



ibaPDA-Request-TwinCAT

Request Data Interface to TwinCAT Systems

Manual Issue 1.3

> 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 2021, 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
1.3	11-21	Data path UDP Realtime	St	7.3.8

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 Manual5		
	1.1	Target group and previous knowledge	5
	1.2	Notations	6
	1.3	Used symbols	7
2	System	requirements	8
3	ibaPDA-	Request-TwinCAT	. 10
	3.1	General Information	. 10
	3.2	Request blocks	.11
4	Request	t TwinCAT via EtherCAT	. 14
	4.1	System integration with data path EtherCAT	.14
	4.2	Configuration and engineering of the TwinCAT controller	.14
	4.2.1	TwinCAT 2	. 15
	4.2.2	TwinCAT 3	.18
	4.3	Configuration in ibaPDA	.21
	4.3.1	Setting up the connection	.21
	4.3.2	TwinCAT request module	. 23
	4.3.2.1	General Settings	.23
	4.3.2.2	Configuration of the control path	.24
	4.3.2.3	Configuration of the data path	. 30
	4.3.3	Selecting symbols	.31
	4.3.4	Diagnostics	. 32
5	Request	t TwinCAT via UDP	.33
	5.1	System integration with data path UDP and UDP realtime (RT)	.33
	5.2	Configuration and engineering of the TwinCAT controller	.33
	5.2.1	TwinCAT 2	.33
	5.2.2	TwinCAT 3	.34
	5.3	Configuration in ibaPDA	. 35
	5.3.1	Setting up the connection	. 35
	5.3.2	TwinCAT request module	. 36
	5.3.2.1	General Settings	.36
	5.3.2.2	Configuration of the control path	.36

8	Support and contact41		11
7	Technica	l data	10
	6.3	Connection diagnostics with PING	39
	6.2	Log files	38
	6.1	Checking the license	38
6	Diagnost	ics	38
	5.3.3	Selecting symbols	37
	5.3.2.3	UDP and UDP RT data path	36

1 About this Manual

This document describes the function and application of the software interface

ibaPDA-Request-TwinCAT.

The product *ibaPDA-Request-TwinCAT* is an extension of *ibaPDA* for the optional access to variables when recording data from TwinCAT 2 and TwinCAT 3 controllers. This manual only shows the extensions and differences. Refer to the manual from *ibaPDA* for all other functions and operating options.

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 of *ibaPDA-Request-TwinCAT* the following basic knowledge is required and/or useful

- Basic knowledge of *ibaPDA*
- Basic knowledge of network technology
- 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	Step 1 – Step 2 – Step 3 – Step x
	Example:
	Select the menu Logic diagram - Add - New function
	block.
Кеуѕ	<key name=""></key>
	Example: <alt>; <f1></f1></alt>
Press the keys simultaneously	<key name=""> + <key name=""></key></key>
	Example: <alt> + <ctrl></ctrl></alt>
Buttons	<key name=""></key>
	Example: <ok>; <cancel></cancel></ok>
File names, paths	"Filename", "Path"
	Example: "Test.doc"

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.

Тір



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

- *ibaPDA* v7.3.8 or higher
- Basic license for *ibaPDA*
- Additional license for *ibaPDA-Request-TwinCAT*
- Additional licenses if UDP is used as data path:
 - ibaPDA-Interface-Generic-UDP
 - Beckhoff TwinCAT TCP/IP server license on the run time system:

TS6310 for TwinCAT 2, TF6310 or TF6311 for TwinCAT 3

- Additionally required if EtherCAT is used as data path:
 - ibaBM-eCAT
 - FO card of the type *ibaFOB-D* or USB adapter *ibaFOB-io-USB* in the *ibaPDA* computer
- Beckhoff controller TwinCAT 2 or TwinCAT 3
- The TwinCAT ADS Communication Library from Beckhoff must be installed on the *ibaPDA* computer. If TwinCAT 2 or 3 is installed on the computer on which the *ibaPDA* service is running, then the library is normally available.

If TwinCAT is not installed on the *ibaPDA* computer, you have to download TwinCAT 3 ADS Runtime from the Beckhoff website http://www.beckhoff.com. Start the installation program and select full installation. TwinCAT 3 ADS Runtime also supports TwinCAT 2.

- Ethernet connection to the controller
- Libraries with iba Request blocks
 - TwinCATRequestLibCommon.lib for shared blocks and the connection via EtherCAT
 - TwinCATRequestLibUDP.lib for the connection via UDP (Beckhoff TF6310)
 - TwinCATRequestLibUDPRT.lib for the connection via UDP Realtime (Beckhoff TF6311)

System specification

- A maximum of 64 request blocks are supported per controller
- A maximum of 500 signals (analog or digital) or a maximum of 2000 bytes are possible per request block
- A maximum of 512 analog signals and 512 digital signals are supported per *ibaBM-eCAT* device

Licenses

Order no.	Product name	Description
31.001303	ibaPDA-Request-TwinCAT	Extension license for an ibaPDA system to be able to use the request functionality with Beckhoff TwinCAT controllers
31.001075	ibaPDA-Interface-Generic-UDP	Extension license for an ibaPDA system for a generic UDP interface Number of connections: 64

Table 1: Available licenses

Hardware

Order no.	Product name	Description
13.127000	ibaBM-eCAT	Bus monitor for EtherCAT

Table 2: Hardware



3 ibaPDA-Request-TwinCAT

3.1 General Information

The interface *ibaPDA-Request-TwinCAT* is suitable for the measuring data acquisition with a free symbol selection from Beckhoff-TwinCAT controllers via EtherCAT or Ethernet (UDP/IP). The measuring data is actively sent here from the controller to *ibaPDA*. For this purpose, it is necessary to integrate request blocks into the TwinCAT controller. These request blocks serve to cyclically send the current values of the variables selected by the user within *ibaPDA* to *ibaPDA* for recording.

In *ibaPDA*, the variables to be measured are selected with a browser. This makes it possible to access all of the variables available in the controller. The values of the variables can be sent to *ibaPDA* via three different data paths:

- EtherCAT connection via *ibaBM-eCAT*
- UDP connection via *ibaPDA-Interface-Generic-UDP*
- UDP realtime connection via *ibaPDA-Interface-Generic-UDP*

ibaPDA-Request-TwinCAT supports TwinCAT 2 and 3 on industrial PCs and embedded PCs. Bus terminal controllers of the BX/BC series are not supported.

An ibaTwinCAT library must be added to the project in the TwinCAT controller:

- TwinCATRequestLibCommon library for shared blocks and support of EtherCAT as data path
- TwinCATRequestLibUDP library supports UDP as data path. For communication via UDP, the TwinCAT TCP/IP server library must be integrated. This library requires a license and must be installed separately:
 - TS6310 for TwinCAT 2 controllers
 - TF6310 for TwinCAT 3 controllers
- TwinCATRequestLibUDPRT library supports UDP realtime as data path. For communication via UDP realtime, the TwinCAT TCP/IP server library must be integrated. This library requires a license and must be installed separately:
 - TF6311 for TwinCAT 3 controllers

The ibaTwinCATRequest libraries contain the following function blocks:

- Management block IBA_TCREQ_MAN
- Signal data blocks IBA_TCREQ_DATA_ECAT, IBA_TCREQ_DATA_UDP and IBA_TCREQ_DATA_UDPRT

The management block can also be inserted in a (slow) task with low priority. It communicates with *ibaPDA* (via ADS communication) via the control path and checks the list of variables.

