



ibaPDA-Interface-Generic-UDP

Data Interface Generic UDP

Manual
Issue 1.9

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
1.9	02-2023	Calculation timebase corrected, license terms	iip, mm	8.2.0

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.

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	Data Interface Generic-UDP	7
3.1	General information	7
3.2	Configuration of the controller.....	8
3.3	Configuration and engineering ibaPDA.....	9
3.3.1	General settings.....	9
3.3.2	General interface settings.....	10
3.3.3	Adding a module.....	10
3.3.4	General module settings.....	12
3.3.5	Signal configuration	14
3.3.6	Module diagnostics.....	15
3.4	Configuration of the ibaPDA output modules	16
3.4.1	General module settings ibaPDA output modules.....	16
3.4.2	Signal configuration	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	Checking the connection for messages received.....	22
4.6	Checking the connection for sent messages.....	23
5	Support and contact.....	24

1 About this Manual

This document describes the function and application of the software interface

ibaPDA-Interface-Generic-UDP

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 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 of *ibaPDA-Interface-Generic-UDP* the following basic knowledge is required and/or useful:

- Windows operating system
- Basic knowledge of *ibaPDA*
- Knowledge of configuration and operation of the relevant measuring device/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

The following system requirements are necessary for the use of the data interface *Generic UDP*:

- *ibaPDA* v8.1.0 or higher
- Base license for *ibaPDA* + license for *ibaPDA-Interface-Generic-UDP*
- Network connection 10/100 Mbits

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

Note



It is highly recommended to operate the Generic-UDP communication on a separate network segment in order to exclude a mutual influence by other network components.

License information

Order no.	Product designation	Description
31.001075	ibaPDA-Interface-Generic-UDP	Extension license for an <i>ibaPDA</i> system providing an additional Generic-UDP interface Number of connections: 64
31.101075	one-step-up-Interface-Generic-UDP	Extension license for the extension of an existing interface by another 64 Generic-UDP connections, max. 3 permitted

3 Data Interface Generic-UDP

3.1 General information

ibaPDA-Interface-Generic-UDP can be used by any controller capable of sending UDP/IP messages.

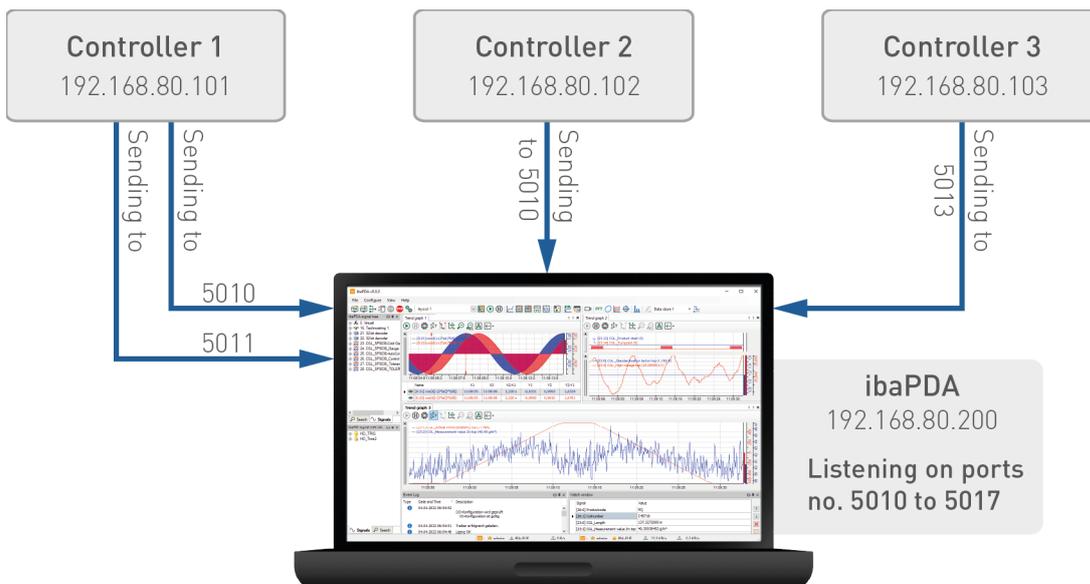
The User Datagram Protocol (UDP) is one of the core protocols of the Internet protocol suite. Using UDP, programs on networked computers can send messages to one another. These messages are also called datagrams.

