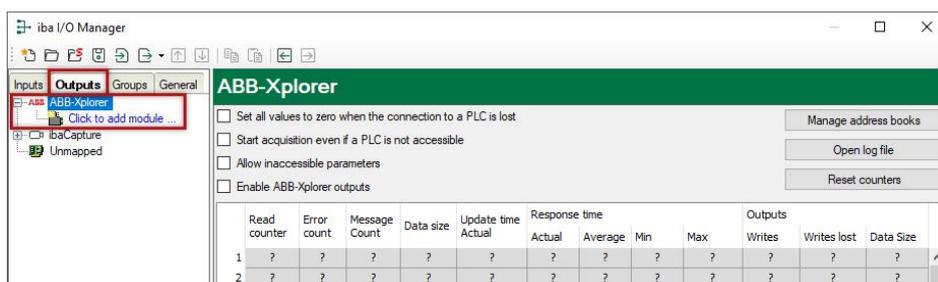
By clicking on this button, *ibaPDA* establishes a connection to the PLC, reads the address book and saves it on the *ibaPDA* server. In the process, all previous address books relating to the selected IP address are overwritten.

<Create address book offline>

Clicking this button will generate the address book from the project.

3.4.5 Configuration ibaPDA output module

If all system requirements are met, the interface "ABB-Xplorer" is displayed in the tree structure of the *ibaPDA*-I/O-Manager, tab *Outputs*.



3.4.5.1 General module settings

The parameters are almost identical to those of the input modules, see ↗ *General module settings*, page 15

Consider the differences in the settings in contrast to the input modules:

Calculation timebase

Time base (in ms) used for the calculation of the PLC output values.

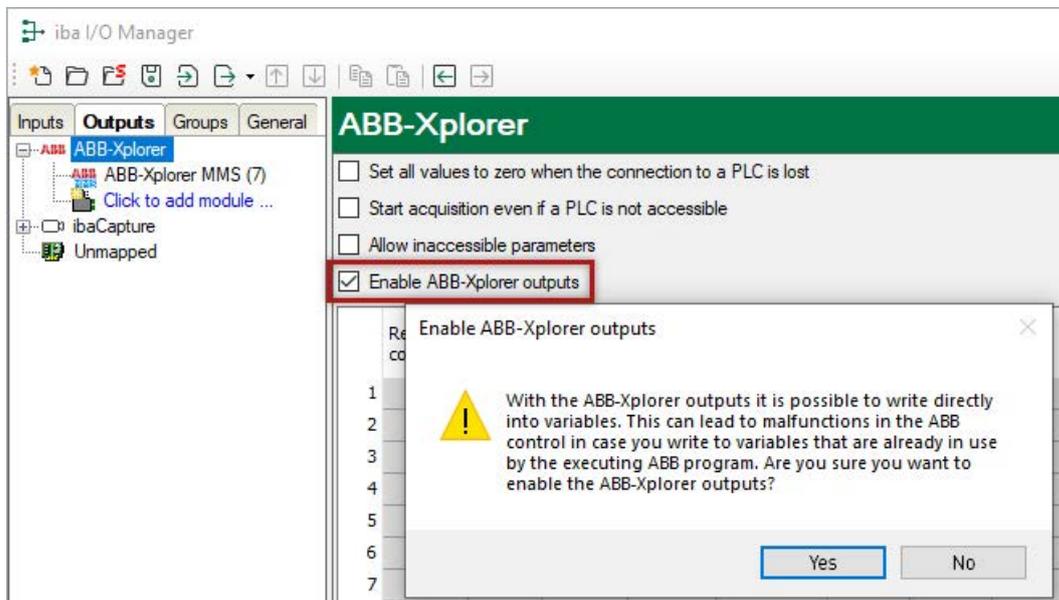
Write mode

There are 3 options, with the update time always determining the fastest write rate:

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

3.4.5.2 Use of the output module

In the *Outputs* tab the ABB-Xplorer module can be used to send data to the PLC. For this activate the checkbox *Enable ABB-Xplorer outputs*.