The signal data block is assigned to a faster task with a higher priority. It collects the data and sends it to *ibaPDA* with each access on the data path. The EtherCAT signal data block is recommended if you want to measure very short cycle times.



You can find the libraries as archive files on the DVD or USB stick "iba Software & Manuals" at \04_Libraries_and_Examples\10_Libraries\04_TwinCAT\ibaTwinCATLib_Vx.y.z.zip The libraries are available in different versions:

- TwinCAT 2
 - ibaTwinCATRequest.lib (for EtherCAT)
 - ibaTwinCATRequestWithUDP.LIB (for UDP)
- TwinCAT 3
 - TwinCATRequestLibCommon.compiled-library (shared blocks and EtherCAT)
 - TwinCATRequestLibUDP.COMPILED-LIBRARY (for UDP)
 - TwinCATRequestLibUDPRT.COMPILED-LIBRARY (for UDP realtime)

3.2 Request blocks

The request blocks are used to initialize and control the communication between the TwinCAT controller and *ibaPDA*.

A request block set always consists of a management block and a signal data block. There are separate signal data blocks for the connection via EtherCAT and UDP or UDP realtime. The blocks are components of the ibaTwinCAT libraries.

Management block IBA_TCREQ_MAN

Name	Туре	In/Out	Description
Name	STRING(20)	IN	Name of the function block. The same name must be used for the corresponding IBA_TCREQ_DATA function block.
State	IBA_TCREQ_ STATE_MAN	OUT	Status of the function block

The block IBA_TCREQ_MAN may assume the following states (IBA_TCREQ_STATE_MAN):

Status	Description
TCREQ_MAN_INIT	Initial state before the block has registered with its name
TCREQ_MAN_IDLE	Waiting for messages from ibaPDA
TCREQ_MAN_VALIDATING	Validating of the tags received by ibaPDA
TCREQ_MAN_RELEASING_ HANDLES	Releasing handles to variables

Signal data block IBA_TCREQ_DATA_ECAT

Name	Туре	In/Out	Description
Name	STRING(20)	IN	Name of the function block. The same name must be used for the corresponding IBA_TCREQ_MAN function block.
Data buffer	POINTER TO BYTE	IN	Pointer to the data buffer where the values of the requested variables should be written to. This data buffer must be linked with EtherCAT output variables.
MaxDataSize	UINT	IN	The size of the data buffer
State	IBA_TCREQ_ STATE_DATA	OUT	Status of the function block
ADSError	UDINT	OUT	The last error code that was received when reading the data of the requested variables.
Size	UINT	OUT	The actual size of the data written in the data buffer.

Signal data blocks IBA_TCREQ_DATA_UDP and IBA_TCREQ_DATA_UDPRT

Name	Туре	In/Out	Description
Name	STRING(20)	IN	Name of the function block. The same name must be used for the corresponding IBA_TCREQ_MAN function block.
Data buffer	POINTER TO BYTE	IN	Pointer to the data buffer where the values of the requested variables should be written to.
MaxDataSize	UINT	IN	The size of the data buffer.
Adapter	STRING(15)	IN	The IP address of the network adapter that needs to be used to send data to ibaPDA. If this is empty, the default adapter is used.
State	IBA_TCREQ_ STATE_DATA	OUT	Status of the function block
ADSError	UDINT	OUT	The last error code that was received when reading the data of the requested variables.
Size	UINT	OUT	The actual size of the data written in the data buffer.
UsedAdapter	STRING(15)	OUT	The IP address of the network adapter that is used to send data to ibaPDA.
SendCounter	UINT	OUT	Counter that is incremented every time a message is sent to ibaPDA.

The signal data blocks may assume the following states (IBA_TCREQ_STATE_DATA):

Status	Description
TCREQ_DATA_INIT	Initial state. Search for the block with the same name IBA_TCREQ_MAN.
TCREQ_DATA_NO_PATH	Connected to the management block, but no data path available.
TCREQ_DATA_READY	Connected to the management block and data path found. The variables list is empty.
TCREQ_DATA_COPYING	Copying data for the variables list.
TCREQ_DATA_ONLINECHANGE	An online change has occurred. Waiting for the management block to react to this.

4 Request TwinCAT via EtherCAT

4.1 System integration with data path EtherCAT

The measurement data is transmitted via EtherCAT to the *ibaBM-eCAT* device. You need the following connections:

- Ethernet connection between *ibaPDA* and the TwinCAT PLC
- Fiber optic connection between *ibaPDA/ibaFOB-io-D* and *ibaBM-eCAT*
- EtherCAT connection between *ibaBM-eCAT* and TwinCAT PLC



Fig. 1: Topology request via EtherCAT

An additional prerequisite is the ibaTwinCAT request library in the TwinCAT controller.

4.2 Configuration and engineering of the TwinCAT controller

The following configuration and engineering steps are generally to be made in TwinCAT on the TwinCAT side:

Hardware engineering

Integration of the *ibaBM-eCAT* device in the device configuration. You can find a description of this in the corresponding chapters of the *ibaBM-eCAT* manual *System integration* and *Configura-tion in the EtherCAT-Master*.

The only difference when using Request-TwinCAT is the selection of the request device (*ibaBM-eCAT for TwinCAT Vx Request*) and the resulting signal grouping in the EtherCAT configuration.

Software engineering

The integration of the request blocks in the TwinCAT program is described in the following chapters.



4.2.1 TwinCAT 2

You call them as follows:

Open the library manager in your project and add the library *ibaTwinCATRequest.lib* from the directory ...\Vx.y.z\TwinCAT_V2. Now add an instance of the management block IBA_TCREQ_MAN and an instance of the signal data block IBA_TCREQ_DATA_ECAT. You can add the instances to the same program or to different programs.

Create a buffer that is linked with EtherCAT output variables. The syntax AT %Q* labels the data buffer as an output variable. The size of the buffer should be between 256 and 2560 bytes. The size depends on how much data you would like to measure via this request function block pair.

It is recommended to create 2 pairs of blocks to utilize the maximum data volume.

For simplicity's sake, both blocks are accessed in one task in this example.

