



# ibaPDA-Request-SD-TDC

Interface for the measurement data acquisition  
for SIMADYN D/SIMATIC TDC

Manual  
Issue 5.0

Measurement Systems for Industry and Energy

[www.iba-ag.com](http://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](http://www.iba-ag.com).

Version	Date	Revision	Author	Version SW
5.0	07-2023	GUI new, change in license description	st, mm	8.0.0

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

<b>1</b>	<b>About this documentation .....</b>	<b>6</b>
1.1	Target group and previous knowledge .....	6
1.2	Notations .....	6
1.3	Used symbols.....	7
<b>2</b>	<b>System requirements .....</b>	<b>8</b>
<b>3</b>	<b>Product description .....</b>	<b>9</b>
3.1	Overview.....	9
3.2	Basics .....	10
3.3	Configurations .....	11
3.3.1	ibaFOB-SD/-SDexp to SD-master-slave system (CS14).....	12
3.3.2	ibaFOB-SD/-SDexp to SD-master (STRUC) and TDC-slave .....	12
3.3.3	ibaFOB-SD/-SDexp to SD-master (CFC) and TDC-slave .....	13
3.3.4	ibaFOB-SD/-SDexp to TDC-master and -slave (CP53).....	14
3.3.5	ibaFOB-SD/-SDexp to TDC-master and SD-slave (CFC) .....	14
3.3.6	ibaFOB-SD/-SDexp to TDC-master and SD-slave (STRUC) .....	15
3.3.7	ibaFOB-TDC/-TDCexp to SIMATIC TDC (GDM) .....	15
3.4	Functional principle .....	16
3.4.1	Request with SER04A/-B.....	17
3.4.2	Request with SER05A/-B.....	18
<b>4</b>	<b>Configuration on SIMADYN D/SIMATIC TDC.....</b>	<b>20</b>
4.1	Service function blocks SER04A/-B .....	20
4.2	Service function blocks SER05A/-B .....	22
4.3	ibaPDA technosting (TS) .....	24
4.4	Time synchronization on Siemens side .....	25
4.5	Address books .....	26
4.6	Other characteristics.....	26
4.6.1	Processor load .....	26
4.6.2	Load limitation.....	28
4.6.3	Transmission cycle .....	29
4.6.4	Storage requirement.....	29

<b>5</b>	<b>Configuration and engineering ibaPDA .....</b>	<b>30</b>
5.1	ibaPDA standard settings .....	30
5.2	Hardware interfaces ibaFOB-SD/ibaFOB-TDC .....	30
5.2.1	Schematic representations of the iba cards.....	31
5.2.2	Link level .....	33
5.3	TDC Request/Simadyn Request data interface .....	34
5.3.1	Settings .....	34
5.3.2	Creating PDA channels.....	35
5.3.3	Temporal sequence of the measurement for SER04 .....	38
5.3.4	Temporal sequence of the measurement for SER05 .....	39
5.3.5	Address books .....	40
5.3.6	Manual assignment of address books .....	45
5.3.7	Macros .....	45
5.4	Request modules .....	46
5.4.1	General settings .....	46
5.4.2	Signal selection with symbol browser .....	48
5.4.3	Signal selection by Drag & Drop (only with CFC) .....	50
5.4.4	Data modules.....	51
5.5	Time synchronization in ibaPDA .....	52
5.5.1	Time synchronization with DCF77 signal .....	52
5.5.2	Synchronization with IEC time signal .....	53
5.6	ibaFOB-SD/ibaFOB-TDC diagnostics .....	54
5.6.1	Link Info tab .....	54
5.6.2	Processor Info tab.....	55
5.6.3	Report control tab .....	56
5.6.4	Command/Acknowledge tab .....	57
5.6.5	Data control tab .....	58
5.6.6	Configuration tab.....	59
5.6.7	Channels tab .....	59
5.6.8	Timing tab.....	61
5.6.9	Memory view tab .....	63

5.7	ibaFOB-SDexp/ibaFOB-TDCexp diagnostics .....	64
5.7.1	Configuration tab .....	64
5.7.2	System Info tab .....	65
5.7.3	Timing tab .....	66
5.7.4	Active data channels tab .....	68
5.7.5	Channels in system tab .....	68
5.7.6	Memory view tab .....	69
<b>6</b>	<b>Diagnostics .....</b>	<b>70</b>
6.1	License .....	70
6.2	Visibility of the interface .....	70
6.3	Log files .....	71
6.4	Connection diagnostics with PING .....	72
<b>7</b>	<b>Appendix .....</b>	<b>73</b>
7.1	Error of sequence control .....	73
7.1.1	Analyzing the error counters .....	73
7.1.2	Errors with diagnostic functions .....	73
7.1.3	Timeout error of ibaPDA .....	73
7.1.4	Timeout errors of sequence control .....	74
7.2	Errors of the TDC driver .....	75
7.2.1	Error classes .....	75
7.2.2	Errors with ibaPDA log in to the computer coupling .....	75
7.2.3	Errors with channel log in .....	77
7.2.4	Errors with data transfer .....	80
7.3	Display of service blocks .....	81
7.3.1	SER04A/SER05A .....	81
7.3.2	SER04B/SER05B .....	83
7.4	Abbreviations .....	84
<b>8</b>	<b>Support and contact .....</b>	<b>85</b>

# 1 About this documentation

This documentation describes the function and application of the software-interface *ibaPDA-Request-SD-TDC*.

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 of *ibaPDA-Request-SD-TDC* 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	<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

### iba Hardware

- PCI card *ibaFOB-TDC* or *ibaFOB-SD* or
- PCI-Express card *ibaFOB-TDCexp* or *ibaFOB-SDexp*

### iba Software

- Base license for *ibaPDA* v8.0.0 or higher
- Additional license for *ibaPDA-Request-SD* or *ibaPDA-Request-TDC*
- Block library FBAPDA version 140129V001 or higher (for CFC)
- Block library FBAPD5\_990619 (for STRUC)

### Siemens hardware

#### Interface for *ibaFOB-SD/-SDexp*:

- 1 free port on SIMADYN D component CS12, CS13 or CS14
- 1 free port on SIMATIC TDC component CP53 (D7-SYS V7.1 or higher)

#### Interface for *ibaFOB-TDC/-TDCexp*:

- 1 free port on CP52IO in the Global Data Memory (GDM)

### Siemens software

#### For *SD-/TDC-Request* with **SER04**:

- STRUC starting V4.2.1 or higher  
with block library FBA121 or
- CFC V6.0 or higher with D7-SYS V6.0 or higher  
with Siemens block library FBA121 or iba block library FBAPDA

#### For *SD-/TDC-Request* with **SER05**:

- CFC V6.0 or higher  
with D7-SYS V6.0 or higher with iba block library FBAPDA

### Licenses

Order no.	Product name	Description
31.001320	ibaPDA-Request-SD	<i>ibaPDA</i> additional license for optical connector request via SIMADYN D, light rack coupling
31.001330	ibaPDA-Request-TDC	<i>ibaPDA</i> Request package for free access to communicate via fiber optic link with SIMATIC TDC

## 3 Product description

### 3.1 Overview

iba AG offers the following request solutions for Siemens controls:

Platform	Interface	Transmission	iba Interface	Manual	License
<i>SIMADYN D</i>	<i>CS12/13/14</i>	<i>FO</i>	<i>ibaFOB-SD</i>	<i>Request-SD-TDC</i>	<i>Request-SD</i>
<i>SIMATIC TDC</i>	<i>CP53</i>		<i>ibaFOB-SDexp</i>		<i>Request-TDC</i>
	<i>GDM</i>		<i>ibaFOB-TDC</i> <i>ibaFOB-TDC-exp</i>		
	CP50	PROFIBUS	ibaBM-DP	Request-FM458-TDC	Request-FM458/TDC
S7-400	FM458		ibaBM-DPM- ibaCom-L2B		
	S7-CPU	PROFINET	ibaBM-PN	Request-S7-DP/PN/ ibaNet-E	Request-S7-DP/PN/ ibaNet-E
S7-300	UDP		ibaNet-E		
S7-1200	CP x43		Ethernet	Request-S7-UDP	Request-S7-UDP
S7-1500			TCP/IP, MPI, DP	Ethernet, MPI-adapter	S7-Xplorer

This manual describes the ranges SIMADYN D and SIMATIC TDC which are marked in italics in the table.

#### Other documentation



For older SIMADYN D systems (STRUC V2.2 or higher), iba AG offers a SIMADYN D plug-in card, called ibaLink-SM-64-SD16, with which you can transmit 64 analog and 64 digital signals to the standard ibaFOB cards.

Other than these request interfaces, there are also the standard interfaces via PROFIBUS and TCP/IP available.

The manuals are available on the "iba Software & Manuals" data storage medium and in the download area of the iba website.

## 3.2 Basics

You can connect *ibaPDA* directly to SIMADYN D and SIMATIC TDC systems with the *ibaFOB-SD* and *ibaFOB-TDC* cards.

Communication protocols are implemented on these cards that facilitate direct connection to the Siemens rack links. From the perspective of a Siemens control, the *ibaPDA* computer acts like a connected Siemens rack.

In this document, all coupling components are designated as rack links, to which an *ibaFOB-SD* or *ibaFOB-TDC* card can also be connected. These are:

- With SIMADYN D: CS12, CS13 and CS14
- With SIMATIC TDC: CP53 and GDM (CP52IO interface card)

All CPUs of all racks that are connected by a rack link component can communicate with one another. If the *ibaFOB-SD* or *ibaFOB-TDC* is likewise connected with this rack link, then all these CPUs can basically also communicate with *ibaPDA*.

In practice, this number is limited by:

- The max. channel number of 50 PDA channels per *ibaFOB-SD/ibaFOB-TDC* card
- The limited data rate of *ibaFOB-SD/ibaFOB-TDC* cards
- The capability of the FO connection

In an *ibaPDA* computer, you can operate 4 *ibaFOB-SD* or *ibaFOB-TDC* cards. The cards differ as follows:

	<b>ibaFOB-SD/-SDexp</b>	<b>ibaFOB-TDC/-TDCexp</b>
FO interface	ST technology	SC technology
Baud rate	96 Mbit/s	640 Mbit/s
Possible coupling partners	SIMADYN D CS12/CS13/CS14	
	SIMATIC TDC CP53	SIMATIC TDC CP52IO (GDM)

	<b>ibaFOB-SD/-TDC</b>	<b>ibaFOB-SDexp/-TDCexp</b>
PC interface	PCI	PCI Express
Memory access	via CPU	DMA
Request procedure	SER04A/-B	SER04A/-B
		SER05A/-B

### Other documentation



The HW descriptions of the *ibaFOB-SD* and *ibaFOB-TDC* components are available on the "iba Software & Manuals" data storage medium and in the download area of the iba website.