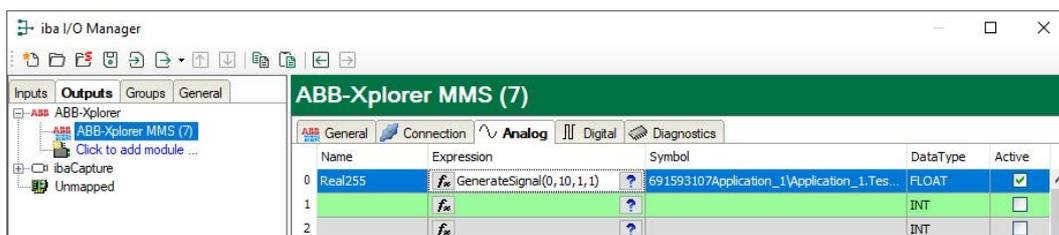


Note



A warning indicates that the ABB system does not contain any access control and can therefore be written to all variables. This can lead to malfunctions of the program running on the PLC. The user may only select variables that are intended for this purpose by *ibaPDA*.

In order to define output signals, the symbol browser entering the access path, a symbol name based on the path, and the data type associated with that symbol, can be used. Now standard expressions can be used to define values.



3.4.6 Signal configuration

The selection of the signals to be measured is carried out in the I/O Manager by means of the symbol name supported by the address book browser.

The configuration of the signals to be measured is made in the signal tables in the *Analog* or *Digital* tab.

The length of the signal tables or the number of signals per table is defined for each table in the General Module Settings, module structure. See chapter ↗ *General module settings*, page 15

Note



Observe the maximum permissible signal number based on your license.

Note

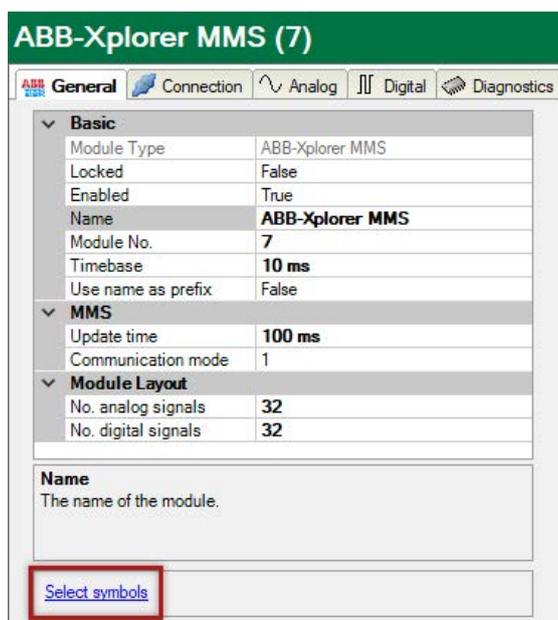


Take into consideration that the number of signals, which are read by a CPU, influences the minimum achievable read cycle. The more signals recorded, the slower the reachable reading cycle.

Selection of measuring signals

You have two options to select the measuring signals:

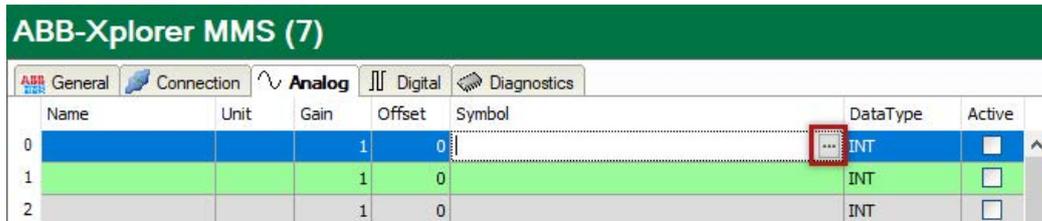
1. Click on the <Select symbols> hyperlink in the *General* tab of the module.



Use the mouse to click on the link and open the ABB MMS symbol browser:

- Both the analog values as well as the digital values can be seen.
- Double click to apply the selected variables in the current row of the analog or digital table. Or select several variables and click <Add>.
- Close the browser by pressing <Close>.