As the sender does not get any confirmation that the data have been received, UDP is in contrast to TCP not connection oriented and not reliable. However, one advantage resulting from this are the higher data rates that can be transferred with UDP.

UDP is explained in chapter 6 of RFC1180 and in RFC768 (see references).

The Generic-UDP messages are IP Unicast messages sent from one or several controllers to the *ibaPDA* system using a defined port number.

The following drawing gives an overview of a possible configuration scenario where three controllers are sending UDP messages to one *ibaPDA* system.



Properties

- The messages sent over each connection must have a fixed layout.
- You can define a port range on the Generic-UDP interface in the *ibaPDA* I/O Manager. In the above example, the *ibaPDA* driver is listening on port 5010 to port 5017 for incoming UDP messages.
- Each UDP connection is uniquely identified within *ibaPDA* by the destination port number and the source IP address.

Thus, *ibaPDA* can receive data from different controllers, which use the same destination port. It is also possible to send messages from one controller to *ibaPDA* over different ports.

This is shown clearly in the above example: Controller 1 and 2 use the same port 5010, but have different IP addresses. Controller 1 sends several messages and uses different ports for sending (5010 and 5011).

ibaPDA specific limitations

- The maximum length of the UDP message is limited to 4096 bytes.
- The number of the supported connections in *ibaPDA* depends on the Generic-UDP license (64, 128, 192 or 256).

The following controllers apply:

- Any system capable of sending UDP/IP messages

Other documentation



Reference to additional documentation or further literature.

- *ibaPDA* manual (<https://www.iba-ag.com/en/downloads>)
 - TCP/IP Tutorial, RFC1180 (<ftp://ftp.ripe.net/rfc/rfc1180.txt>)
 - User Datagram Protocol, RFC768 (<ftp://ftp.ripe.net/rfc/rfc768.txt>)
-

3.2 Configuration of the controller

For the controller that is intended to send data to *ibaPDA*, you have to set up a UDP communication. The destination address is the IP address of the *ibaPDA* server. You have to declare the destination port within the defined port range in *ibaPDA* (see ↗ *General interface settings*, page 10).

The default destination port range defined in *ibaPDA* is 5010 – 5017. The maximum supported length of the UDP data is 4096 bytes. Longer messages will be rejected by *ibaPDA*.

The controller should cyclically send a message with a fixed layout to the *ibaPDA* system using a fixed destination port. If more than one message with a different content is needed, the same destination IP address must be used, but to another destination port within the defined range.

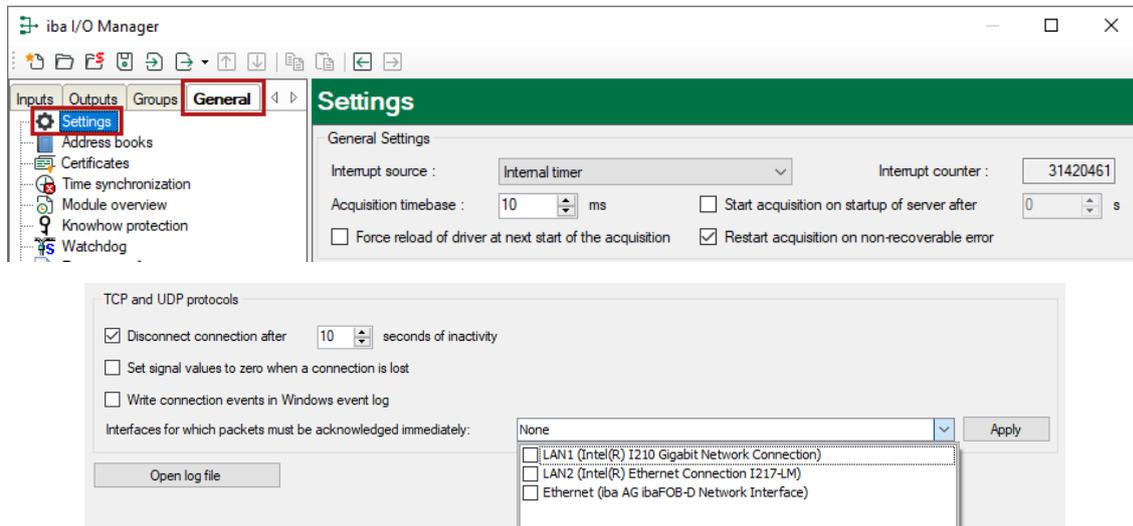
Each UDP connection is uniquely identified on the *ibaPDA* side by the "Destination port number" and "Source IP address". This means that one controller can send several messages to the same *ibaPDA* even using a destination port already used by other controllers.