**Special features of the CP53 component (available in SIMATIC TDC starting with D7-SYS V7.1)**

you can configure the component a master or slave. In the master mode, you can use one connection to connect to *ibaPDA* (*ibaFOB-SD* card), the other to connect to a slave rack. The slave rack is either SIMADYN D with the CS22 component or SIMATIC TDC with the CP53 component (slave mode). In slave mode, you cannot use the 2<sup>nd</sup> connection.

If the component is initialized in the STRUC mode for connection with SD under STRUC, then no *ibaPDA* access to the CFC-CPU is possible.

**Special features of the CS12/13/14 components (SIMADYN D)**

The components differ only in the number of FO connections (1, 4 or 8). When mentioning the CS14 in the following, the information also applies to CS12 and CS13. Any free connection can be occupied on one of these components.

**Special features of the Global Data Memory (GDM) (in SIMATIC TDC)**

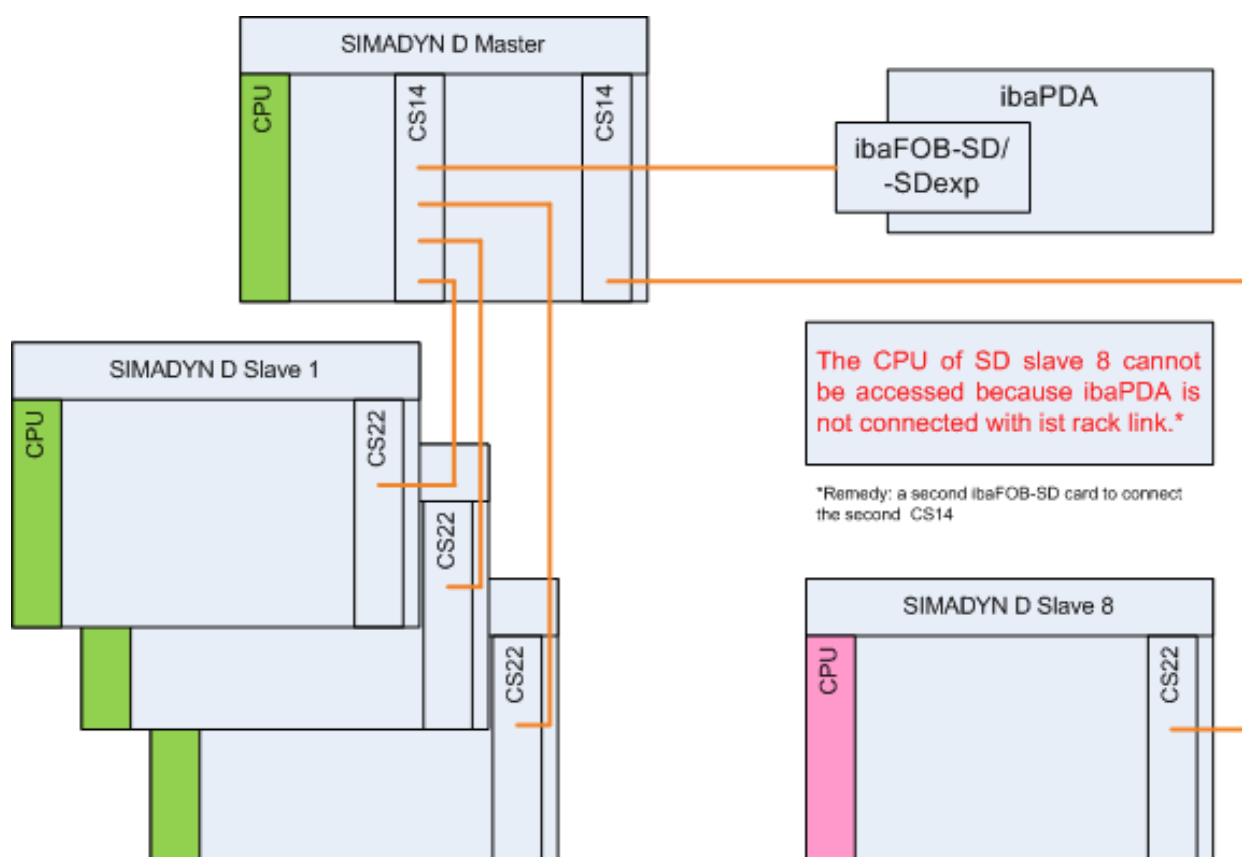
The GDM is a complete rack containing only the CP52M0 memory component and up to 11 interface components (CP52IO). Each interface component provides 4 connections. All connections are on an equal footing.

### 3.3 Configurations

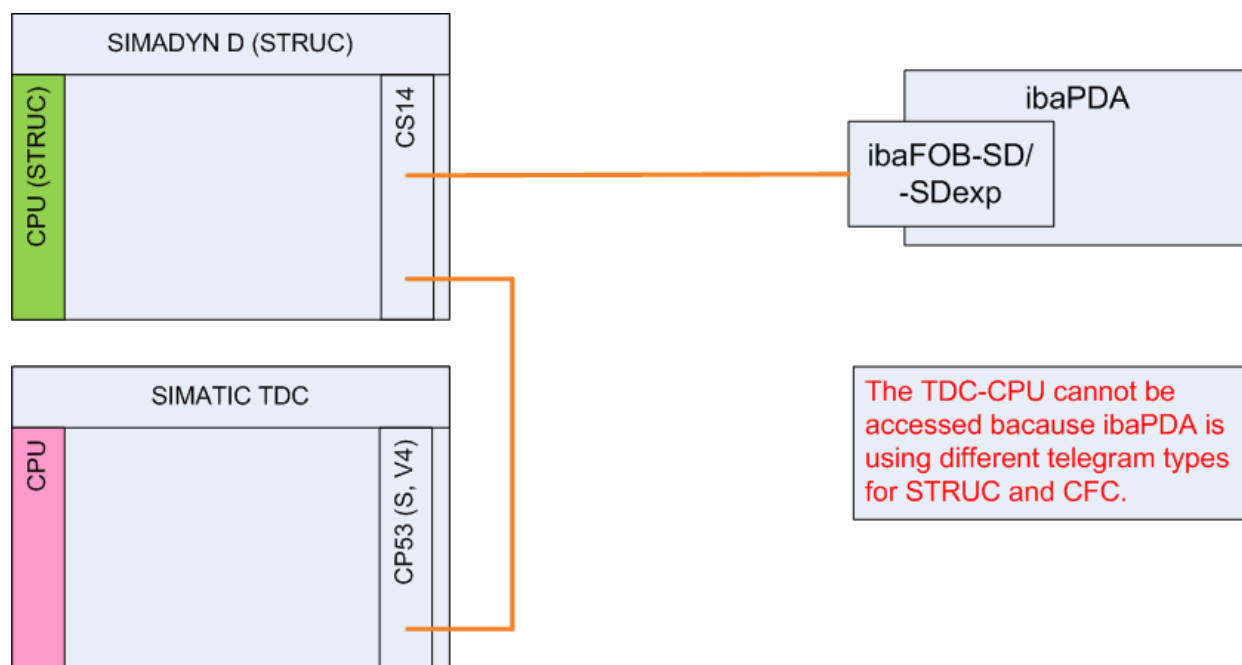
Explanations for the following examples:

- Green CPU: CPU to which *ibaPDA* has access
- Red CPU: CPU to which *ibaPDA* does not have access
- CP53 parameter:
  - S: CP53 as slave
  - M: CP as master
  - V4: CP53 in STRUC mode
  - V6: CP53 in CFC mode

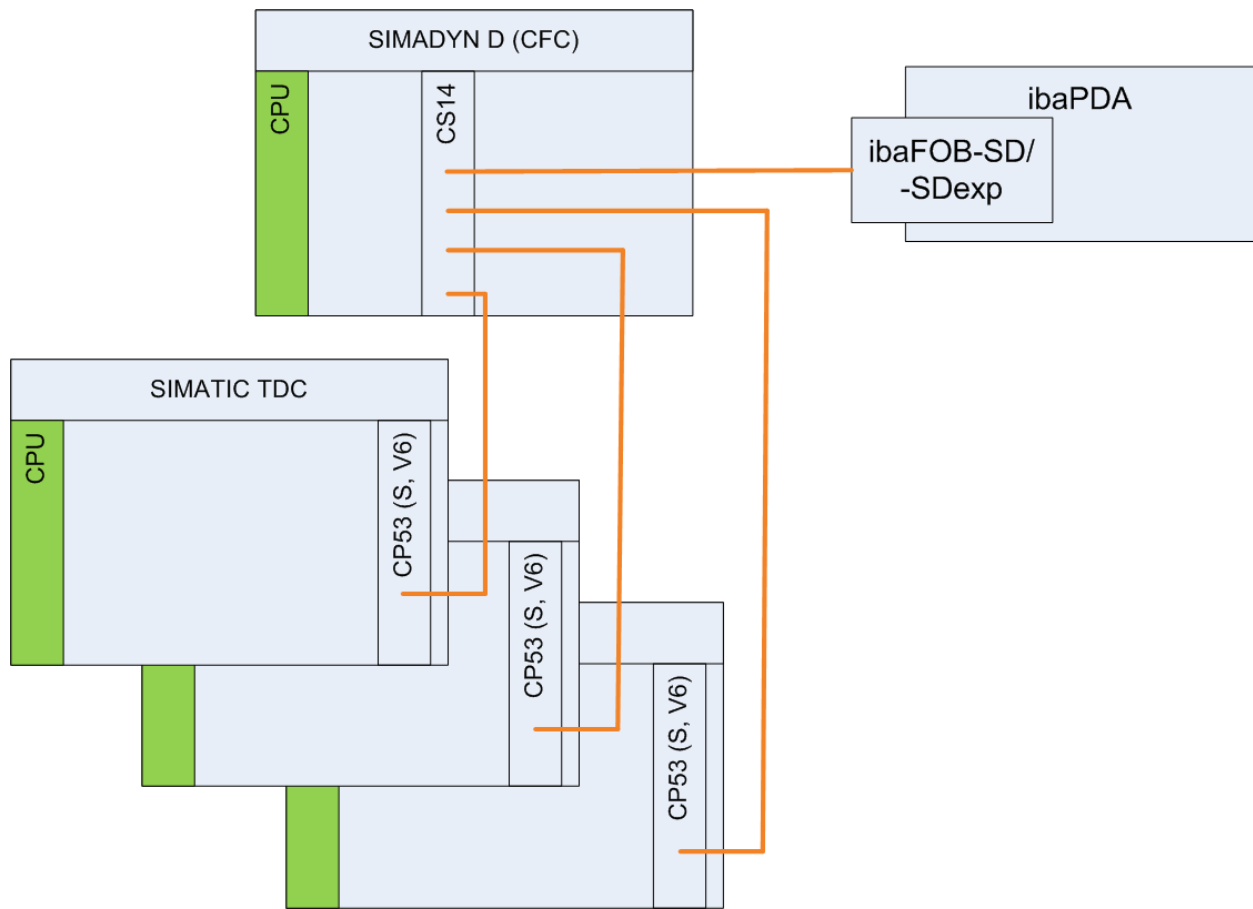
### 3.3.1 ibaFOB-SD/-SDexp to SD-master-slave system (CS14)