2. In the *Analog* or *Digital* tab, click on the small browser button (...) in a field of the symbol column.



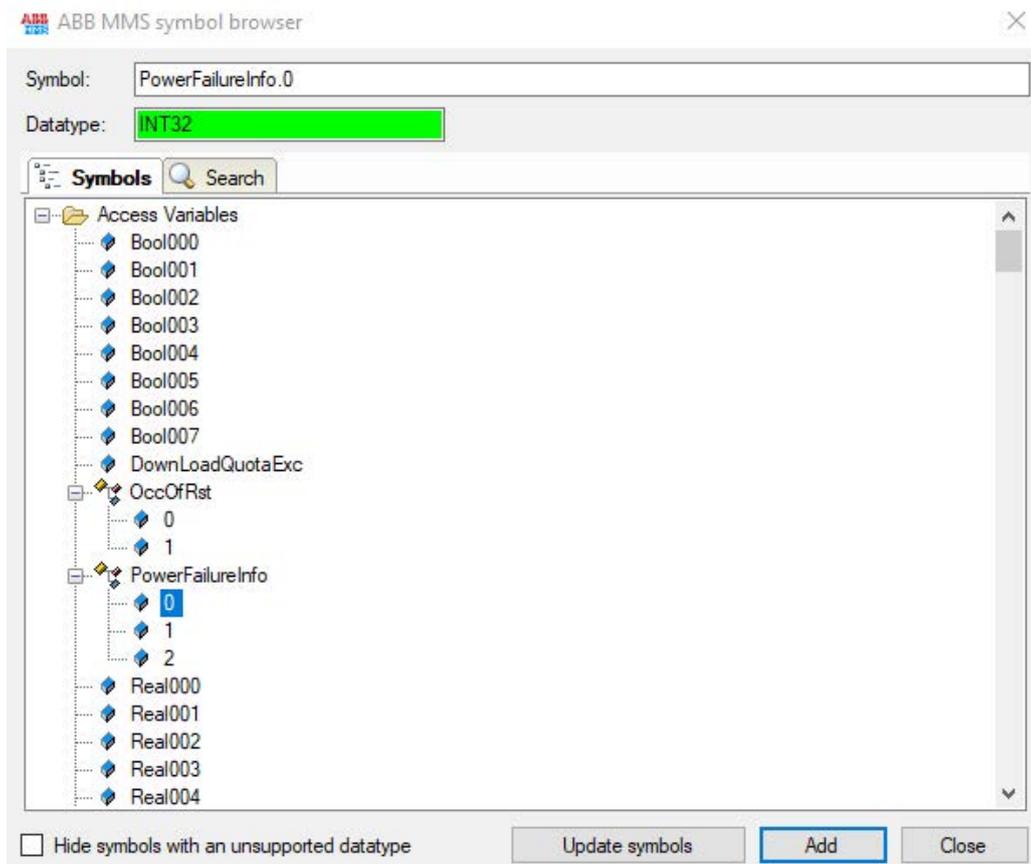
A click on the icon opens the ABB MMS symbol browser:

- Both the analog values as well as the digital values can be seen.
- Double click or click <Add> to apply the selected variable in the signal table and the browser is closed.

Meaning of the colors:

Green	The operand is valid and can be included in the module setting with OK.
Yellow	The operand has a data type that does not match the selected column, e.g., if you have selected a Boolean variable as an analog value or an integer value as a digital signal.

ABB MMS symbol browser:



A connection to the target PLC must have been configured and tested first so that variables are visible in the symbol browser.

Only variables can be used that have been defined in the ABB Compact Control Builder as “ABB MMS access variables.” See [Access variables in the ABB Compact Control Builder](#), page 9

On the *Search* tab, you can search variables by name. The search result tree works in the same way as the complete variable tree.

By selecting the checkbox “Hide symbols with an unsupported datatype” you can hide all not supported variables.

3.4.7 Module diagnostics

While measuring, the real values of the analog and digital signals can be viewed in the *Diagnostics* tab of the respective module.