PROGRAM ibaRequest VAR DataBuffer AT %Q*: ARRAY[0..2559] OF BYTE; ibaReqMan_1 : IBA_TCREQ_MAN; ibaReqData_1 : IBA_TCREQ_DATA_ECAT; ibaReqMan_2 : IBA_TCREQ_MAN; ibaReqData_2: IBA_TCREQ_DATA_ECAT; END_VAR ibaReqMan_1(Name:= 'RequestData_1', State=>); ibaRegData_1(Name:= 'RequestData 1', DataBuffer:= ADR(DataBuffer[0]), MaxDataSize:= 1280, State=> ADSError=>, Size=>); ibaReqMan_2(Name:= 'RequestData_2', State=>); ibaReqData_2(Name:= 'RequestData_2', DataBuffer:= ADR(DataBuffer[1280]), MaxDataSize:= 1280, State=>, ADSError=>, Size=>);

Now link the data buffer variable with the EtherCAT variables from *ibaBM-eCAT*. To do this, open the system manager and add the device *ibaBM-eCAT for TwinCAT V2 request* to your EtherCAT network.





ibaBM-eCAT has 10 request slots. Each request slot consists of 64 DWORD variables, which correspond to 256 bytes. The DWORD variables can be linked with arrays. DWORD variables are used instead of BYTE variables in order to reduce the number of variables.



Fig. 3: 10 request slots per device

Box 1 (ibaBM-eCAT for TwinCAT V2 reques	st) 🔺 Variable Elac	Online		
RequestSlot 0				
•1 Data_0_0	Name:	Data_0_0		
• Data_0_1	T	DWORD		
•↓ Data_0_2	Type.	DWORD		
•↓ Data_0_3	Group:	RequestSlot 0	Size:	4.0
• Data_0_4	Addross	3098 (0xC1A)	Licor ID:	0
• Data_0_5	Address.		User ID.	1
• Data_0_6	Linked to			
•- Data_0_7				
🗣 Data_0_8	Comment			
•1 Data_0_9				
•- Data_0_10				
•- Data_0_11				
	=			

Fig. 4: Data type of the variables

If you select the *ibaBM-eCAT* device in the tree, a table with all variables will be shown on the right side. You can hold down the Shift key to select several variables, which are to be linked with the data buffer.

Click the right mouse button and select *Change multi-link…* in the context menu. The multi-link command is only active if you select variables from the same EtherCAT telegram. By default, the first 5 slots in the first EtherCAT telegram and the last 5 slots in a second EtherCAT telegram are sent.



Fig. 5: Attach variables

The dialog "Attach variable" is opened.

Double click on the data buffer array to begin attaching from the first byte, or select some elements of the data buffer to attach them to other parts of the data buffer.

Attach Variable 256.0 Byte(s) (Output)		×
■ PLC - Configuration □ Itel Standard □ Itel MAIN.DataBuffer > QB 0.0, ARRAY [0.2559] OF BYTE [2560.0] □ Itel MAIN.DataBuffer[0] > QB 0.0, BYTE [1 0] - Itel MAIN.DataBuffer[1] > QB 1.0, BYTE [1 0] - Itel MAIN.DataBuffer[2] > QB 2.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 1.0, BYTE [1 0] - MAIN.DataBuffer[3] > QB 3.0, BYTE [1 0] - MAIN.DataBuffer[4] > QB 4.0, BYTE [1 0] - MAIN.DataBuffer[5] > QB 5.0, BYTE [1 0] - MAIN.DataBuffer[6] > QB 6.0, BYTE [1 0] - MAIN.DataBuffer[7] > QB 1.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 1.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 10.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 10.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 11.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 13.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 15.0, BYTE [1 0] - MAIN.DataBuffer[1] > QB 15.0, BYTE [1 0] -		Show Variables Unused Used and unused Exclude disabled Exclude disabled Exclude same Image Show Tooltips Show Variable Types Matching Type Matching Size Array Mode Offsets Continuous Show Dialog Variable Name Hand over
MAIN DetaBuffer[18] → QB 18.0, BYTE [1.0]	-	Take over Cancel OK

Fig. 6: Attach variables

If you have selected and linked all bytes, the next bytes appear in the selection list when selecting the following slot, here starting at byte 256.

Attach Variable 256.0 Byte(s) (Output)	×
	Show Variables Ounused Used and unused Exclude disabled Exclude other Devices
→ MAIN DotaBuffer[257] > GB 257.6 BTTE [1.0] → MAIN DotaBuffer[258] > GB 258.0 BYTE [1.0] → MAIN DotaBuffer[259] > GB 258.0 BYTE [1.0]	 Exclude same Image Show Tooltips
→ MAINDataBuffer[261] > OB 261.0. BYTE [1:0] → MAINDataBuffer[262] > OB 262.0. BYTE [1:0] → MAINDataBuffer[263] > OB 263.0. BYTE [1:0] → MAINDataBuffer[264] > OB 264.0. BYTE [1:0] → MAINDataBuffer[265] > OB 266.0. BYTE [1:0] → MAINDataBuffer[266] > OB 266.0. BYTE [1:0] → MAINDataBuffer[266] > OB 266.0. BYTE [1:0] → MAINDataBuffer[267] > OB 266.0. BYTE [1:0] → MAINDataBuffer[268] > OB 268.0. BYTE [1:0] → MAINDataBuffer[268] > OB 268.0. BYTE [1:0] → MAINDataBuffer[269] > OB 268.0. BYTE [1:0] → MAINDataBuffer[269] > OB 268.0. BYTE [1:0] → MAINDataBuffer[270] > OB 268.0. BYTE [1:0]	Show Variable Types Matching Type Matching Size All Types Array Mode Offsets Continuous Show Dialog
◆I MAIN DataBuffer[271] > ○B 271.0, BYTE [1.0] →I MAIN DataBuffer[272] > ○B 272.0, BYTE [1.0] →I MAIN DataBuffer[273] > ○B 273.0, BYTE [1.0] →I MAIN DataBuffer[274] > ○B 274.0, BYTE [1.0] →I MAIN DataBuffer[275] > ○B 275.0, BYTE [1.0] →I MAIN DataBuffer[276] > ○B 275.0, BYTE [1.0] →I MAIN DataBuffer[277] > ○B 277.0, BYTE [1.0]	Variable Name Hand over Take over Cancel OK

Fig. 7: Attaching the following variables

If you link the variables in another way, please do not attach more than 1024 bytes at once. Then enable the configuration and download your program.

4.2.2 TwinCAT 3

Add the library *TwinCATRequestLibCommon.compiled-library* from the directory ...\Vx.y.z\Twin-CAT_V3 to your project. Now add an instance of the management block IBA_TCREQ_MAN and an instance of the signal data block IBA_TCREQ_DATA_ECAT.

In addition, create a buffer that is linked with EtherCAT output variables. The syntax AT %Q* labels the data buffer as an output variable. The size of the buffer should be between 256 and 2560 bytes. The size depends on how much data you would like to measure via this request function block pair.

It is recommended to create 2 pairs of blocks to utilize the maximum data volume.

reqManEcat : IBA_TCREQ_MAN; reqDataEcat : IBA_TCREQ_DATA_ECAT; DataBufferEcat AT %Q* : ARRAY[0..2559] OF BYTE;

The management and signal data blocks can be found in the same program or in separate programs. You call them as follows:

```
reqManEcat(Name := 'DemoRequestECAT');
reqDataEcat(Name := 'DemoRequestECAT', DataBuffer := ADR(DataBufferEcat[0]), MaxDataSize := SIZEOF(DataBufferEcat));
```

In the I/O section, add an *ibaBM-eCAT* device to your EtherCAT network. First copy the ESI file of the *ibaBM-eCAT* device into the subdirectory Config/Io of your TwinCAT installation directory.