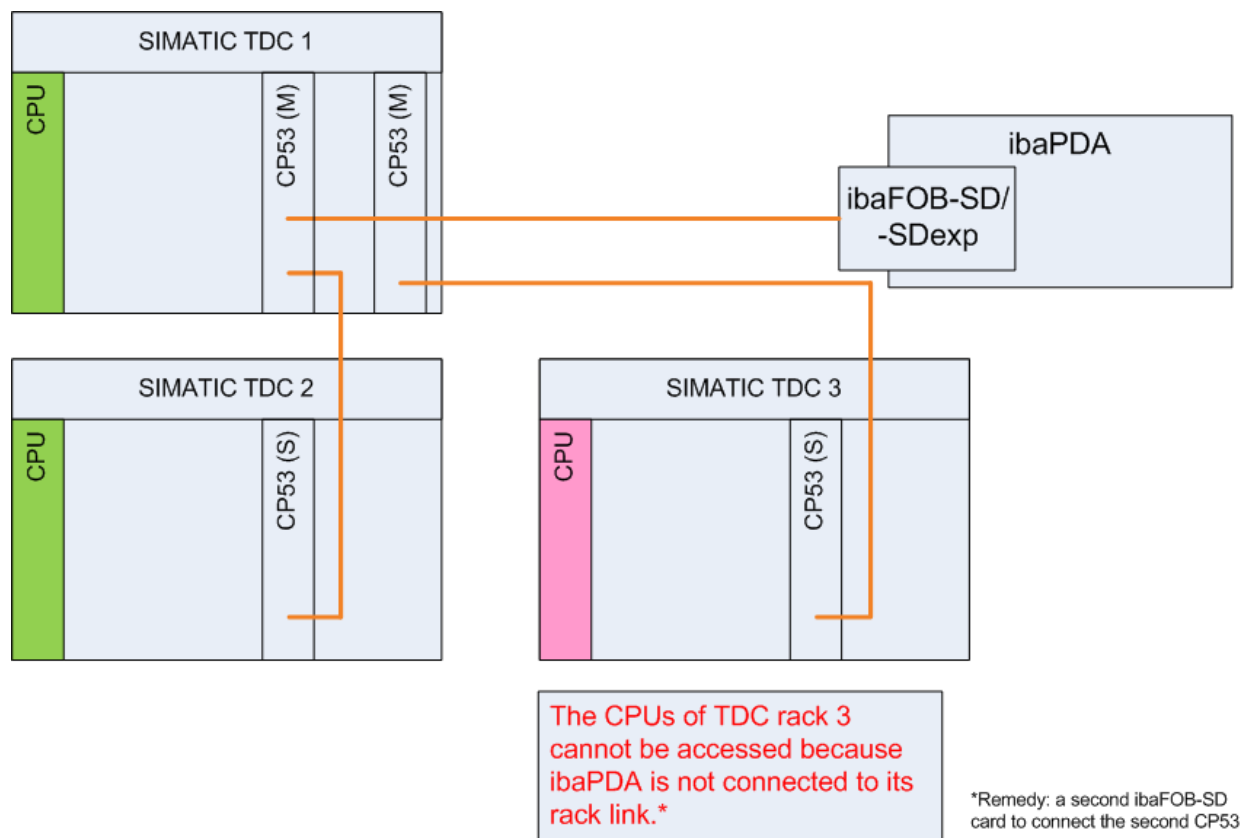
### 3.3.2 ibaFOB-SD/-SDexp to SD-master (STRUC) and TDC-slave



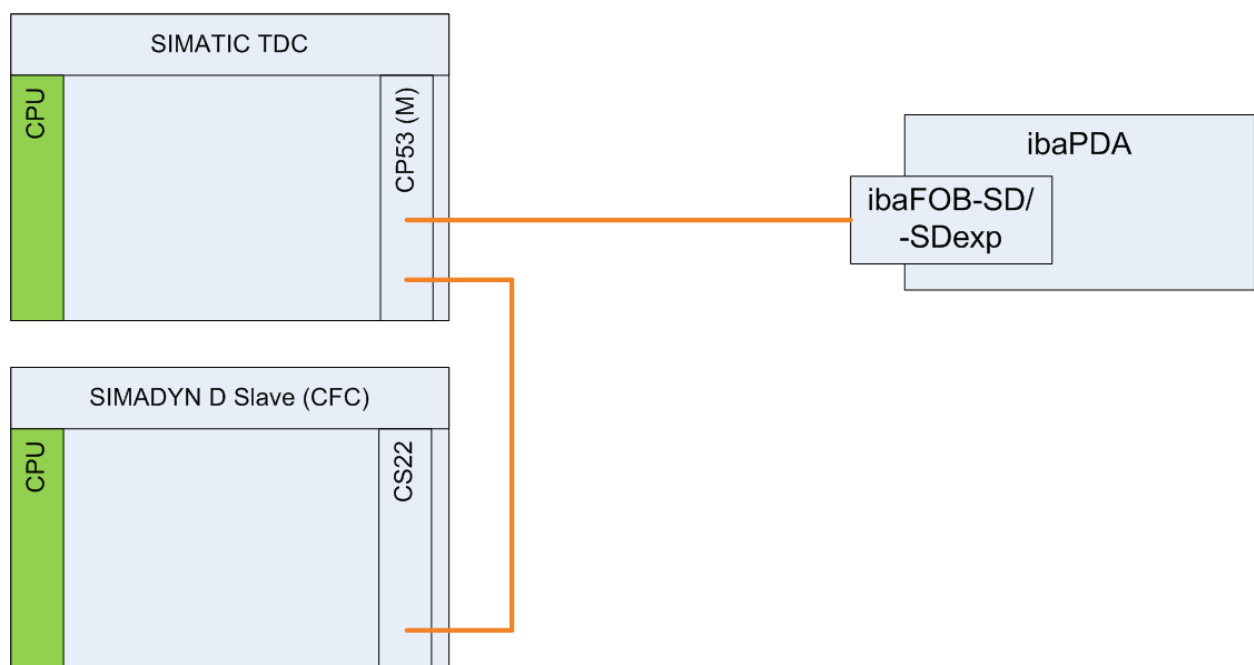
### 3.3.3 ibaFOB-SD/-SDexp to SD-master (CFC) and TDC-slave



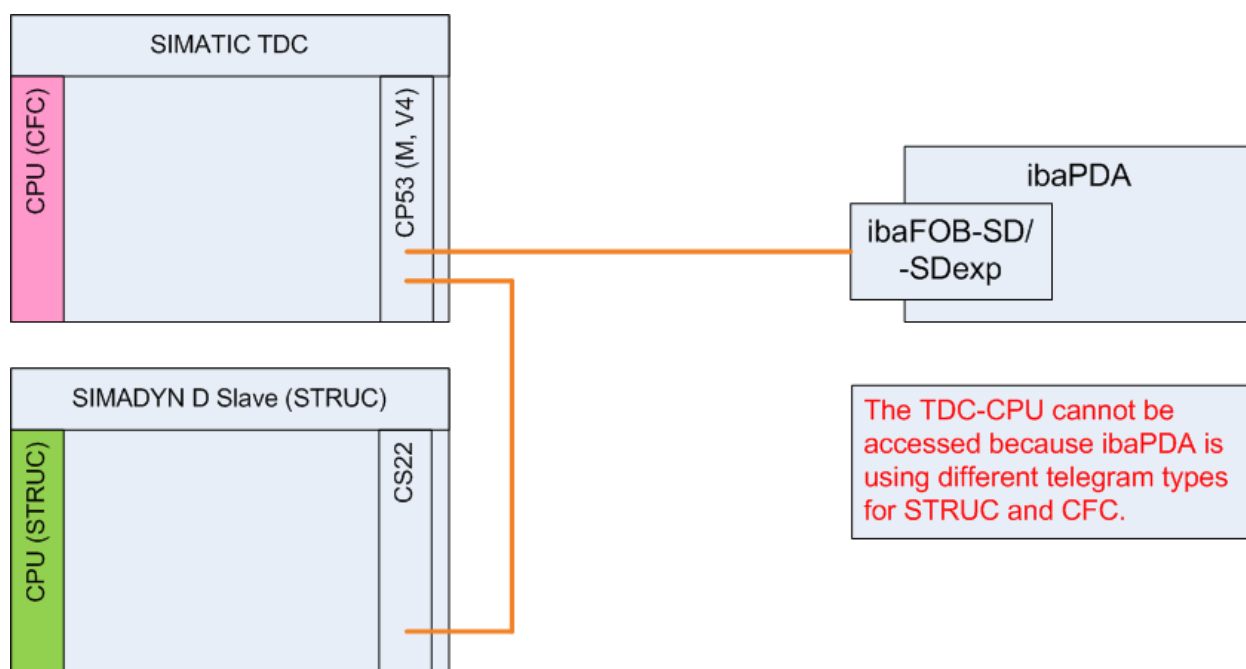
### 3.3.4 ibaFOB-SD/-SDexp to TDC-master and -slave (CP53)