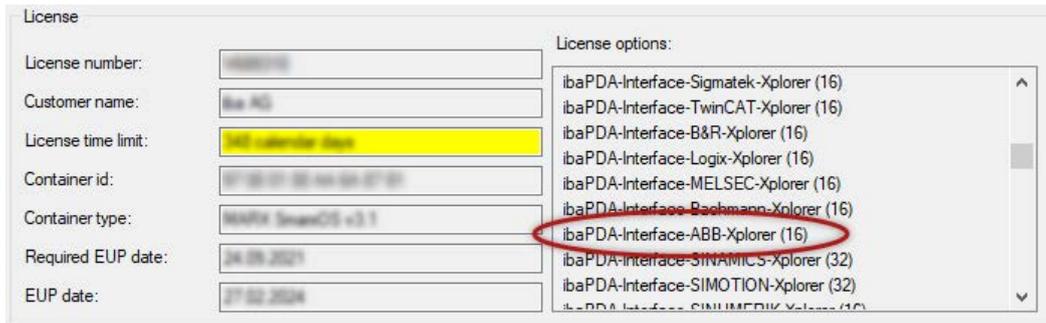
The screenshot shows the 'Diagnostics' tab in the software interface. It contains two sub-tabs: 'Analog values' (selected) and 'Digital values'. Below the sub-tabs is a table with the following data:

	Name	Symbol	Datatype	Value	Unit
11	Real003	Real003	REAL	0,552276	
12	Real004	Real004	REAL	-0,389983	
13	Real005	Real005	REAL	-0,695827	
14	Real006	Real006	REAL	-0,0316494	
15	Real007	Real007	REAL	-0,997997	
16	Real008	Real008	REAL	0,991995	
17	Real009	Real009	REAL	0,968106	
18	Real010	Real010	REAL	10	
19	Real011	Real011	REAL	11	
20	Real012	Real012	REAL	12	
21	Real013	Real013	REAL	13	
22	TestInt	TestInt	DINT	-32	
23	TestString	TestString	STRING[255]	Hello!	

4 Diagnostics

4.1 License

If the "ABB Xplorer" interface is not displayed in the signal tree, you can either check in *ibaPDA* under *General - Settings - License* in the I/O manager or in the *ibaPDA* service status application to see whether your license "Interface - ABB Xplorer" has been properly recognized. The number of licensed connections is indicated in brackets.



4.2 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.

The log file can be opened 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 will find the log files in the program path of the *ibaPDA* server (...\\Programs\\iba\\ibaPDA\\Server\\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)

4.3 Connection diagnostics with PING

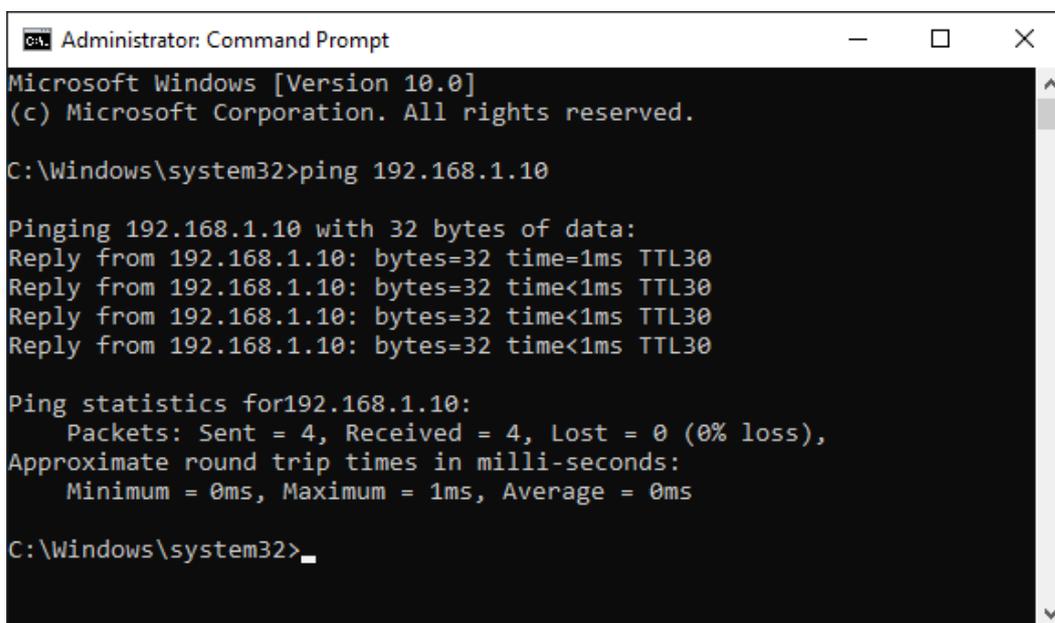
PING is a system command with which you can check if a certain communication partner can be reached in an IP network.