Then select *ibaBM-eCAT for TwinCAT V3 request* as a TwinCAT 3 request device and add it. *ibaBM-eCAT* has 10 request slots. Each slot consists of an array with 256 bytes. You can use one or more slots per signal data block IBA_TCREQ_DATA_ECAT.



<u>م</u> الآ - 0`	Variable Flag	s Online		
<mark>-</mark> ۵	Name:	Data_0		
 Box 1 (ibaBM-eCAT for TwinCAT V3 request) RequestSlot 0 	Туре:	ARRAY [0255] OF BYT	E	
🔺 📕 Data_0	Group:	RequestSlot 0	Size:	256.0
➡ Data_0[0]	Address:	2750 (0xABE)	User ID:	0
➡ Data_0[1]				
Data_0[2]	Linked to			
Data_0[3]				
Data_0[4]	Comment			
Data_0[5]				

Fig. 8: Request slots in the TwinCAT 3 request device

To link the data buffer with the request slots, select the device in the tree and then select the output variables in the right table. Click the right mouse button and select *Change multi link...* in the context menu. The multi-link command is only active if you select variables from the same EtherCAT telegram. By default, the first 5 slots in the first EtherCAT telegram and the last 5 slots in a second EtherCAT telegram are sent.

TwinCATRequestDemoProject - Microsoft Visual Studio (Administrator) FILE EDIT VIEW PROJECT BUILD DEBUG TWINCAT TWINSA	FE PLC TO	OLS SCOPE WINDOW HE	LP					٣,	Quick Li	aunch (Ctrl+Q)	ρ.	×
0 · 0 図 · 10 · 10 単 単 2 0 · 0 · ● Attac	h	Release - T	winCAT RT (x86) - 2 ADR(lwAdrFb)		- 50	-	± € ₩ □ · .				
E Demo	Project	• 1 • 1 • = E			198.							
Solution Explorer + # X	TwinCATReque	tDemoProject 🔹 🗙 Library Mar	ager 0 MAIN	-					÷	Toolbax		- 4 ×
0000-0-0	General Div								-	Search Toolbox		ρ.
Search Solution Ecology (Citie-)	Contra Interio	CRT riocess bata ornine							10	⊿ General		
H north and a second se	Name:	Box 1 (baBM-eCAT for TwinCAT V.	3 request) Id: 1									
A ReTark (PicTark)	Object Id:	Dx03020001								There are no u	isable controls i	n this group.
MAIN .	Turne	haBMaCaT for TwoCaT V3 more								breg an item c	toolbox.	add it to the
DemoProject Instance	type:	bedrivecht für Twiticht Kollege	*									
B I PicTask Outputs SAFETY C++ Z VO C++	Comment.											
"go bevice 1 (EnerCAT) "" mage "" mage-lefo " mage-lefo " " mage-lefo " " prouts:		Daabled	Create symbols 🗌									
🕫 Frm0State			1.6911	1.2017		1.000	10.7.1.02	1212-022	1.1.1.1			
	Name	Online	Туре	Size	>Addr	In/Out	User ID	Linked to				
♥ Frm0InputToggle	• WcState0	0	BUT	0.1	1522.0	Input	0					
* Frm1State	* WcStatel	0	BIT	0.1	3058.0	Input	0					
• Frm1WcState	* State	8	UINT	2.0	3084.0	Input	0		-			
Finunputioggie Sieur Gaunt	Data 0	X 00 00 00 00 0	ARRAY (0.255) OF BYTE	200.0	20.0	Cutput		MAIN.DataBufferEcat[0]_MAIN.L	lataBultert			
Parifectoria	and the second s		THE ARRAY (0.255) OF BYTE	230.0	100.0	Unifor		Maard Datassumerscall, Sel Maar	Distantiant			
Outputs	Prote 2	X 00 00 00 00 0	ADDAY (0.255) OF BTTE	230.0	0000	output		Main Databumencation2, Main	a Databurra			
Frm0Ctrl	E Data 4 +3	Online Write	00 APPAY (0.255) OF BYTE	256.0	1050.0	Outout	0	MAIN DataBufferEcat(10241 MAI	N DataBuft			
Frm0WcCtrl	St Data 5	Online Energy	00 APPAVIO 2551 OF BYTE	256.0	1562.0	Output	0	MAIN DataBullerEcal[1200] MAI	N DataBuff			
Frm1Ctrl	Data 6	Delever Terre	00 ARRAY 10 2551 OF BYTE	256.0	1818.0	Output	0	MAIN DataBufferErat[1536] MAI	N DataBuff			
Frm1WcCtrl	Tota 7	NEICOSETUTCE	00 APRAV IO 2551 OF BYTE	256.0	2074.0	Output	0	MAIN DataBufferEcat(17921 MAI	N DataBuff			
E DevCtrl	Data 8	Display Mode •	00 ARRAY (0.255) OF BYTE	256.0	2330.0	Output	0	MAIN.DataBufferEcat(20481_MAI	N.DataBuff			
P La InfoData	Data 9	Add to Watch	00 ARRAY (0.255) OF BYTE	256.0	2586.0	Output	0	MAIN.DataBufferEcat(2304), MAI	N.DataBufi			
Box 1 (IDEDIVI-ECATI FOR TWINCAT VS request)	2	Remove from Watch							100 C			
b Sc Data 0	4	Change Link			_			0	- P.			
B RequestSlot 1	Error List	Change MultiPlate a										
RequestSlot 2	T . O .		Clear					Search Error List	· Q •			
RequestSlot 3		Clear Link(s)	cital cital						1.4			
RequestSlot 4	Desi	Save Item As			PI	le		L. Column Project				
RequestSlot 5	3 11/12/20	17 11:58:05 647 ms 'TwinCAT Syste	em' (10000): Saving configuration of COM	server								
P RequestStot 6	TcEventl	ogger!										
RequestSlot 7	4 11/12/20	17 11:58:05 914 ms 'TwinCAT Syste	em' (10000): Shutting down COM Server T	plpServer !						1		
P RequestSlot 8	5 11/12/20	17 11:58:05 914 ms TwinCAT Syste	em' (10000): Shutting down COM Server T	EventLogger	1 ()							
h Courses	0 6 11/12/20 TableSau	17 11:58:06 802 ms TwinCAT Systems	em' (10000): Loading configuration of CO	A server						1		
b InfoData	T 11 (1 2/20)	17 11 59 06 902 out 1 Turof AT Suit	and 20000 London conferentiation of CO							1		
A 21 Mappings	TcEventl	ogger I	ern (2000), coloring configuration of Col	n servel						1		
DemoProject Instance - Device 1 (EtherCAT) 1	A # 11 (12/00	7 11-58-06 804 mr L TuinCAT Curt	m' /10000: Initializing COM Server Toole	anuel.					*		-	_
· · · · · · · · · · · · · · · · · · ·	Error List Outp	ut:								Properties Too	abox	
												100

Fig. 9: Link variables

In the following dialog, select the data buffer or parts of the data buffer. This is then linked with the EtherCAT output variables.