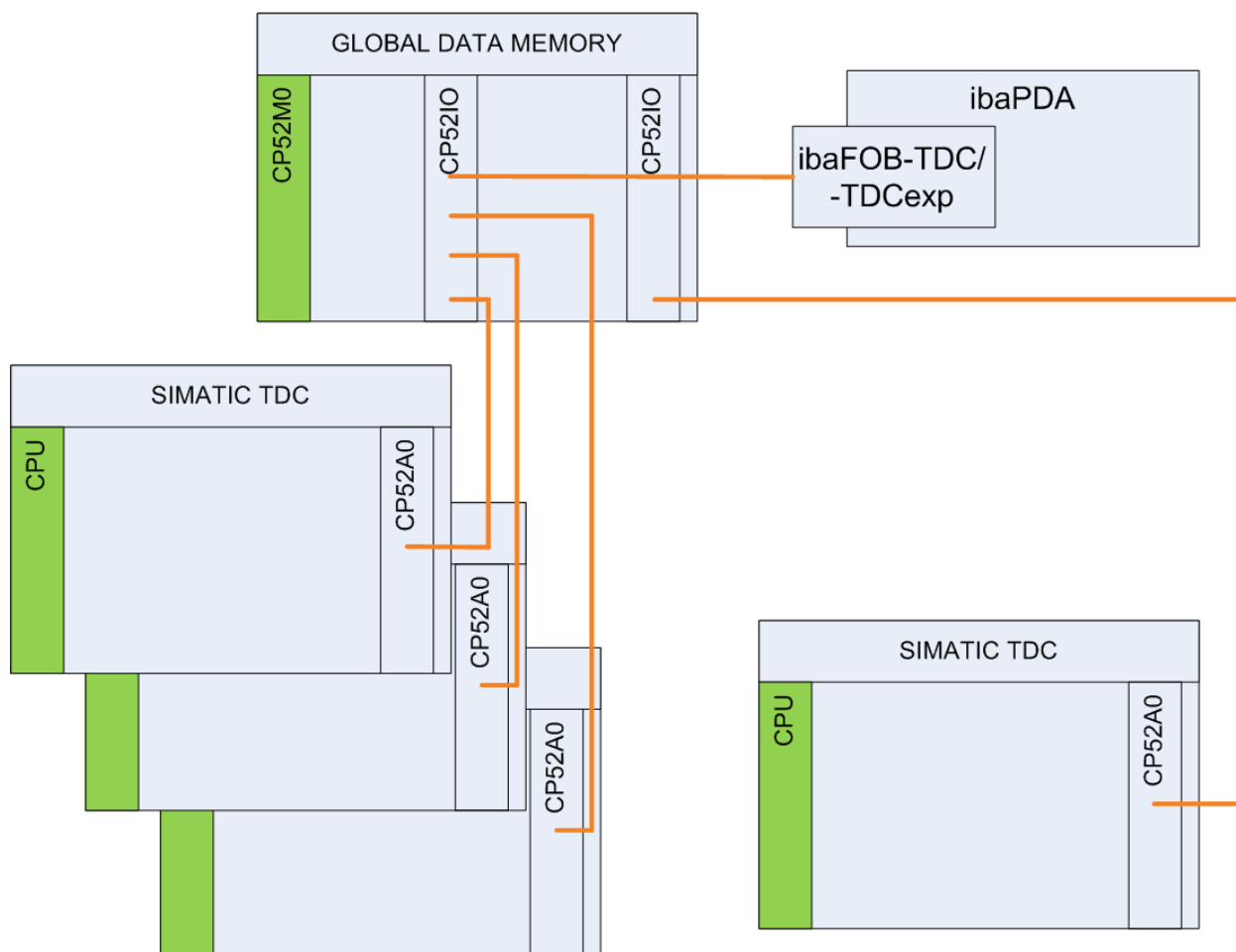
### 3.3.5 ibaFOB-SD/-SDexp to TDC-master and SD-slave (CFC)



### 3.3.6 ibaFOB-SD/-SDexp to TDC-master and SD-slave (STRUC)



### 3.3.7 ibaFOB-TDC/-TDCexp to SIMATIC TDC (GDM)



## 3.4 Functional principle

In the measuring operation, numerical and digital signals are measured, i.e., variables are read out of the computer system and recorded. In defining the variables to be measured, there are two different procedures: SD-TDC-Lite and Symbolic Request.

The iba cards and *ibaPDA* support both procedures. However, only one mode per card can be selected. If there are multiple cards and both licenses available, both procedures can be used simultaneously in *ibaPDA*.

For one card, the following applies: If there is defined at least 1 Lite channel, a Request channel cannot be assigned to that card. In case, a Request channel has been assigned to that card, no more Lite channels can be added.

### SD-TDC Lite

All values that you want to measure with *ibaPDA* first have to be configured in process data telegrams. To change the measurement scenario, you have to change the configuration.

---

#### Other documentation



You can find the description of the SD-TDC Lite access mode in the manuals iba-FOB-SDexp and ibaFOB-TDCexp. The manuals are available on the "iba Software & Manuals" data storage medium and in the download area of the iba website.

---

### Symbolic Request

You define the variables to be measured not in the control, but rather in *ibaPDA*.

The information about the configured variables, the so-called address book, is either read directly from the CPUs by *ibaPDA* in a configuration operation (with configuration under CFC) or provided by the configuration as an ASCII file (with STRUC).

You can select the variables to be measured from the address book with a browser. Thus, you can apply different measurement scenarios without having to change the configuration.

On the control side, for each PDA channel you have to configure 2 or 5 service function blocks (agents) on the CPU.

In Symbolic Request, the connection of a processor to *ibaPDA* is referred to as *PDA channel*. One channel is identified by 2 signs *nn*. It is recommended to use the processor short name as name for the channel<sup>1)</sup>.

For each CPU, multiple PDA channels can be configured. The channel names need to be different and have to be unambiguous per rack link and per *ibaPDA* system

<sup>1)</sup> The component names of the processors are only unambiguous within one rack, but not within a communication island. For an unambiguous designation of the communication channels between the processors of different racks, processor short names consisting of 2 signs are used. These short names are defined by the user and need to be unambiguous for all systems that are connected to *ibaPDA*. For Siemens projects, the short names are usually pre-defined in the Siemens configuration guidelines.

### 3.4.1 Request with SER04A/-B

Two function blocks SER04A and SER04B are required per PDA channel.

The SER04A administrator component establishes two communication connections to *ibaPDA*. Via the "PDAnnCMD" reception channel, the function block receives the requests of *ibaPDA* either for transmitting the address book or the measured data.

When transmitting address books, the names of all charts, blocks and connectors are transferred to *ibaPDA*. The block creates a transmission channel "ADRB\_PDA", that is used by all CPUs of a rack. For a measurement data request, *ibaPDA* passes the names of chart, block and connector for every variable. Using this information, the block calculates the memory addresses and data types of the measured values and transfers both in a common memory area to the connected SER04B block. In a "nnPDAACK" transmission channel, the SER04A acknowledges the request or returns error messages, in case the connector does not exist or the data type is not permitted.

The SER04B transfer block creates a transmission channel "nnPDADAT" for the fast cyclic transfer of the measurement data. After the starting trigger by SER04A, the addresses and data types of the measurement values are read out of the common memory area in every cycle, the associated measurement values are entered in the telegram buffer and sent.

Via a "LIM" connector, you can enter a limit value for the maximum permissible CPU load, see [↗ Load limitation, page 28](#).

One PDA channel consists of 3 telegrams (communication channels, *nn* = CPU short name):

- One Request telegram (PDAnnCMD)
- One Acknowledge telegram (nnPDAACK)
- One data telegram (nnPDADAT)

A maximum of 32 analog values and 32 digital values can be transferred per PDA channel. Before the transmission, all analog values are converted to the "IEEE-FLOAT" data format.

The following data types are supported:

	STRUC	CFC
1 bit values	B1	BOOL
16 bit values	I2, O2, V2, N2	INT, WORD
32 bit values <sup>2)</sup>	I4, O4, N4	DINT
Floating point values	NF	REAL

<sup>2)</sup> When converting to REAL, the precision of the 32 bit Integer values is reduced to 23 bit.

### 3.4.2 Request with SER05A/-B

#### Note



- Request with SER05A/-B is only possible with ibaFOB-SDexp or ibaFOB-TDC-exp card.
- The blocks are called S5A and S5B under STRUC. All specifications for SER05A and SER05B are also valid for S5A and S5B.

#### Modifications compared to SER04A/-B

- 5 transfer blocks SER05B are connected to each SER05A administrator block, one block per time slice.

Note: You can also connect a SER05B block with a time slice that is slower than initially planned. E.g., also a SER05B can be connected in T2 to CV1 for preventing a threatening overload of the CPU.

- No channel names are configured on the blocks. Only the "CPU short name" (2 signs) is given once on the SER05A.NAM connector. Using these data, the blocks create the following channel names:
  - PDAnnCMX Request telegram
  - nnPDAAcX Acknowledge telegram
  - nnPDADA1 Data telegram time slice 1
  - nnPDADA2 Data telegram time slice 2
  - nnPDADA3 Data telegram time slice 3
  - nnPDADA4 Data telegram time slice 4
  - nnPDADA5 Data telegram time slice 5
- The "LIM" connector for the CPU load limitation, is entered once at the SER05A block and applies to all connected SER05B blocks.
- A maximum of 128 analog and 128 digital signals can be requested per time slice, i.e. a total of 1280 signals.
- The SER05A administrator block does not need to be configured on the slowest time slice. Depending on the project, it might be useful to configure SER05A in T3 or T4 for reducing the times for address book transmission and the start of the measurement (e.g. 100 ... 200 ms).
- Increased data security: No "invalid values" acquired: On SER04, "slower" connectors have been acquired in a faster time, i.e. partially invalid intermediate values have been measured at the output connectors.
- The measured values are not converted, but transferred densely packed in their data type. This reduces the CPU computing time and increases the precision for DINT and DWORD values.
- The user does not need to care for the access optimization concerning the run time. SER05 does this automatically.

- "Slow" connectors can be integrated in the measurement process without an increase in the CPU load: Measurements show that values in T3–T5 only marginally load the CPU.
- Reduction in the CPU load: Even when more than one SER04B is used in T1, SER05B provides for an advantage in CPU load.

One **PDA channel** consists of 7 telegrams (communication channels, *nn* = CPU short name, *i* = time slice 1...5):

- One Request telegram (PDAnnCMX)
- One Acknowledge telegram (nnPDAAACX)
- Five data telegrams (nnPDADA*i*)

A maximum of 128 analog values and 128 digital values are transmitted per time slice over a PDA channel, hence a total of 640 analog values and 640 digital values.

The following data types are supported:

	STRUC	CFC
1 bit values	B1	BOOL
8 bit values	V1	BYTE
16 bit values	I2, O2, V2, N2	INT, WORD
32 bit values	I4, O4, N4, V4	DINT, DWORD
Floating point values	NF, TF	REAL

## 4 Configuration on SIMADYN D/SIMATIC TDC

### 4.1 Service function blocks SER04A/-B

For the measurement, the functional blocks SER04A and SER04B are needed for each PDA channel. You configure these blocks on each processor that *ibaPDA* should have access to.

#### SER04A

Configure the SER04A function block in a slow sampling time (e.g.,  $T_5 > 200$  ms). In any case, you must configure this block more slowly than the SER04B function block.