3.3 Configuration and engineering ibaPDA

Subsequently, the engineering for *ibaPDA* is described. If all system requirements are met, *ibaPDA* offers the *Generic UDP* interface in the signal tree of the I/O Manager.

3.3.1 General settings

The "Alive timeout" is configured jointly for all TCP/IP and UDP protocols supported by *ibaPDA*.



Disconnect connection after ... seconds of inactivity

Behavior and timeout duration can be specified.

Set signal values to zero when a connection is lost

If this option is disabled, the value read last will be kept.

Write connection events in Windows event log

Current events are logged in Windows.

Interfaces for which packets must be acknowledged immediately

Selection of required interfaces.

3.3.2 General interface settings

The interface provides the following functions and configuration options.

	Address	Port	Message counter	Packet size Actual	Incorrect message type	Multicast join error	Time Actual	Time Min	Time Max
0	192.168.80.101	5010	2541	206	0	-	100,2 ms	99,3 ms	101,0 ms
1	192.168.80.103	5012	1131	58	0	0	100,6 ms	98,9 ms	101,1 ms
2	192.168.80.101	5011	0	0	0	-	0,0 ms	0,0 ms	0,0 ms

UDP port list

Ports on which *ibaPDA* is waiting for incoming UDP messages. You can enter the ports as port range, as list of individual ports or as a combination of both. If you enter a range, please separate it by a hyphen. Do not separate successive port numbers by commas. The default setting is the range 5010-5017. The identical port number has to be used in the controller (see [➤ Configuration of the controller](#), page 8).

Allow ports through firewall

When installing *ibaPDA*, the default port numbers of the used protocols are automatically entered in the firewall. If you change the port number or enable the interface subsequently, you have to enable this port in the firewall with this button.

Counter for invalid messages

see [➤ Checking the connection for messages received](#), page 22.

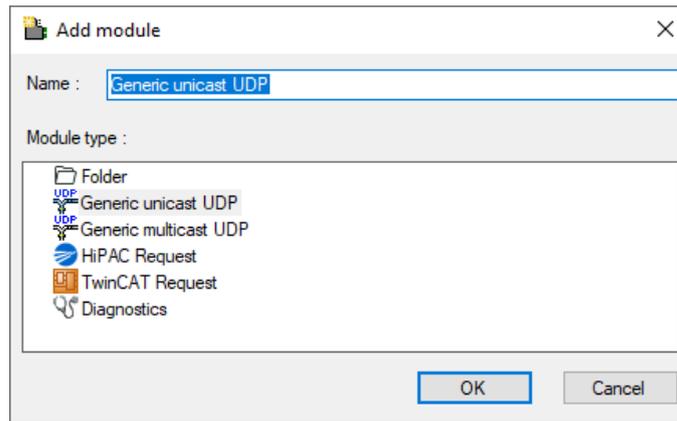
Connection table

see [➤ Checking the connection for messages received](#), page 22.

3.3.3 Adding a module

Procedure

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>.



Module types

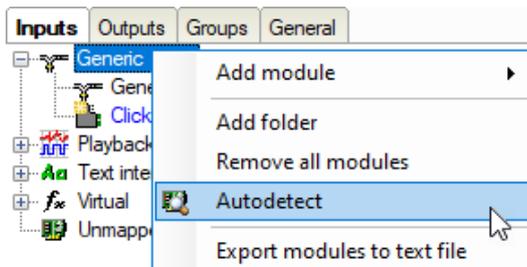
You can add the following module types to the interface:

- Generic unicast UDP: see ↗ *General module settings*, page 12
- Generic multicast UDP: see ↗ *General module settings*, page 12
- HiPAC Request: see manual *ibaPDA-Request-HiPAC*
- TwinCAT Request: see manual *ibaPDA-Request-TwinCAT*

Tip



Cyclic unicast UDP messages, that have their destination port in the UDP port range of the interface are displayed automatically in the connection table, even if no corresponding modules have been defined and *ibaPDA* was not started before. You can create modules for these links easily with a right click on the *Generic UDP* interface icon in the tree view and select *Autodetect* in the context menu.