Open a Windows command prompt.



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

With an existing connection you receive several replies.

A screenshot of the Windows Command Prompt window. The window title is 'Administrator: Command Prompt'. The text in the window shows the execution of the 'ping' command for the IP address 192.168.1.10. The output shows four successful replies with a time of less than 1ms and a TTL of 30. The ping statistics show 4 packets sent, 4 received, and 0% loss.

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

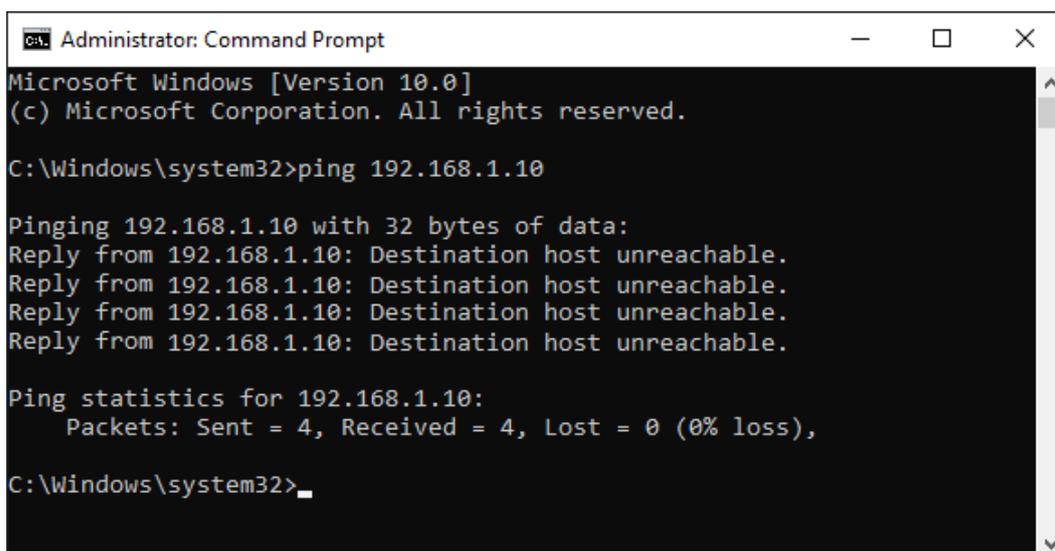
C:\Windows\system32>ping 192.168.1.10

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

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.

A screenshot of the Windows Command Prompt window. The window title is 'Administrator: Command Prompt'. The text in the window shows the execution of the 'ping' command for the IP address 192.168.1.10. The output shows four failed replies with the message 'Destination host unreachable'. The ping statistics show 4 packets sent, 4 received, and 0% loss.