Search: PLC DemoProject DemoProject Instance DemoProject Instance	Show Variables Unused Used and unused Exclude disabled VExclude other Device
MAIN.DataBufferEcat[0] > QB 561180.0, 8YTE [1.0] MAIN.DataBufferEcat[0] > QB 561181.0, 8YTE [1.0] MAIN.DataBufferEcat[2] > QB 561182.0, 8YTE [1.0]	Exclude same Image Show Tooltips Sort by Address
MAIN.DataBufferEcat(3) > QB 561183.0, BYTE [1.0] MAIN.DataBufferEcat(3) > QB 561185.0, BYTE [1.0] MAIN.DataBufferEcat(5) > QB 561185.0, BYTE [1.0] MAIN.DataBufferEcat(6) > QB 561185.0, BYTE [1.0] MAIN.DataBufferEcat(7) > QB 561187.0, BYTE [1.0] MAIN.DataBufferEcat(8) > QB 561188.0, BYTE [1.0] MAIN.DataBufferEcat(8) > QB 561188.0, BYTE [1.0]	Show Variable Types Matching Type Matching Size All Types Array Mode
MAIN.DataBufferEcat[9] > QB 561198.0, BYTE [1.0] MAIN.DataBufferEcat[10] > QB 561191.0, BYTE [1.0] MAIN.DataBufferEcat[11] > QB 561192.0, BYTE [1.0] MAIN.DataBufferEcat[12] > QB 561192.0, BYTE [1.0] MAIN.DataBufferEcat[12] > QB 561193.0, BYTE [1.0] MAIN.DataBufferEcat[13] > QB 561193.0, BYTE [1.0] MAIN.DataBufferEcat[14] > QB 561193.0, BYTE [1.0] MAIN.DataBufferEcat[15] > QB 561195.0, BYTE [1.0] MAIN.DataBufferEcat[16] > QB 561195.0, BYTE [1.0] MAIN.DataBufferEcat[17] > QB 561195.0, BYTE [1.0] MAIN.DataBufferEcat[17] > QB 561195.0, BYTE [1.0]	Offsets Continuous Show Dialog Variable Name / Commer / Hand over / Take over
	Cancel OK

Fig. 10: Attach variables

Enable the configuration and restart the TwinCAT system in order to accept the I/O changes.



4.3 Configuration in ibaPDA

The configuration of is done in the I/O Manager of *ibaPDA*. First set up the connection from *ibaPDA* to the TwinCAT controller via *ibaBM-eCAT*.

Once the connection is set up, add a TwinCAT request module accordingly. See chapter **7** *TwinCAT request module*, page 36.

The configuration of the signals and the selection in the symbol browser is described in the chapter **7** *Selecting symbols*, page 31.

4.3.1 Setting up the connection

The prerequisite for using EtherCAT as data path is the bus monitor *ibaBM-eCAT* and the installation of an *ibaFOB-D* card in the *ibaPDA* computer. After successfully installing the *ibaFOB-D* card, this will be shown in the interface tree.

Other documentation



You can find additional information about the *ibaFOB-D* card in the associated device manual.

To use the TwinCAT request function, *ibaBM-eCAT* must be an active part of the EtherCAT network. That is why the option "active slave" must be enabled in *ibaPDA* and the device must be integrated into the EtherCAT configurator.

For this purpose, the associated ESI file (an XML file) from the supplied DVD or USB stick "iba Software & Manuals" must be imported into the EtherCAT configurator. Then the signals are configured and linked and exported as an ENI configuration file. The configuration file can then be loaded in *ibaPDA*. For additional information, see the chapter **7** *Configuration and engineering of the TwinCAT controller*, page 14.

Configuring ibaBM-eCAT

- 1. Start *ibaPDA* and open the I/O Manager.
- 2. In the I/O Manager, search for the corresponding link of the *ibaFOB-D* card to which *ibaBM-eCAT* is connected. Use the right mouse button to click on the link and select "Autom. detection."

If the device is connected and switched on correctly, the module "ibaBM-eCAT" is shown below the link of the *ibaFOB-D* card. The slot number corresponds to the device address, which is set on the rotary switch S1.

- 3. In the I/O Manager, open the *General* tab of the *ibaBM-eCAT* device module and select the option "True" in the "Enable active slave" field.
- 4. Add a TwinCAT request module to the *ibaBM-eCAT* device module by clicking under the device module (or several if you should need additional connections to other PLCs).

The following figure shows the *General* tab of the *ibaBM-eCAT* device module with the required setting as an "Active slave." In the *General* tab, also load the ENI configuration file of the Ether-CAT network by clicking on the link *Configure symbols*.

G	ieneral 🧼 Diagnostics	C EtherCAT browser
~	Basic	
	Module Type	ibaBM-eCAT
	Locked	False
	Enabled	True
	Name	ibaBM-eCAT
	Use name as prefix	False
	Timebase	1 ms
,	Advanced	
	Timeout	100 ms
	Enable default values	False
	Enable active slave	True
1	Connection	
	IP Address	172.29.0.101
	Auto enable/disable	False
Au Iden to to fti accord	to enable/disable en this option is enabled vice during the start of the dule and start the acquisi "ill ry to reconnect to the tomatically restart the acq his option is not enabled th quisition when it can't con	and ibaPDA can't connect to this acquisition then it will disable this ion without it. During the acquisition device. When it succeeds it will uistion with this module enabled, hen ibaPDA won't start the hect to the device.
		1

Fig. 11: Load the configuration file in the General tab

The ENI configuration file can be exported in TwinCAT 2 from the system manager and in Twin-CAT 3 from the I/O section.

Example of system manager in TwinCAT 2:



Fig. 12: Export of the ENI file in the TwinCAT system manager

4.3.2 TwinCAT request module

4.3.2.1 General Settings

The general settings are identical for the data path via EtherCAT, UDP and UDP Realtime.

~	Connection	
1	Basic	
	Module Type	ibaBM-eCAT\TwinCAT Request
	Locked	False
	Enabled	Irue
	Name	IwinCAI Request
	Module No.	14
	limebase	10 ms
	Use name as prefix	False
1	Module Layout	
	No. analog signals	32
		- Lease
Na	me	
Na	me e name of the module.	

Fig. 13: TwinCAT request module, General tab

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.

Module layout

No. analog signals Determination of the number of analog signals for this module (max. 500)

No. digital signals

Determination of the number of digital signals for this module (max. 500)

4.3.2.2 Configuration of the control path

Requirement

ibaPDA uses the TwinCAT ADS communication library from Beckhoff for the control path. This library is not part of the *ibaPDA* installation. If TwinCAT 2 or 3 is installed on the PC on which the *ibaPDA* service is running, the library is available. If TwinCAT is not installed, you have to download the TwinCAT 3 ADS runtime from the Beckhoff website (http://www.beckhoff.com) (see chapter **7** System requirements, page 8).

Configuration

The control path is configured in the *Connection* tab and is identical to the data path via Ether-CAT and UDP.