By applying *Autodetect*, a *Generic Unicast UDP* module with 32 analog signals (data type *FLOAT*) and 32 digital signals is created for each link.

Note



The *Autodetect* functionality only works with UDP unicast messages.

3.3.4 General module settings

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

All modules have the following setting options.

Generic multicast UDP (7)	
<div style="display: flex; justify-content: space-between;"> UDP General Digital </div>	
Basic	
Module Type	Generic multicast UDP
Locked	False
Enabled	True
Name	Generic multicast UDP
Module No.	7
Timebase	10 ms
Use name as prefix	False
Text encoding	Default system locale
Advanced	
Swap analog signals	No swap
Swap digital signals	False
Connection	
Source address	127.0.0.1
Destination port	5010
Multicast IP address	239.0.0.1
Module Layout	
No. analog signals	32
No. digital signals	32
Name	
The name of the module.	

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.

Text encoding

You can select the type of text coding or the code page here for a correct interpretation and display of the received text data. Available for selection are:

- System locale (default): Coding according to the Windows system settings
- Western European (1252): 8-bit coding, including ASCII and Western European umlauts, special characters
- UTF-8: Unicode, for universal characters, including ASCII, Asian languages, etc.

Advanced**Swap analog signals, Swap digital signals**

Option to change the order of the byte evaluation

Connection**Source address, Destination port**

With these two values the connection to a certain controller can be clearly identified. The port has to be within the port range defined for the interface and has to be enabled for the firewall (see ↗ *General interface settings*, page 10).

Multicast IP address (only for module type Generic Multicast UDP)

Specify here the multicast address used by the controller to send data to *ibaPDA* using Multicast. The multicast addresses are in the range from 224.0.0.0 through 239.255.255.255.

Note

The only difference between the multicast and unicast modules is the destination IP address on side of the sending controller. For the unicast modules this is the IP address of the *ibaPDA* server PC. For the multicast modules, this is a multicast address.

There are the following limitations: The combination of source address and destination port number of a multicast module must be different than that combination of all the other unicast and multicast modules. So, you cannot have one sender sending to different multicast addresses with the same port number. You also cannot have a sender that is sending both unicast and multicast messages to the same port number.

Module Layout**No. analog signals, No. digital signals**

Define the number of configurable analog and digital signals in the signal tables. The default value is 32 for each. The Maximum value is 4096. The signal tables are adjusted accordingly. Note that the total amount of data must not exceed 4096 bytes for analog and digital signals together.

Other documentation

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

3.3.5 Signal configuration

The data to be measured are selected on the controller side by mapping the signals in the data-gram, which is cyclically sent to *ibaPDA*

Analog and Digital tab

	Name	Unit	Gain	Offset	Address	DataType	Active	Actual	+
0	length		1	0	0	INT	<input checked="" type="checkbox"/>	138	^
1	Id		1	0	2	INT	<input checked="" type="checkbox"/>	101	
2	Counter		1	0	4	INT	<input checked="" type="checkbox"/>	24388	
3	Bit vector		1	0	6	DWORD	<input checked="" type="checkbox"/>	327693	
4	Sine		1	0	10	FLOAT	<input checked="" type="checkbox"/>	0,938289	
5	Cosine		1	0	14	FLOAT	<input checked="" type="checkbox"/>	-0,345852	
6	Triangle		1	0	18	FLOAT	<input checked="" type="checkbox"/>	2194,76	

You can assign name, unit, address and data type or bit number to the analog and digital signals. Moreover, you can enable or disable the signals.

Other documentation



For a description of the columns, please see the *ibaPDA* manual.

Specific columns for Generic UDP modules:

Address

The address indicates the offset of the first byte of this value within the user data telegram. You can enter the offset and be entered as hexadecimal or decimal value by selecting the desired setting in the context menu.