Description of connectors:

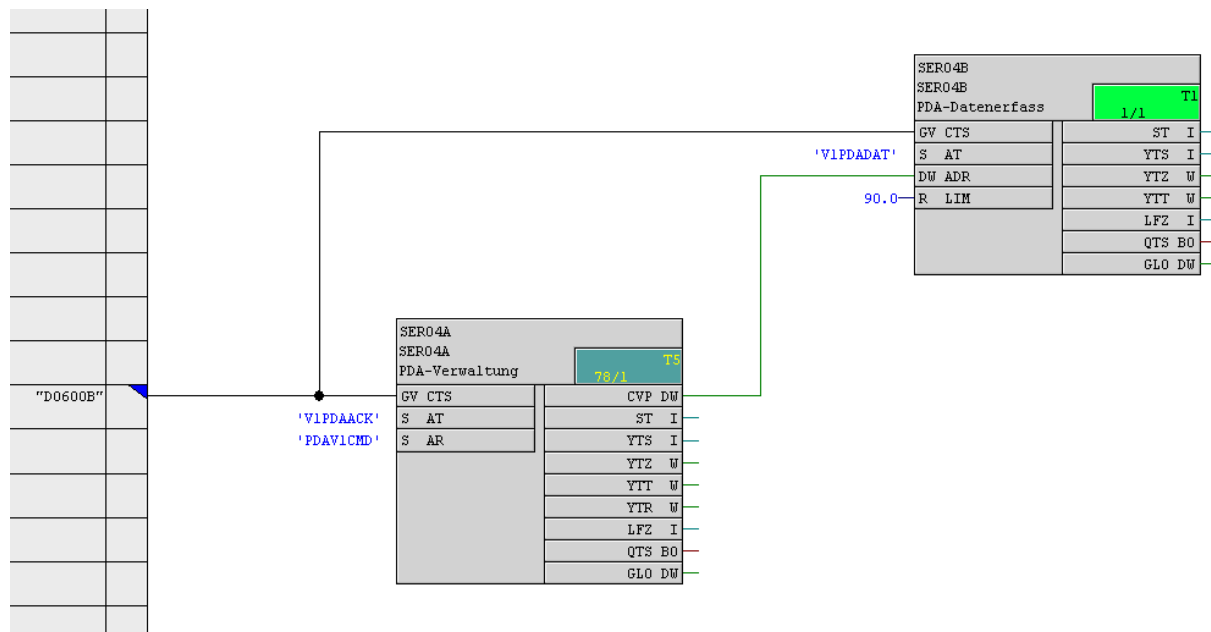
Connector	Description	Value (example)
CTS	Initialization connector for the connection to the component group, the data interface the communication has to run over. (with SD: CS14 or CP53, with TDC: CP52A0).	"D0600B"
AT	Initialization connector for the channel name of the acknowledgement channel in the form " <i>nn</i> PDAACK"	"V1PDAACK"
AR	Initialization connector for the channel name of the request channel in the form "PDA <i>nn</i> CMD"	"PDAV1CMD"
CVP	Connection with connector SER04B.ADR	
ST	Status indication of internal function block status	see ↗ <i>SER04A/SER05A</i> , page 81
YTS	Status indication: error code	
YTZ	Status indication: additional error identification	
YTT	Status indication: transmission channel	
YTR	Status indication: reception channel	
LFZ	Indication of last error status	
QTS	Indication of operational availability	1 = ready for operation

**SER04B**

Configure the SER04B function block in the fastest sampling time.

Description of connectors:

Connector	Description	Value (example)
CTS	Initialization connector for the configured component group, the data interface the communication is to take place over. (with SD: CS14 or CP53, with TDC: CP52A0)	"D0600B"
AT	Initialization connector for the channel name of the data channel in the form "nnPDADAT"	"V1PDADAT"
ADR	Connection with connector SER04A.CVP	
LIM	Limitation of maximum processor load in percent (%)	"90.0", see ↗ <i>Load limitation</i> , page 28
ST	Status indication of internal function block status	see ↗ <i>SER04B/SER05B</i> , page 83
YTS	Status indication: error code	
YTZ	Status indication: additional error identification	
YTT	Status indication: transmission channel	
LFZ	Indication of last error status	
QTS	Indication of operational availability	1 = ready for operation

**Interconnection**

## 4.2 Service function blocks SER05A/-B

For the measurement, the functional blocks SER05A and SER05B are needed on each processor, *ibaPDA* should have access to.

### SER05A

The SER05A function block should be configured in a slow sampling time T3, T4 or T5. A cycle time of 100 – 200 ms is recommended. The slower the sampling time, the longer the booting process for the start of the measurement and the transfer of the address books.

Description of the connectors:

Connector	Description	Value (example)
CTS	Initialization connector for the projected component group, over whose data interface the communication is to take place. (with SD: CS14 or CP53, with TDC: CP52A0).	"D0600B"
NAM	Initialization connector for the short name of the CPU.	"AB"
LIM	Limitation of the CPU in percent. By means of "0.0", the load monitoring can be disabled.	"95", see ↗ <i>Load limitation</i> , page 28
CV1	Connection to SER05B.ADR configured in T1	
CV2	Connection to SER05B.ADR configured in T2	
CV3	Connection to SER05B.ADR configured in T3	
CV4	Connection to SER05B.ADR configured in T4	
CV5	Connection to SER05B.ADR configured in T5	
ST	Status indication internal function block status	See ↗ <i>SER04A/SER05A</i> , page 81
YTS	Status indication error code	
YTZ	Status indication additional error identification	
YTT	Status indication transmission channel	
YTR	Status indication reception channel	
QTS	Indication of operational availability	1 = ready for operation



### 4.3 ibaPDA technosttring (TS)

The ibaPDA technosttring (TS) is a process data channel over which additional non-cyclical alphanumeric data accompanying the measurement can be transmitted to *ibaPDA*. The *ibaPDA* technosttring runs independently of the *ibaPDA* data channels.

The technostoring channel must be configured on the control side. The user has to make sure that the TS is transmitted to *ibaPDA* only in case of a modification or in a slow cycle ( $\geq 1$  second).

The TS channel is queried each second by *ibaPDA*.

### Note



Only 1 technostoring is possible for each rack link. However, beyond the rack link you have the possibility of transmitting further text signals, e.g., via TCP/IP or serial interface.

The channel characteristics are defined as follows:

	<b>ibaPDA technosttring</b>
Channel name	PDA_ASCII
Channel length	512 bytes
Data format	Unformatted alphanumeric character string
Channel mode	Handshake
Application	Transmission of strip attributes, technology and material related data before entry of a new strip in the rolling mill
Cycle	1 second

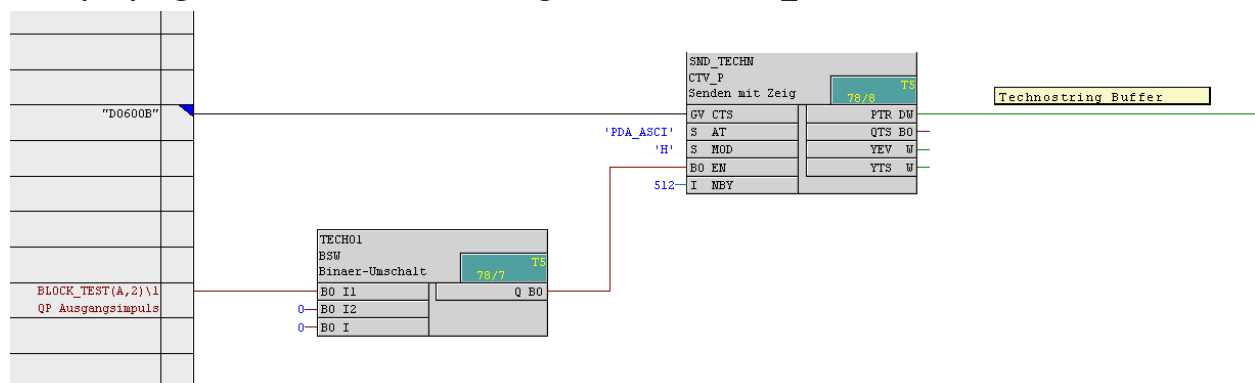
### Note



On *ibaPDA* side, the Technostring message is configured in a special text module *TDC text* or *SD text* under the interface node *ibaFOB-TDCexp* or *ibaFOB-SDexp*.

This module can cut the string into separate text sections. If required, numbers can be interpreted as numeric values.

### Example program "Transmit technostring in CFC" with CTV\_P



## 4.4 Time synchronization on Siemens side

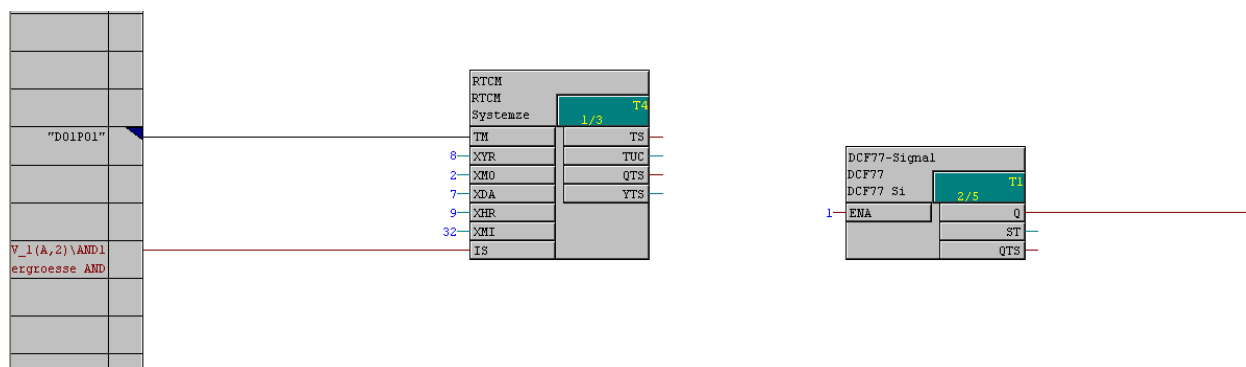
If you synchronize the time of the *ibaPDA* system with the Siemens control, then you have to configure an RTCM function block that reads the system time from a source or generates it itself and distributes it to the connected racks. This function block can only be configured once per rack.

### Procedure for synchronization using DCF77

Synchronization under CFC is done by means of a DCF77 digital signal that is generated by the control.

On the control side, configure a DCF77 function block in the fastest sampling time on one CPU. This function block supplies a digital DCF77 signal that is taken up into the list of digital signals acquired by *ibaPDA* and has to be used in *ibaPDA* for synchronization (see chapter [Time synchronization with DCF77 signal](#), page 52)

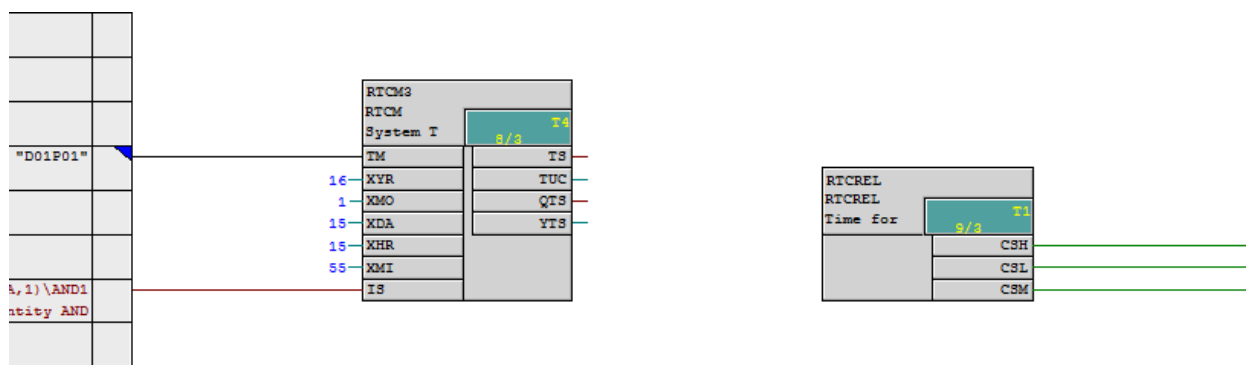
The function block is included in the FBAPDA block library of iba AG.