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

C:\Windows\system32>ping 192.168.1.10

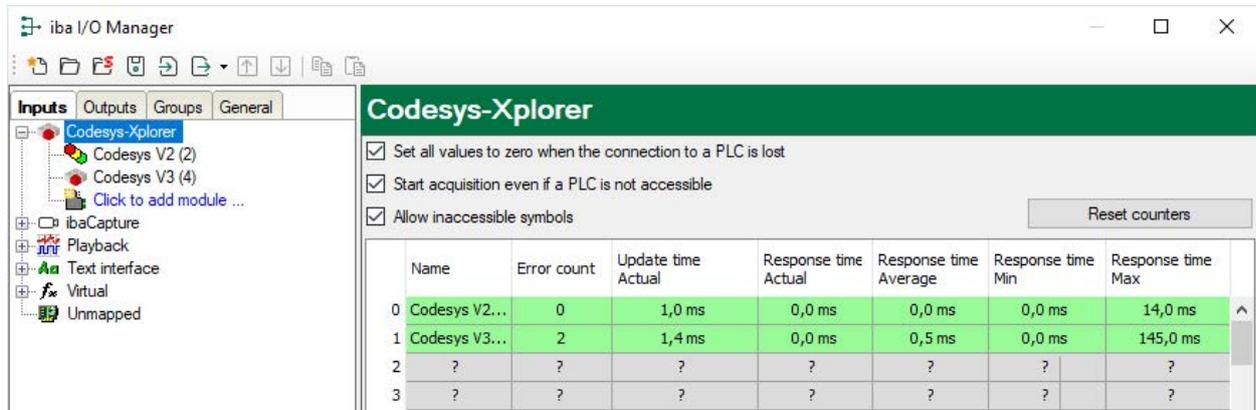
Pinging 192.168.1.10 with 32 bytes of data:
Reply from 192.168.1.10: Destination host unreachable.

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

C:\Windows\system32>
```

4.4 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 counters> 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.

Table 1: Meaning of background colors

4.5 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.

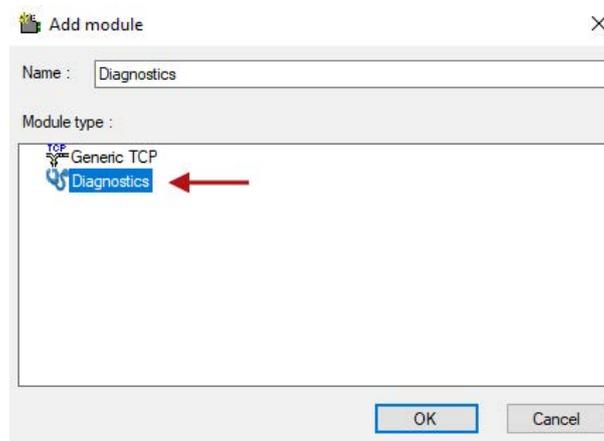
A diagnostic module is always assigned to a data acquisition module of the same interface and supplies its connection information. By using a diagnostic module you can record and analyze the diagnostic information continuously in the *ibaPDA* system.

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:

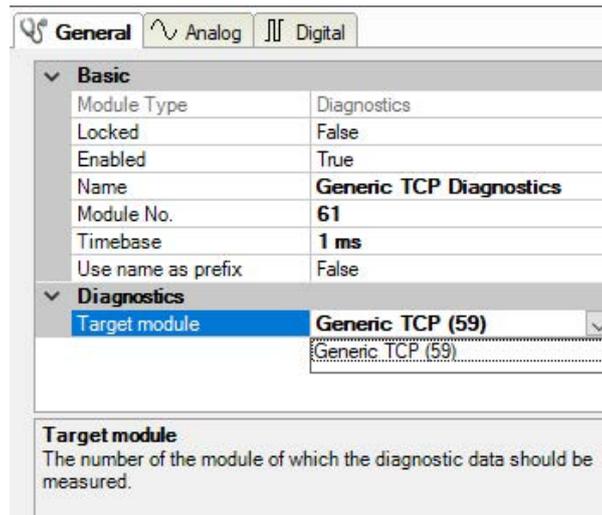
- A notification can be generated, whenever the error counter of a communication connection exceeds a certain value or the connection gets lost.
- In case of a disturbance, the current response times in the telegram traffic may be documented in an incident report.
- The connection status can be visualized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

General Analog Digital						
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"/>	
11 Time between data (max)	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.

General Analog Digital		
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
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

Signal name	Description
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
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)

Signal name	Description
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

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 license number or the CodeMeter container number (WIBU dongle). 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.