Unit	Gain	Offset	Address	DataType	Active	Actual	+
					<input checked="" type="checkbox"/>	138	
					<input checked="" type="checkbox"/>	101	
					<input checked="" type="checkbox"/>	8266	
					<input checked="" type="checkbox"/>	327685	

The digital signals are addressed via the *Address* and *Bit no.* (0 – 31) columns.

DataType (analog signals only)

ibaPDA supports the following data types: BYTE, WORD, DWORD, SINT, INT, DINT, LINT, FLOAT, DOUBLE, S5 FLOAT and STRING[32].

The address range depends on the data type. Hence, after changing the data type, you possibly have to adjust the address entries.

Note

The module *Generic ... UDP* supports the acquisition and processing of strings as text signals. Therefore, you can select the data type STRING[32] in the *Analog* tab. In order to convert a text signal or to split it up into several text signals, use the *Text splitter* module under the *Virtual* interface.

Tip

You can use the automatic fill function in the columns (see *ibaPDA* manual).

3.3.6 Module diagnostics

The tables *Analog* and *Digital* of the Generic UDP modules show the telegram contents (actual values).

General Analog Digital									
	Name	Unit	Gain	Offset	Address	DataType	Active	Actual	+
0	length		1	0	0	INT	<input checked="" type="checkbox"/>	138	^
1	Id		1	0	2	INT	<input checked="" type="checkbox"/>	101	
2	Counter		1	0	4	INT	<input checked="" type="checkbox"/>	24388	
3	Bit vector		1	0	6	DWORD	<input checked="" type="checkbox"/>	327693	
4	Sine		1	0	10	FLOAT	<input checked="" type="checkbox"/>	0,938289	
5	Cosine		1	0	14	FLOAT	<input checked="" type="checkbox"/>	-0,345852	
6	Triangle		1	0	18	FLOAT	<input checked="" type="checkbox"/>	2194,76	

The following errors may occur:

- No data are displayed:
 - The telegram buffer on the controller side is not filled.
 - The connectors of the send block are connected incorrectly.
- Incorrect values are displayed:
 - The telegram buffer on the controller side is not filled correctly (offset error).
 - The byte order is set incorrectly, see ↗ *General module settings*, page 12.
- The digital signals are sorted incorrectly:
 - The byte order is set incorrectly, see ↗ *General module settings*, page 12.

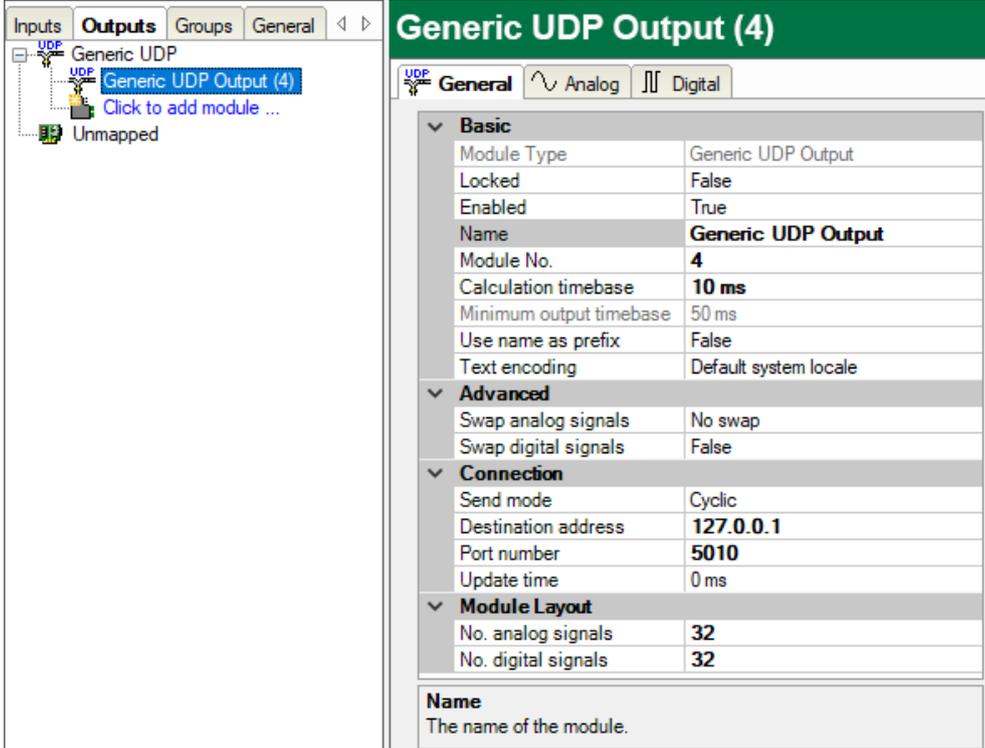
3.4 Configuration of the ibaPDA output modules