### Procedure for synchronization using IEC time signal

Alternatively, you can synchronize with an IEC time signal.

On the controller side, configure an RTCREL function block (in CFC) or an RTC005 function block (in STRUC) to get the values for the seconds since 01.01.1988 (CSH, CSL) and the value for the 1/10 millisecond counter.



## 4.5 Address books

Under STRUC, an address book must be generated for each processor after configuration. It contains the information about all configured variables (function packets, function blocks, connectors and time slices). This address book file named `xxxxxx_y.adr` (xxxxxx = BGT name, y = processor number) must be available to the *ibaPDA* system. Either *ibaPDA* must be able to directly access the SIMADYN D project path via network functions, or you must copy this file into an address book directory on the *ibaPDA* computer.

Under CFC, the address books generated by the compiler are not usable for *ibaPDA* because only those variables are included that have the *Operable and observable* checkbox activated when configuring. Therefore the necessary information is read directly from the individual processors via FO connection and stored in files, see [➤ Address books](#), page 40.

## 4.6 Other characteristics

### 4.6.1 Processor load

#### Note



Note that the processor load increases due to an active PDA channel.

For the sporadic load due to transmission of the address book, the SER04A or SER05A function block requires a computing time of max. ca. 10 ms to 20 ms per cycle. Therefore, choose the time slice in which the block is configured so that sufficient processing time remains for the sporadic T5 task "Transmit address book".

If this time is not available, then the overload indication "E" is displayed during address book transmission. Since this overload is only sporadic and is caused in T5 and therefore has no influence on T1-T4 processing, you can ignore this display and delete it again by acknowledgement.

The cyclical transmission operation by the SER04B or SER05B function block requires computing time per transmission cycle for assignment of the measurement values in the local data buffer and for transmission of the data into the rack link.

The measurements with **SER04** showed the following results:

	Measurement 1	Measurement 1	Measurement 3
System	SIMATIC TDC	SIMATIC TDC	SIMADYN D
Software	PCS7 V7.1, D7-Sys V7.1	PCS7 V5.2, D7-Sys V5.2	STRUC V4.2.5
CPU	<b>CPU551</b>	<b>CPU550</b>	<b>PM5</b>
ibaFOB-SD (B5)	-	-	CS13
ibaFOB-TDC (B5)	CP52A0 (GDM)	CP52A0 (GDM)	-
SER04A/B	100204V001	020417V007	990603V420
Transfer time	$(38 + n * 2.7) \mu\text{s}$	$(44 + n * 3.7) \mu\text{s}$	$(80 + n * 6) \mu\text{s}$

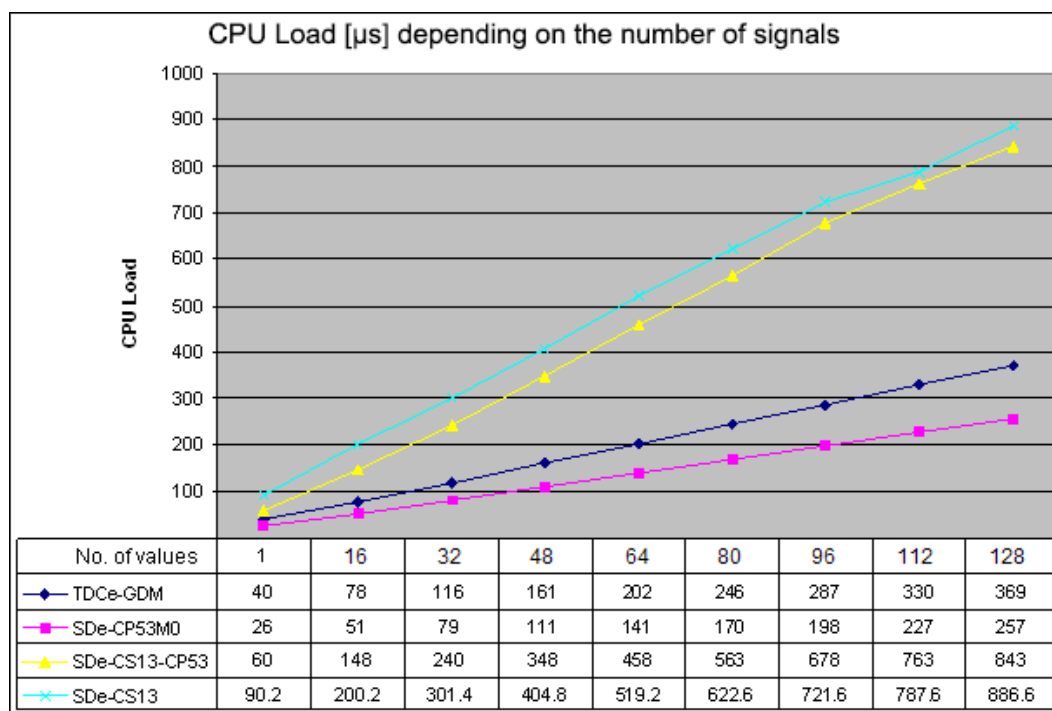
	Measurement 1	Measurement 1	Measurement 3
full PDA channel (32 A + 32D)	appr. 125 µs	appr. 160 µs	appr. 280 µs

with n = number DWORD (analog values (FLOAT) and digital values), max. 33

The measurements with SER05 delivered the following results:

	Measurement 1	Measurement 2	Measurement 3	Measurement 4
System	SIMATIC TDC	SIMATIC TDC	SIMATIC TDC	SIMADYN D
Software	PCS7 V7.1, D7-Sys V7.1	PCS7 V7.1, D7-Sys V7.1	PCS7 V7.1 D7-Sys V7.1	PCS7 V7.1 D7-Sys V7.1
CPU	CPU551	CPU551	CPU551	PM5
ibaFOB-TDCexp	CP52A0 (GDM)	-	-	-
ibaFOB-SDexp	-	CP53M0 (M)	CS13-CP53M0	CS13
SER05A/-B	140129V001	140129V001	140129V001	140129V001
Transfer time	$(38 + n \cdot 2.7) \mu s$	$(26 + n \cdot 1.9) \mu s$	$(60 + n \cdot 6.4) \mu s$	$(90 + n \cdot 7.0) \mu s$
Full channel of a time slice (128 A+ 128 D)	appr. 385 µs	appr. 268 µs	appr. 883 µs	appr. 920 µs

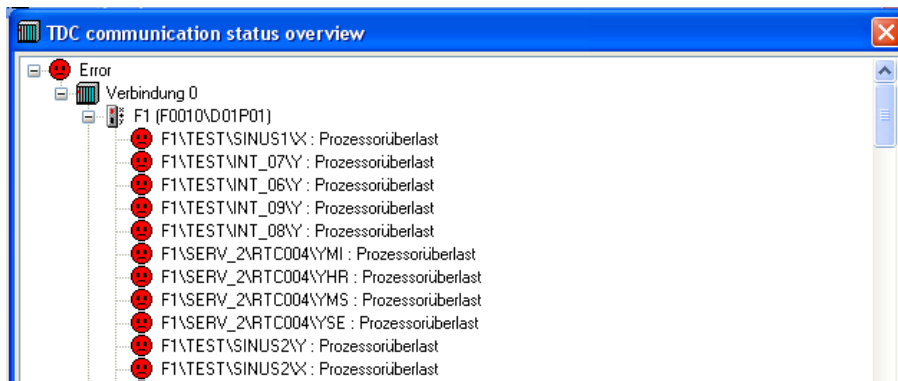
with n = number DWORD (analog values and digital values packed), max. 132



### 4.6.2 Load limitation

In order to prevent a CPU overload due to the increase in computing time when making a request, the blocks need to have a load limitation (connector LIM).

After the start of acquisition, the function block follows a kind of run-up ramp when taking only parts of the requested values per cycle into its processing. During this phase the required computing time is checked. If the load exceeds the value set at the LIM connector, then the measurement values that have not yet been processed are rejected and a "Processor overload" error message appears in *ibaPDA* (see figure).



Possible solution for "Processor overload" error message:

- Reduce the number of the requested values in *ibaPDA*.
- Increase the sampling time base (*ibaPDA* base measuring rate or the PDA channel time bases).
- Increase the time slice for the SER04B in the TDC system.

#### Note



The load is checked only at the start of acquisition. The SER04 function block does not detect a load increase due to other acyclic processes during ongoing measurement.

Due to unfavorable temporal relations overload errors can occur despite monitoring. The load increase in high-priority tasks has repercussions on the utilization of low-priority tasks. But the latter lead to a load increase only after expiration of their task cycle. But with slow low-priority task times the run-up of the SER04B function block is already concluded at this point.

#### Note



The load limitation does not work when the SIMATIC TDC processor boards CPU551 or CPU550 work with a "synchronized" base clock cycle. In this case you should switch off the load limitation by entering "LIM=0".

In the CPU 555, this bug is fixed.

### 4.6.3 Transmission cycle

The **SER04B** works in the configured sampling time (primarily T1).

Starting with *ibaPDA* v6.24.1 a time base can be specified for each PDA channel in order to optimize the load of the CPUs, see chapter [↗ Creating PDA channels](#), page 35.

When requesting with **SER05B**, every measured value is sent in the cycle it has been configured.

The following applies for **both blocks** (SER04B and SER05B):

If *ibaPDA* requests all data in a slower cycle, the block reduces its own transmission cycle (for SER05B see [↗ Timing tab](#), page 66). This means the block transmits no longer in the projected cycle time, but rather in the cycle in which *ibaPDA* reads the data (rounded to a multiple of its own cycle). This can reduce the processor.

### 4.6.4 Storage requirement

One SER04A/-B PDA channel needs appr. 1800 Bytes in the communication storage of the rack coupling link (CS14, CP53 or GDM).

One SER05A/-B PDA channel needs appr. 7500 Bytes in the communication storage of the rack coupling link.

The address book telegram requires once 16516 Bytes.

## 5 Configuration and engineering ibaPDA

The following sequence is recommended for the configuration:

1. *ibaPDA* standard settings
2. Request presettings
3. Automatic detection of connected CPUs
4. Load address books
5. Setting up request modules
6. Selection of signals

With subsequent changes in configuration, you only have to update the address books of the changed CPUs.

Configuration of the PDA channels is automated as extensively as possible. The following "special cases" are also covered.

- Several PDA channels are projected on one processor.
- One *ibaPDA* has several "parallel" links to a rack link.
- One *ibaPDA* several links to different rack links.
- Several *ibaPDAs* each have a link to the same rack link.

### 5.1 ibaPDA standard settings

---

#### Other documentation



For settings of time base, interrupt, etc. see *ibaPDA* manual.