🔢 iba I/O Manager				- 0	×
🗋 📂 🎽 🛃 🌖 🕨 🛛 Hardwa	re Groups Technostri	ng Outputs 🛛 🛍 🛍			
⊕ deneral ∧	TestReq (1)			
Click to add module	General S Con	nection 🔨 Analog 🗍 Digital			
Brite Link 0	AMS Net Id:	192.168.80.25.1.1	~	Edit routes	
Elik V	Port:	801 (TwinCAT 2 Runtime 1)	~ [Test connection	
ibaBM-eCAT	Timeout (s):	5		Open log file	
Click to add module	Management FB:	MAIN.REQMAN			~
E 215	Disable module whe	en PLC isn't accessible at the start of the acquisition			
	Disable signals that	have missing symbols			

Fig. 14: Configuration of the control path

You have the following setting and configuration options in the control path section:

AMS Net-ID

The address of the control consists of the AMS net ID and a port number. The AMS net ID is the address of an ADS router and consists of 6 bytes. Normally, the first 4 bytes are the IP address and the last 2 bytes are 1.

Port

The port number determines the ADS device, which is connected to the router. There are predefined port numbers for the TwinCAT runtimes:

- 801, 811, 821, 831: for the 4 possible TwinCAT 2 runtimes
- 851, 852, 853, 854: for the 4 possible TwinCAT 3 runtimes

<Edit routes>

An ADS router is also installed on the *ibaPDA* server computer during the installation of the TwinCAT ADS library. An entry must be made for the remote controller in the routing table of this ADS router. In addition, an entry must be made for the AMS net ID of the *ibaPDA* server computer in the routing table of the ADS router. This can be done via the ADS router configuration tool.

Start the tool with the <Edit routes> button. The following dialog appears:

. 168. 123. 7. 1. 1	192.168.123.7	TCP TP	
		ICI_II	
168.123.8.1.1	192.168.123.8	TCP_IP	
168.123.136.1.1	192.168.123.136	TCP_IP	
168.123.140.1.1	192.168.123.140	TCP_IP	
	106.123.136.1.1 168.123.136.1.1 168.123.140.1.1	1061 123.0.1.1 192.106.123.136 168.123.136.1.1 192.168.123.136 168.123.140.1.1 192.168.123.140	. 168. 123. 136. 1. 1 192. 168. 123. 136 ТСР_IP . 168. 123. 140. 1. 1 192. 168. 123. 140 ТСР_IP

Fig. 15: Start the configuration tool

You can add a new entry using <Add>:

Enter Host Name / IP:				Refresh Statu	s	Broadcast Searc	h
Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Comment	
CX-220CE4	x	192.168.12	192.168.123.7	3.1.4018	Win CE (7.0)	
CX_16FBF7	x	192.168.12	192.168.123.8	2.10.1342	Win CE (5.0))	
devpc-nic3		192.168.12	10.1.10.100.1.1	3.1.4018	Windows 7	2	
devpc-nic4		172,29.0,100	192.168.123.3	3.1.4018	Windows 7		-
devpc-nic4		192.168.12	192.168.123.3	3.1.4018	Windows 7		
devpc-nic4		169.254.12	192.168.123.3	3.1.4018	Windows 7		
ibaBenelux-PC		192.168.12	192.168.123.1	2.11.1536	Windows 7		-
ibaPDA		192.168.12	192.168.123.7	3.1.4018	Windows 7		2
*		III				1	F
oute Name <mark>(</mark> Target):	devpc-ni	c3	Rout	e Name (Remo	ote): DE	VPC-NIC4	
msNetId:	10.1.10	100.1.1	Taro	et Route	Re	mote Route	
			0	Proiect	C	None	
ransport Type:	TCP_IP	•	0	Static	(0	Static	
ddress Info:	devpc-ni	ic3	0	Temporary	e	Temporary	
O Host Name	IP Address		0	remporary		/ remporary	
	-						
onnection limeout (s):	2	· ·					_

Fig. 16: Edit connections

You can search for compatible TwinCAT controllers in the local network by using the <Broadcast Search> button. Either select a found controller or enter the route name, the AmsNetId and the address information manually.

Besides the broadcast search there is also the possibility to configure a connection directly by entering the IP address:

Enter Host Name / IP:	192.168	.80.203	Re	fresh Status	Broad	cast Search
Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Comment
CX-220AA7		192.168.00.000	5.34.10.167.1.1	3.1.4018	Win CE (7.0)	
<						
c pute Name (Target):	CX-2204	IA7	Route N	iame (Remote):	IBA-FUE≺	NOTEH27
v oute Name (Target): nsNetId:	CX-220A 5.34.10	LA7 167.1.1	Route N	lame (Remote): Route	IBA-FUE-	NOTEH27 Route
 oute Name (Target): nsNetId: ansport Type: 	CX-220A 5.34.10 TCP_IP	IA7 167.1.1	Route N Target O Pro	lame (Remote): Route ject	IBA-FUE Remote I O None	Nottines Route

Fig. 17: Connection setting via IP address

Select "Static" for the target route. For the remote route, select "Static" when connecting to a TwinCAT connection and "None" when connecting to a bus controller. Clicking on <Add Route> will enter the route in the routing table. The router attempts to connect to the removed ADS router and may require a user name and a password. If the connection was successful, an "X" will appear in the "Connected" column.

You can also open the configuration tool via the context menu on the TwinCAT task bar symbol. Select *Router - Edit Routes*.

	About TwinCAT	0		
•	Tools			
5	목 <u>R</u> ealtime Settings	- 0 ²		
	Rou <u>t</u> er		Info	į)
+	System		<u>C</u> leanup	Š
1.01	🙀 🕂 🔟 🚥	- 1	Change AMS NetId	
e	📴 📫 🚺 🔺 woe	<mark>کا</mark> کا	Change AMS NetId Edit Routes	

Fig. 18: Context menu TwinCAT task bar symbol

If TwinCAT 2 is installed, the following dialog appears:

Local Computer	AMS Router
AMS Net Id:	192.168.122.108.1.1
Remote Comput	ers
CX_16FBF7	
0	
Add	Remove

Fig. 19: System properties dialog

You can add a new route using <Add>:

Name:	Test	ОК
AMS Net Id:	192.168.123.8.1.1	Cancel
Address:	192.168.123.8	Browse
Transport:	TCP/IP T	Slow Connection

Fig. 20: Add connection

The fields Name, AMS Net ID and Address must be filled out manually. "OK" enters the route in the routing table.

You can also open the configuration tool via the context menu on the TwinCAT task bar symbol. Select *Properties*.



Fig. 21: Open the configuration tool

If the routing tables are correct, you can connect to the PLC with *ibaPDA*. To test the connection, select the *Connection* tab of the request module.