If all system requirements are met, *ibaPDA* offers the *Generic UDP* interface in the signal tree of the *Outputs* tab. There is no need to add the interface manually.

Add the output modules in the same way as input modules.

3.4.1 General module settings ibaPDA output modules

If you want to configure an output module, mark the module in the tree structure of the *Outputs* tab.



The screenshot displays the configuration window for a 'Generic UDP Output (4)' module. The interface is divided into two main sections: a tree view on the left and a configuration panel on the right.

Tree View (Left): Shows a hierarchy under 'Outputs'. A 'Generic UDP' folder contains 'Generic UDP Output (4)', which is highlighted in blue. Below it is a 'Click to add module ...' button and an 'Unmapped' folder.

Configuration Panel (Right): Titled 'Generic UDP Output (4)', it has tabs for 'General', 'Analog', and 'Digital'. The 'General' tab is active, showing a table of parameters:

Basic	
Module Type	Generic UDP Output
Locked	False
Enabled	True
Name	Generic UDP Output
Module No.	4
Calculation timebase	10 ms
Minimum output timebase	50 ms
Use name as prefix	False
Text encoding	Default system locale
Advanced	
Swap analog signals	No swap
Swap digital signals	False
Connection	
Send mode	Cyclic
Destination address	127.0.0.1
Port number	5010
Update time	0 ms
Module Layout	
No. analog signals	32
No. digital signals	32

Below the table, there is a 'Name' field with the description 'The name of the module.'

The parameters are almost identical to those of the input module, see ↗ *General module settings*, page 12.

Consider the following differences when it comes to the settings of the input modules:

Calculation timebase

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

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.

No. analog signals, No. digital signals

The length of the TCP/IP telegram is determined by the number of the signals and the assigned data types. For disabled signals, *ibaPDA* writes 0 in the telegram buffer.

The following parameters determine the telegram send cycle:

Update time

If you enter 0 or a value smaller than 50, the send cycle corresponds to the *ibaPDA* task cycle or the calculation timebase respectively (at least 50 ms).

Send mode

Defines when *ibaPDA* sends new messages. The update time always determines the fastest send rate.

- **Cyclic:** *ibaPDA* sends a message every update time.
- **On change:** *ibaPDA* sends a message each time the signal data changes.
- **On trigger:** *ibaPDA* sends a message each time a rising edge is detected on the trigger signal.

3.4.2 Signal configuration

In the *Analog* and *Digital* tab, select the signals that you want to send in a message.

Tip



If you define the output data in a virtual module and only enter here the references to these data, you can also include these data in the data recording as an option.

Analog and Digital tab

General Analog Digital							
	Name	Expression	Address	Data Type	Active	Actual	
0	ibaPDA run time (ms)	f_x T() ?		0 FLOAT	<input checked="" type="checkbox"/>	243,38	^
1	Act. Charge no.	f_x [2:14] ?		4 DINT	<input checked="" type="checkbox"/>	4534	
2	FobD-Status Link 0	f_x FobDLinkStatus(0, 0) ?		8 WORD	<input checked="" type="checkbox"/>	0	

Other documentation



For a description of the columns, please see the *ibaPDA* manual.

Expression

In the *Expression* column, define the output signals in a similar way as the virtual signals. You can enter simple expressions or references to existing signals directly in the tables. You can also open the Expression editor via the button <fx>. You can analyze an incorrect expression via the button <?>.