---

### 5.2 Hardware interfaces ibaFOB-SD/ibaFOB-TDC

#### Procedure

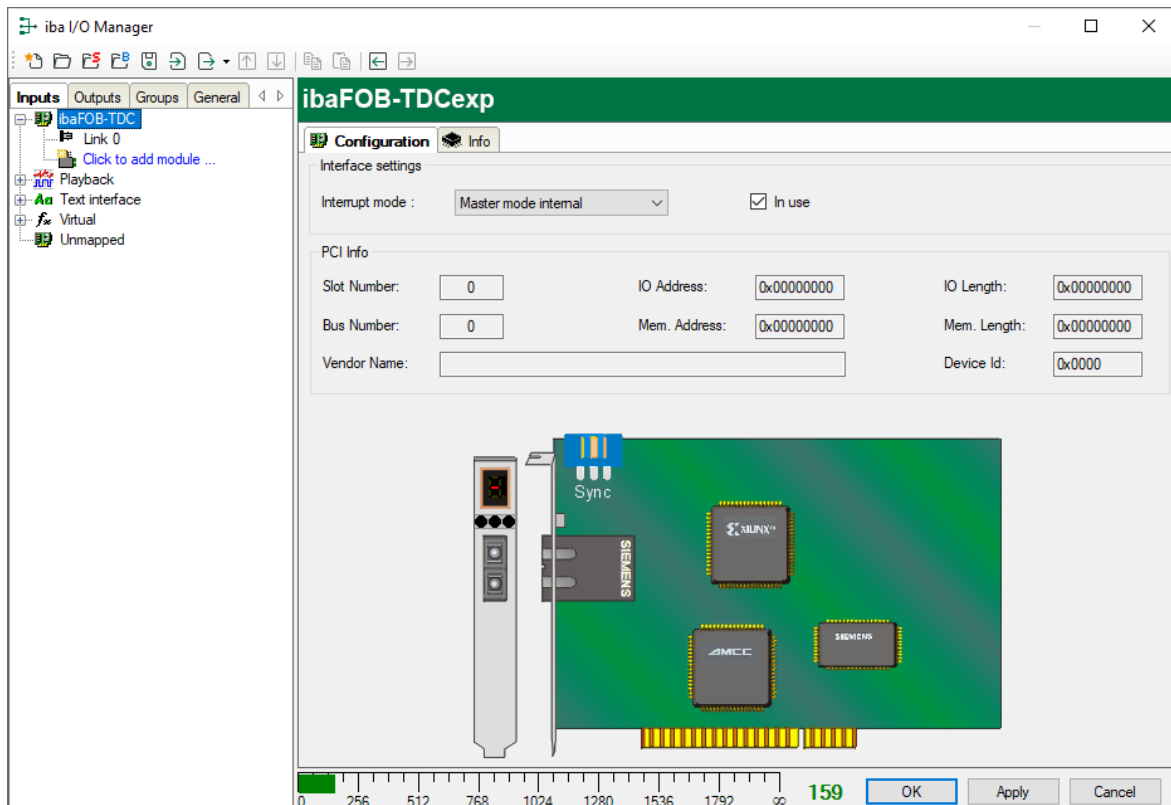
1. Open the I/O Manager in the *ibaPDA* client.
2. Click on <New configuration>.  
*ibaPDA* detects both the installed components as well as the licensed data interfaces and depicts them in the tree structure.  
If you select an iba component in the interface tree, the component is schematically displayed.

Because the *ibaFOB-SD* and *ibaFOB-TDC* cards are functionally not different, only the *ibaFOB-TDC* card is described in the following. This description also applies to the *ibaFOB-SD* card, unless otherwise expressly mentioned.

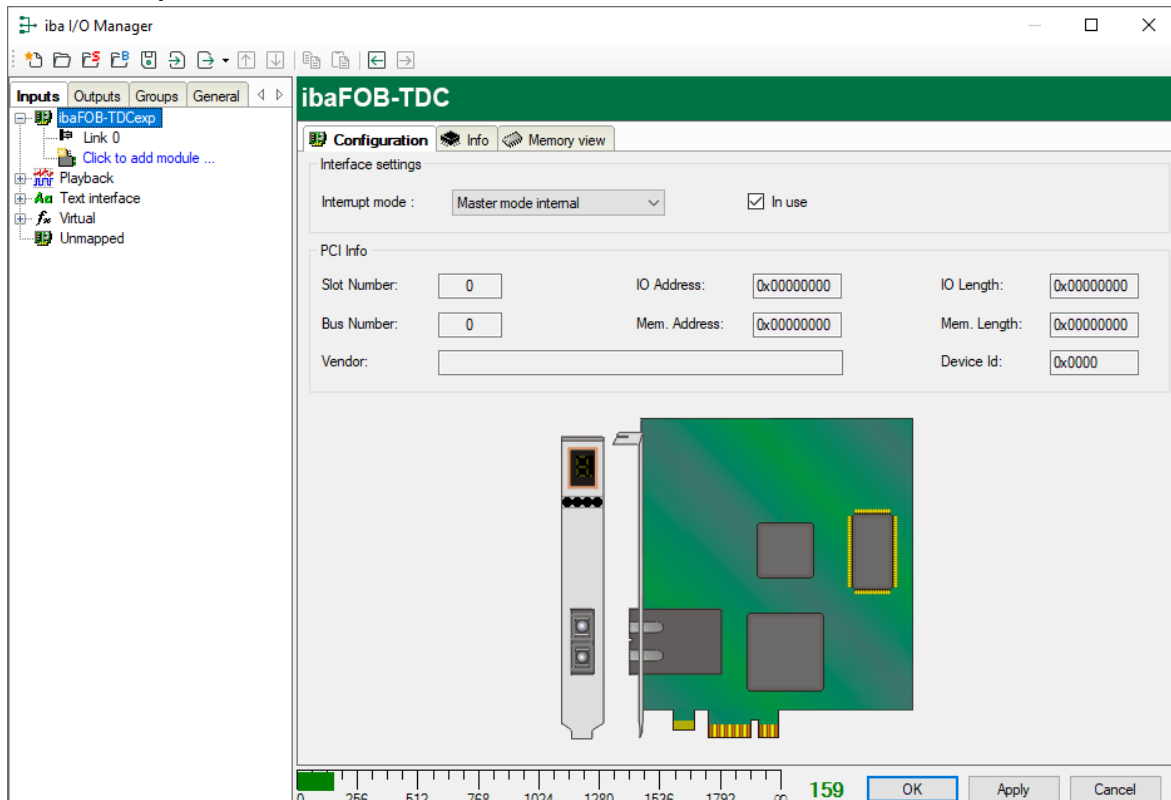
The same applies to the *ibaFOB SDexp* and *ibaFOB TDCexp* card.

## 5.2.1 Schematic representations of the iba cards

### ibaFOB-TDC card



### ibaFOB-TDCexp card



You get the following information in the dialog window:

### Interface settings

Set the interrupt mode and tick the *In use* checkbox.

#### Note



Set the ibaFOB-SD or ibaFOB-TDC as interrupt master if you use only ibaFOB-ni-S or ibaComL2B-n/8 cards.

### PCI Info

You can find information about the PCI interface in the *PCI info* area of the dialog. It is of interest only to developers.

### ibaFOB-SD-/TDC card display

The graphic depiction of the card is dynamic, i.e., the 7-segment display with the card number and the LEDs for the connection status reflect the same status that can also be seen on the card itself.

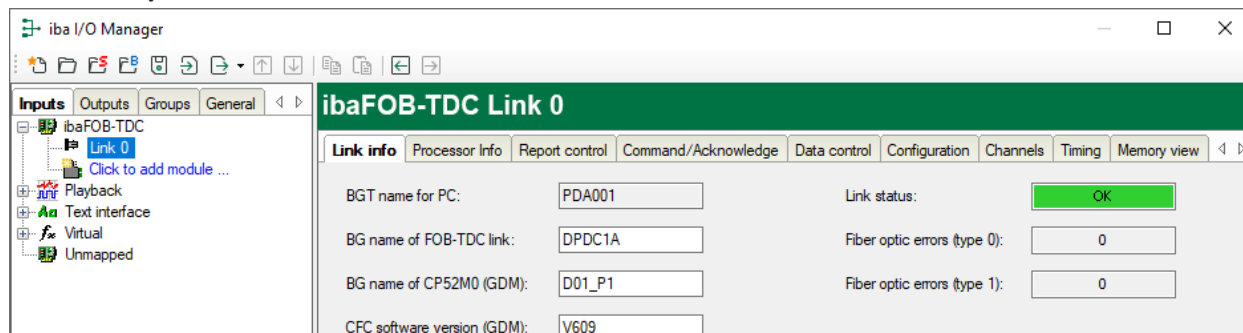
In the following table, you see an overview of the displays and their meanings:

	Values/states	Meaning
7-segment display	0 ... 3	Card number of this type (0...3)→ OK
	-	Card is not initialized → error
	8	Card does not start up (μ-processor stops) → error
Point in the 7 segment display	On	Card set-up: "Interrupt master/internal"
	Off	Card set-up: "Interrupt slave"
Green LED (RUN)	Flashes	Voltage applied, card is working
	Off or on	μ-processor stops → error
Yellow LED (LINK)	On	Connection to the coupling partner exists → ok
	Off	No input signal
	Flashes	Connection ok, but partner not ready
White LED (only with -exp)	On	Data transfer active
	Off	No data transfer
Red LED (ERR)	Off	Normal state
	On	Internal component error
	Flashes	FPGA Factory Rescue Mode

## 5.2.2 Link level

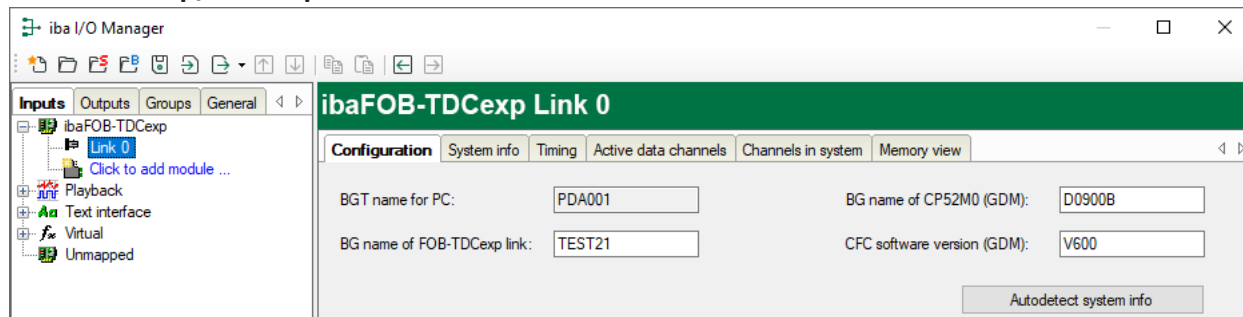
If you mark the *Link 0* node under the card in the signal tree in the *Inputs* tab, then the tabs for diagnosis of the communication connection appear on the right side.