E I I I I I I I I I I I I I I I I I I I	Technostring Outputs	17h				
Service (22)	TwinCAT Re	quest (25)				_
Click to add module	BECK General J Connec	ction 🔨 Analog 👖 Digital				
	Control Path					
10 Sniffer (21)	AMS Net Id:	192.168.123.56.1.1 (DEVPC-NIC3)	~	Edit ro	utes	
Service (23)	Port:	801 (TwinCAT 2 Runtime 1)	~	Test conr	ection	
Click to add module	Timeout (s):	5				
⊕-••••• Link 1	Management FB:	A MAIN.REQMAN				~
ien -	Disable module when	PLC isn't accessible at the start of the acquisition				
Standard (12)	Disable signals that ha	ve missing symbols				
	Data Path					
TwinCAT Request (25)	Maximum data size:	256 🗢 byte				
E→→→ 215	EtherCAT data slot:	Levice 1 (EtherCAT).Box 1 (eCAT-Monitor).SIGNAL_GROUP_0				~
Click to add module DiaF0B-2a-Dexp DiaF0B-2a-Dexp DiaF0B-2a-Dexp DiaF0B-2a-Dexp DiaF0B-2a-Dexp DiaF0B-2a-Dexp DiaF0B-2a-Dexp DiaF0B-2a-Dexp Dia	TwinCAT ADS communicati Successfully connected to PLC state: RUM PLC name: TCatP Runtime version: 2,11 (Loading symbols from PLC Successfully loaded 1985: FB version: v1.0.C Data path: Ethert FB Name: TestR Variables: 0/500 Data bytes: 0/256 Management FB: Waitin Data FB: Stand	on library v31.0.41 is installed 192.168.123.56.1.1:801 IcCrit build 2606) symbols () (TwinCAT 2) AT eq g for messages from ibaPDA by				

Fig. 22: TwinCAT request module, connection tab

<Test connection>

Connection test and output of available diagnostic data. If the connection is successful, *ibaPDA* loads the symbols from the controller.

Timeout

Connection timeout

Management function block

In a normal case, select the next still unused management function block from the address book. If no management function block has been configured, *ibaPDA* searches for the first instance of the IBA_TCREQ_MAN function block in the symbols and uses this. If the management function block is properly configured, *ibaPDA* reads out its properties: Version, name, data path, the number of supported variables, the number of supported data bytes and the states of the management and data function blocks.

If the name of the TwinCAT request module is still the standard name, *ibaPDA* automatically changes the name in the instance name of the management function block. The maximum data size in the data path is automatically adjusted.

<Disable module when PLC is not accessible at the start of the acquisition>

If this option is enabled, the acquisition is started, even if no connection to the PLC can be established. The module is disabled. During the acquisition, ibaPDA tries again to connect with the PLC. If successful, the acquisition is restarted.

If this option is not enabled, the acquisition is not started, even if no connection to the PLC is possible.

<Disable signals that have missing symbols>

If the symbol configuration has changed, the module may contain a symbol that is no longer available. If *ibaPDA* then tries to read the data for this variable, the PLC will return an error. If the option "Disable signals that have missing symbols" is enabled, *ibaPDA* ignores this signal and starts the acquisition without this signal. If this option is not enabled, the acquisition is not started.

4.3.2.3 Configuration of the data path

Carry out the settings for the data path in the Connection tab:

i link 1	Data Path		
Click to add module	Maximum data size:	2560 🜲 byte	
ibaFOB-PlusControl	EtherCAT data slot:	. Device 1 (EtherCAT).Box 1 (ibaBM-eCAT for TwinCAT V2 request).RequestSlot 0	~
Click to add module			

Fig. 23: Select the request slot

Maximum data size

You can set the maximum data size here.

EtherCAT data slot

All of the *ibaBM-eCAT* devices in the network are listed in the drop-down menu. Select a request slot.

Select the variables to be selected. See chapter **7** Selecting symbols, page 31.

4.3.3 Selecting symbols

Once the connection to the PLC has been successfully established, the symbols are loaded and can be selected in the symbol browser.

Open the symbol browser by clicking on the link "Select symbols" in the *General* tab of the TwinCAT request module.



Fig. 24: TwinCAT symbol browser

In the *Symbols* tab, you can select individual or several symbols in the tree. Clicking on <Add> inserts the symbols in the corresponding signal table (analog or digital).

If you have selected an individual symbol, the next symbol is selected after you have clicked on <Add>. You can add consecutive symbols by clicking on <Add> several times.

By double clicking on the symbol, this is also adopted in the symbol table.

The symbols are re-loaded from the PLC with <Update symbols>.

TwinCA1	symbol browser			C X
Symbol:	TwinCAT_SystemInfoVarListAppInfo.OnlineC	hangeCnt		
Datatype:	UDINT			
Address:	0x4040:1098368			
Comment:				
Symb 🖓	ols 🔍 Search		20-0 23	
chan				Search
	And Standard Angeon Appinto ♥ OnlineChangeOn			
Hide syn	bols with an unsupported datatype	Update symbols	Add	Close

Fig. 25: Signal search in the TwinCAT symbol browser

You can search for symbols by name in the *Search* tab. The handling and selection in the search result tree is identical to the selection in the symbol tree.

Note



Text signals (data type String) are only supported by Request TwinCAT via UDP.

4.3.4 Diagnostics

You can find a list of all of the symbols recorded with the TwinCAT request module with the actual value in the *Analog* and *Digital* tabs of the TwinCAT request module.

Issue 1.3



5 Request TwinCAT via UDP

5.1 System integration with data path UDP and UDP realtime (RT)

The measurement data is transmitted via UDP to *ibaPDA*. The prerequisite in *ibaPDA* is the license for the communication interface *ibaPDA*-*Interface-Generic-UDP*.

You need an Ethernet connection via standard network cards.



Fig. 26: Topology request via UDP

An additional prerequisite is the TwinCATRequestLibUDP or TwinCATRequestLibUDPRT library in the TwinCAT controller. In addition, it is necessary to incorporate the TCP/IP server library from TwinCAT. The TCP/IP server library requires a license and must be installed separately:

- TS6310 (UDP) for TwinCAT 2 controllers
- TF6310 (UDP) for TwinCAT 3 controllers
- TF6311 (UDP realtime) for TwinCAT 3 controllers

5.2 Configuration and engineering of the TwinCAT controller

5.2.1 TwinCAT 2

Add the TwinCATRequestLibUDP library from the directory ...\Vx.y.z\TwinCAT_V2 to your project. In addition, TwinCAT TCP server (TS6310) must be installed on the TwinCAT controller.

Create an instance of a management block IBA_TCREQ_MAN and a signal data block IBA_ TCREQ_DATA_UDP. In addition, you must create a buffer in which the data is written. The size of the buffer depends on how much data you would like to measure via this request function block pair.

> ibaReqMan : IBA_TCREQ_MAN; ibaReqData : IBA_TCREQ_DATA_UDP; DataBuffer : ARRAY[0..255] OF BYTE;

The management and signal data blocks can be in the same program or in separate programs. You call them as follows:

iba

```
ibaReqMan(
    Name:= 'DemoRequest',
    State=>
);
ibaReqData(
    Name:= 'DemoRequest',
    DataBuffer:= ADR(DataBuffer[0]),
    MaxDataSize:= SIZEOF(DataBuffer),
    Adapter:= '',
    State=> ,
    ADSError=> ,
    Size=> ,
    UsedAdapter=>);
```

You can leave the entry for "Adapter" empty and mark the output "UsedAdapter" to check whether the correct IP address is used. If the IP address is not correct, then you can set the correct IP address for "Adapter."