Address

The address indicates the offset of the first byte of this value within the sent message. You can enter the offset and be entered as hexadecimal or decimal value by selecting the desired setting in the context menu.

Unit	Gain	Offset	Address	DataType	Active	Actual	↕
					<input checked="" type="checkbox"/>	138	
					<input checked="" type="checkbox"/>	101	
					<input checked="" type="checkbox"/>	8266	
					<input checked="" type="checkbox"/>	327685	

The digital signals are addressed via the *Address* and *Bit no.* (0 – 15) columns.

DataType

ibaPDA supports the following data types: BYTE, WORD, DWORD, SINT, INT, DINT, FLOAT, DOUBLE, and STRING[32].

The address range depends on the data type. Hence, after changing the data type, you possibly have to adjust the address entries.

Active

For disabled signals, *ibaPDA* writes 0 in the telegram buffer.

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:	3-4-10170	ibaPDA-Interface-Codesys-Xplorer (16)
Customer name:	ibm:ibm:Settings	
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.

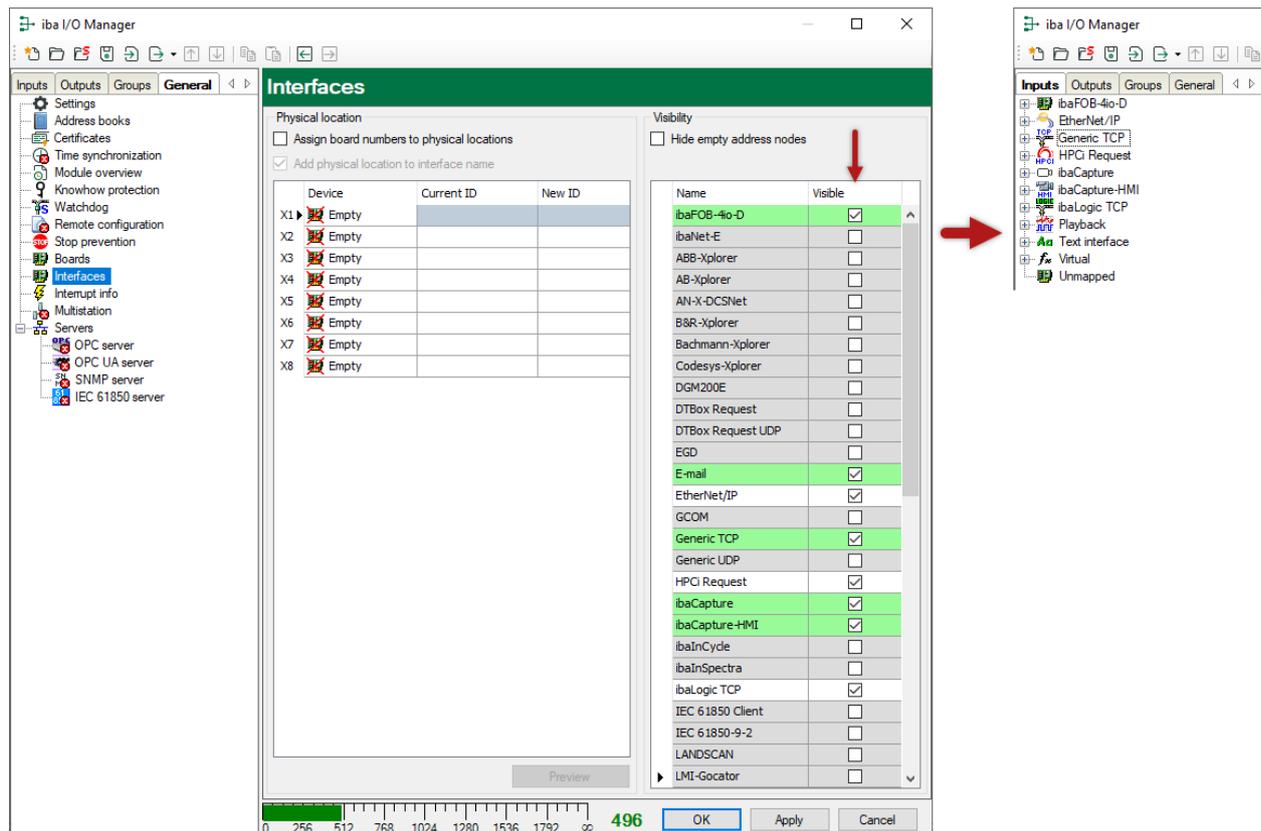
Visibility

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.

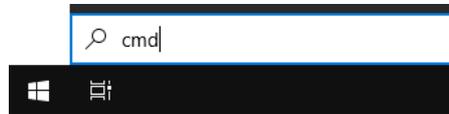
Examples:

- `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 Checking the connection for messages received

After the configuration has been accepted, *ibaPDA* displays the following data for the interface:

The screenshot shows the 'Generic UDP' configuration window in the 'iba I/O Manager'. The 'Connections' tab is active, displaying the UDP port list as '5010-5017'. Below this, there are two input fields for 'Messages received with invalid length' and 'Messages received that can't be mapped', both showing '0'. A 'Reset statistics' button is also present. At the bottom, a table displays connection statistics for three entries:

	Address	Port	Message counter	Packet size Actual	Incorrect message type	Multicast join error	Time Actual	Time Min	Time Max
0	192.168.80.101	5010	2541	206	0	-	100,2 ms	99,3 ms	101,0 ms
1	192.168.80.103	5012	1131	58	0	0	100,6 ms	98,9 ms	101,1 ms
2	192.168.80.101	5011	0	0	0	-	0,0 ms	0,0 ms	0,0 ms

Message counter

Messages received with invalid length

Counts a received message that is longer than 4096 bytes. The message is rejected.

Messages received that cannot be mapped

When receiving messages of a new connection, this value is incremented in case all available connection entries are already in use.

Cause: *ibaPDA* receives more connections than the license supports.

Connection table

Display variants

Green background/ bold text	The connection is ok and there is a module assigned to it.
Green background/ standard text	The connection is ok, but there is no module assigned to it.
Gray background/ bold text	A module is defined, but no connection has been established with these parameters.

Connection data

- Source IP address
- Destination port
- Receive message counter
- Data length of the received telegrams
- Counter for Incorrect message types, e.g. when unicast messages are sent to multicast connections and vice versa.
- Counter for multicast join errors

Possible errors

If the connections are not displayed or only partially, this may have the following causes:

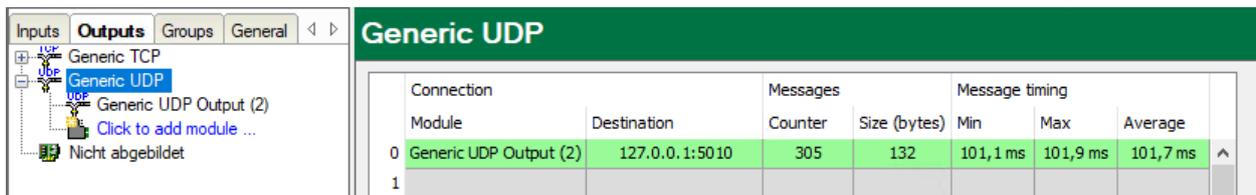
- The controller is not sending.
- No Ethernet connection between *ibaPDA* PC and the controller
- Error in the controller configuration:
 - Incorrect remote IP address
 - The port number does not match the *ibaPDA* settings.
 - The port number is blocked by the firewall.

Other errors:

- If the telegram counters do not increment continuously, the messages are not sent cyclically on the controller side.
- In case the values for the data length change, this is an indication of the following error:
 - Different messages with different layout are sent over the same connection.

4.6 Checking the connection for sent messages

Under the *Generic UDP* output interface you can find the connection table.



Generic UDP							
Connection		Messages		Message timing			
Module	Destination	Counter	Size (bytes)	Min	Max	Average	
0	Generic UDP Output (2)	127.0.0.1:5010	305	132	101,1 ms	101,9 ms	101,7 ms
1							

Display variant

When using UDP messages, the sender does not get any confirmation that the data have been received. Hence, *ibaPDA* cannot indicate faulty connections in the table.

Connection data, telegram data and statistics

- Module name
- Destination: IP address: Port number of the connection
- Sent message counter
- Size of the telegrams sent (user data)
- Telegram cycle, minimum, maximum and average value.

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.