5.2.2 TwinCAT 3

Add either the TwinCATRequestLibUDP.COMPILED library or the TwinCATRequestLibUDPRT. COMPILED library from the directory ...\Vx.y.z\TwinCAT_V3 to your project. In addition, TF6310 or TF6311 must be installed on the TwinCAT controller.

Create an instance of a management block IBA_TCREQ_MAN and a signal data block IBA_ TCREQ_DATA_UDP or IBA_TCREQ_DATA_UDPRT. In addition, you must create a buffer in which the data is written. The size of the buffer depends on how much data you would like to measure via this request function block pair.

> ibaReqMan : IBA_TCREQ_MAN; ibaReqData : IBA_TCREQ_DATA_UDP; DataBuffer : ARRAY[0..255] OF BYTE;

When using UDP realtime replace IBA_TCREQ_DATA_UDP by IBA_TCREQ_DATA_UDPRT.

The management and signal data blocks can be in the same program or in separate programs.

You call them as follows:

reqManUdp(Name := 'DemoRequestUDP');
reqDataUdp(Name := 'DemoRequestUDP', DataBuffer := ADR(DataBufferUdp[0]), MaxDataSize := SIZEOF(DataBufferUdp), Adapter := '');

You can leave the entry for "Adapter" empty and mark the output "UsedAdapter" to check whether the correct IP address is used. If the IP address is not correct, then you can set the correct IP address for "Adapter."

5.3 Configuration in ibaPDA

The configuration is done in the I/O Manager of *ibaPDA*. First set up the connection from *ibaPDA* to the TwinCAT controller via Ethernet.

Once the connection is set up, add a TwinCAT request module accordingly. See chapter **7** *TwinCAT request module*, page 36.

The configuration of the signals and the selection in the symbol browser is described in the chapter **7** *Selecting symbols*, page 31.

5.3.1 Setting up the connection

The prerequisite for using UDP or UDP realtime as data path is the interface *ibaPDA-Inter-face-Generic-UDP*. If all system requirements are met, the "Generic UDP" interface will be displayed in the interface tree. TwinCAT request is a module of this interface.

19 iba I/O Manager											×
E Bernal Ber	are Gro	eneric U	string Outpu	ts to the							
Click to add module Click to add module	UDF Mes Mes	P port list: 5 sages received sages received	010-5017 with invalid leng that can't be ma	th:	0		i Li	A [llow ports thr Reset st	ough firewall atistics	
		Address	Port	Message counter	Packet size Actual	Incorrect message type	Multicast join error	Time Actual	Time Min	Time Max	
	0	?	?	?	?	?	?	?	?	?	^
	1	?	?	?	?	?	?	?	?	?	
	2	?	?	?	?	?	?	?	?	?	

Fig. 27: "Generic UDP" interface

The interface itself has the following functions and configuration options.

UDP port list

Ports where *ibaPDA* waits for incoming UDP telegrams. You can enter the port numbers as a range or as an enumeration or both combined. Enter a range with a hyphen and separate non-consecutive numbers with commas. The range 5010-5017 is default. The port number must be identical in the controller (see *Configuring the controller* in the manual *ibaPDA-Interface-Generic-UDP*).

Allow ports through firewall

When installing *ibaPDA*, the default port numbers of the protocols used are automatically entered in the firewall. When the port number is changed or when the interface was activated subsequently, it is necessary to allow the ports in the firewall here by clicking on this button.

Counter for faulty telegrams Connection table

Note



You can find additional information about the interface *ibaPDA-Interface-Generic-UDP* in the associated manual.

Add a TwinCAT request module by clicking below the interface. Select the module type "Twin-CAT request" and click on <OK>.

5.3.2 TwinCAT request module

5.3.2.1 General Settings

The general settings to be made are identical to chapter **7** General Settings, page 23.

5.3.2.2 Configuration of the control path

The configuration of the control path is identical to the configuration in chapter **7** Configuration of the control path, page 24.

5.3.2.3 UDP and UDP RT data path

The data path is configured in the *Connection* tab. You have the following setting options:

Data Path	
Maximum data size:	1024 文 byte
TwinCAT IP address:	192.168.80.24
ibaPDA IP address:	192.168.80.10 ~
ibaPDA port:	5010 🜩

Fig. 28: Configuration of the UDP data path in the Connection tab

Maximum data size

Maximum number of data bytes that can be received by the PLC. The value is automatically entered when testing the connection.

TwinCAT IP address

IP address of the PLC. *ibaPDA* must know the IP address in order to correctly identify the UPD data coming from the PLC. The IP address is automatically entered when testing the connection.

ibaPDA IP address

IP address of the computer on which the *ibaPDA* service is running. The PLC requires the IP address in order to know where it should send the data. From the drop-down menu, select the IP address that can be accessed from the PLC.

ibaPDA Port

UDP port number where the PLC should send the data. There must be a port within the port range that is configured in the generic UDP interface. The first port is selected by default. If several request function blocks are configured on the same PLC, you have to assign each one a different port number. Request function blocks on different PLCs can use the same port number.

5.3.3 Selecting symbols

The symbols are selected identically to chapter **7** Selecting symbols, page 31.

Note



Text signals are only supported by Request TwinCAT via UDP.

Note



For further processing of the text signals (datatype STRING) or splitting these into other text signals use a text splitter module under the *Virtual* interface.



6 Diagnostics

6.1 Checking the license

If the "TwinCAT request" modules are not shown in the signal tree, you can check in the I/O Manager under "General - Settings - License info" whether your license *ibaPDA-Request-Twin-CAT* is properly detected.

icense no. :	Please Traile T	^
Customer Name:	Request TwinCAT	
icense time limit:	Water Triane 275	
Dongle HW ld:	Search 22 Western 7 Western 270	
Data stores:	Western IF	

Fig. 29: Display of the license in the I/O Manager

6.2 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)

6.3 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.



Fig. 30: PING successful

With no existing connection you receive error messages.



Fig. 31: PING unsuccessful

7 Technical data

Brief description					
Name	ibaPDA-Request-TwinCAT				
Order number	31.001303				
Request interface					
	UDP and UDP RT ibaBM-eCAT				
Request blocks	6	4			
Data volume/block	a total of 500 signals (analog	g or digital), max. 2000 bytes			
Hardware restriction	Max. 512 analog and 512 digital signals				
Transmission speed	fast	cycle precise			
iba licenses					
ibaPDA base license*					
e.g. ibaPDA-V7-64 (30.770064)	\checkmark	\checkmark			
ibaPDA-Request-TwinCAT* (31.001303)	\checkmark	\checkmark			
ibaPDA-Interface-Generic- UDP* (31.001075)	\checkmark	-			
ibaBM-eCAT* (13.127000)	-	\checkmark			
TwinCAT licenses					
ibaRequest blocks	TwinCATRequestLibUDP				
	TwinCATRequestLibUDPRT	IWINCAIREQUESTLIDCOMMON			
TwinCAT TCP/IP Server*	TwinCAT 2: TS6310				
	TwinCAT 3: TF6310 or TF6311	-			
TwinCAT-ADS communication	✓	✓			

* with costs

8 Support and contact

Support

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.