### ibaFOB-SD/-TDC



For more information, please see the description of the diagnosis in [ibaFOB-SD/ibaFOB-TDC diagnostics](#), page 54.

### ibaFOB-SDexp/-TDCexp

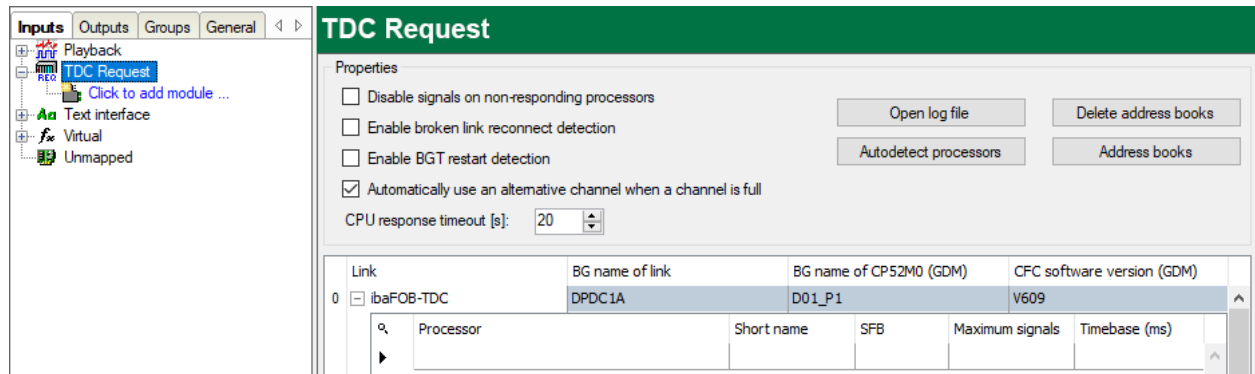


For more information, please see the description of the diagnosis in [ibaFOB-SDexp/ibaFOB-TDCexp diagnostics](#), page 64.

## 5.3 TDC Request/Simadyn Request data interface

The *Simadyn Request* data interface appears in the tree structure if the "ibaPDA-Request-SD" or "ibaPDA-Request-TDC" license is activated in the license container and an *ibaFOB-SD/ibaFOB-SDexp* card is installed.

The *TDC Request* data interface is listed in the tree structure if the "ibaPDA-Request-TDC" license is activated in the license container and an *ibaFOB-TDC/ibaFOB-TDCexp* card is installed.



### 5.3.1 Settings

This interface contains the following options for reacting to communication errors:

#### Disable signals on non-responding processors

If this option is enabled, then at the start of acquisition the measurement channels of the non-responding CPUs are temporarily deactivated. At the next start of acquisition they are then active again. This occurs if, e.g., not all frames are switched on at the *ibaPDA* start.

#### Enable broken link reconnect detection

When the FO link is broken, *ibaPDA* keeps running with the last received values frozen. With this option, you can force an *ibaPDA* restart after the link is reconnected.

#### Enable BGT restart detection

If this option is enabled, then even during measurement it is checked whether missing CPUs become available again. If that is the case, then the measurement is stopped, a new request is executed and the measurement is restarted.

#### Automatically use an alternative channel when a channel is full

Here, you can disable the use of alternative channels. By default this option is enabled in order to make *ibaPDA* backwards compatible.

#### CPU response timeout

Here you can specify a value for the timeout in seconds for reaction to commands to the *ibaFOB-SD/ibaFOB-TDC* or assignments to the controller. If the time set here is exceeded, *ibaPDA* declares the controller as not accessible or not responsive. The response time depends on the number of CPUs and the time slices of the service function blocks.

## Recommended settings

- Disable signals on non-responding processors: yes
- Enable broken link reconnect detection: yes
- Enable BGT restart detection: yes
- Automatically use an alternative channel when a channel is full: yes
- Restart acquisition on non-recoverable error (*General* tab – *Settings* node): yes

These settings leads to the following behavior:

- Failure of individual CPUs does normally not occur, but only failure of entire racks (e.g., when reloading software).
- In this case the recording continues. Switching on again leads to restart of acquisition. In the process, the CPUs of this rack are taken up again.
- In case the entire connection fails *ibaPDA* keeps running. On restart of *ibaPDA*, the missing signals are deactivated. As soon as the connection is OK again the measurement is restarted. The previously deactivated signals are then reactivated.

## 5.3.2 Creating PDA channels

With the <Autodetect processors> button the rack link is searched for PDA channels. The connection data and all PDA channels found are entered in the connection list.

Link	BG name of link	BG name of CP52M0 (GDM)	CFC software version (GDM)
0 + ibaFOB-TDC	DPDA1A	D0600B	V600
1 - ibaFOB-TDCexp	DPDA1B	D0600B	V600

Processor	Short name	SFB	Maximum signals	Timebase (ms)
0 G0010\D01P01	G1	SER04	32	0
1 G0010\D01P02	G2	SER04	32	0
2 G0010\D01P03	G3	SER04	32	0
*				

The connection list contains the following data:

- Connection data necessary for defining ibaPDA as a "Siemens rack" in the rack link
- A list of all PDA channels with the associated parameters

## Connection data

### Link

Name of data interface (*ibaFOB-SD/ibaFOB-TDC* component)

### BG name of link

Name with which *ibaPDA* logs on to the rack link component.

Default setting: 5 digits of the license number and the link number

**Note**

The connection name must be unambiguous within the rack link. If two connections to a rack link are present in an *ibaPDA* system, or two or more *ibaPDA* systems are connected to a rack link, then you have to enter different connection names. Make sure that the name corresponds to the Siemens guideline, i.e. 6 characters A–Z, 09 and \_.

**BG name of the rack link component**

Configured name of the CS14, CP53 or GDM memory component (CP52M0). This name is entered upon autodetection of the CPUs by *ibaPDA*.

**STRUC/CFC software**

Software version of SIMATIC TDC, or SIMADYN D software. The version is entered upon autodetection of the CPUs by *ibaPDA*. *ibaPDA* uses this version designation for differentiating between CFC (V6xx) and STRUC soft-ware (V4xx).

**PDA channel table**

For each connection, the following information is listed in in the table:

**Processor**

Configured BGT name \ BG name of the CPU

**Short name**

Name of the PDA channel, see also ↗ *Functional principle*, page 16

**SFB:**

Type of the service component on the SD-CPU or TDC-CPU (SER04 or SER05)

**Maximum signals**

Maximum number of analog signals of the channel (version: 32 or 128)

**Time base (ms)**

For each PDA channel a time base can be set. It is thus possible to divide the measurement into fast and slow signals. With time base 0 the *ibaPDA* basic sample time is taken as the time base. You can find notes concerning the temporal sequence in the following chapters ↗ *Temporal sequence of the measurement for SER04*, page 38.

**Note**

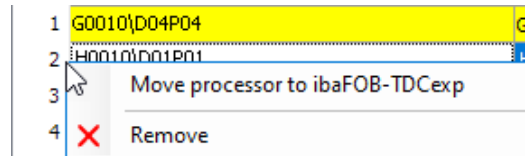
In this list only those processors appear, the *ibaPDA* system has exclusive access to. If multiple *ibaPDA* systems are connected to one rack link, you need to ensure that only one *ibaPDA* has access to each PDA channel. This is why you need to delete the PDA channels that are not used after the automatic CPU detection.

Select the PDA channel and press the <Del> key to delete the channel.

### Moving the processor to a different connection

For evenly distributing the load on the *ibaFOB-TDC* cards when there are parallel connections, move the channels to another connection:

1. Move the cursor to the number of the line, and right-click to open the context menu.
2. Using the *Move processor to ibaFOB-TDCexp* context menu entry, transfer this channel to another link.



### Viewing the channels

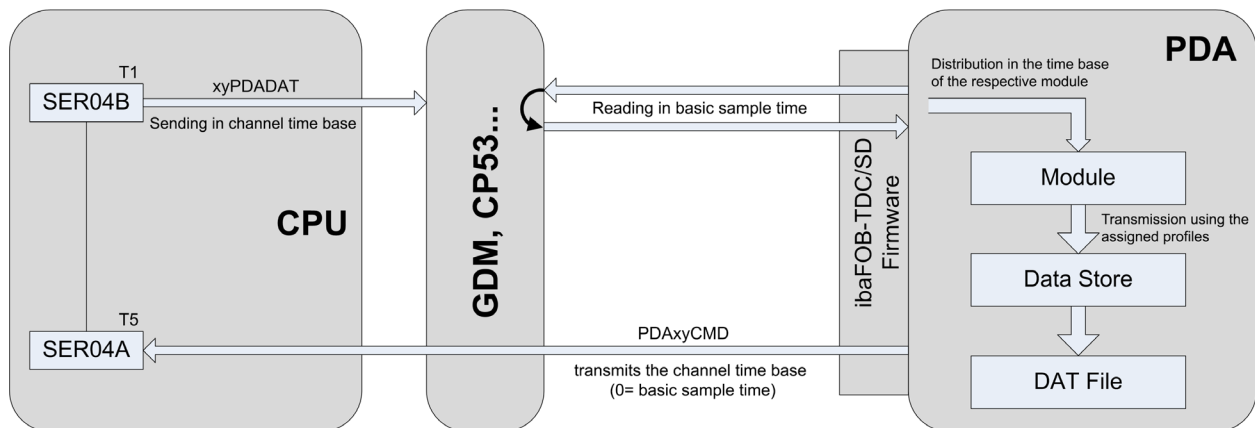
To see all channels, there are two scroll bars on the right.

You can open each connection via the <+> button to display the list of associated channels.

Link	BG name of link
0 + ibaFOB-TDC	DPDA 1A
1 - ibaFOB-TDCexp	DPDA 1B
Prozessor	
0	G0010\DP01P01
1	G0010\DP01P02

### 5.3.3 Temporal sequence of the measurement for SER04

The significance of and the relationship between the channel time base, the basic sample time and the time base of the individual modules is depicted in the following overview:



*ibaPDA* transmits the set channel time base (0 = basic sample time) to the CPU (SER04A function block) for each channel via the PDAxyCMD telegram.

The SER04B function block (called in time slice T1) takes over the actual transmission of data via the xyPDADAT telegram. The coupling partner is one of the interface components presented in section [Overview](#), page 9. Transmission of SER04B is done in the frequency of the specified channel time base (if applicable, rounded to a multiple of T1). If the channel time base is faster than T1, then SER04B transmits in each T1 cycle.

From the interface component in the basic sample time, *ibaPDA* reads the data of the required channels (depending on projected signals in the I/O Manager). Processing of the reading assignments is done by the *ibaFOB-TDC/ibaFOB-SD* card. The capacity utilization of the *ibaFOB-TDC/ibaFOB-SD* card depends on the number of channels and the basic sample time, see also [Timing tab](#), page 61.

The signals thus received are distributed to the various modules using the respective module time base.

In the next step, the data are transmitted to the data stores and then to the data files (DAT files) using the projected/assigned recording profiles.

#### Note



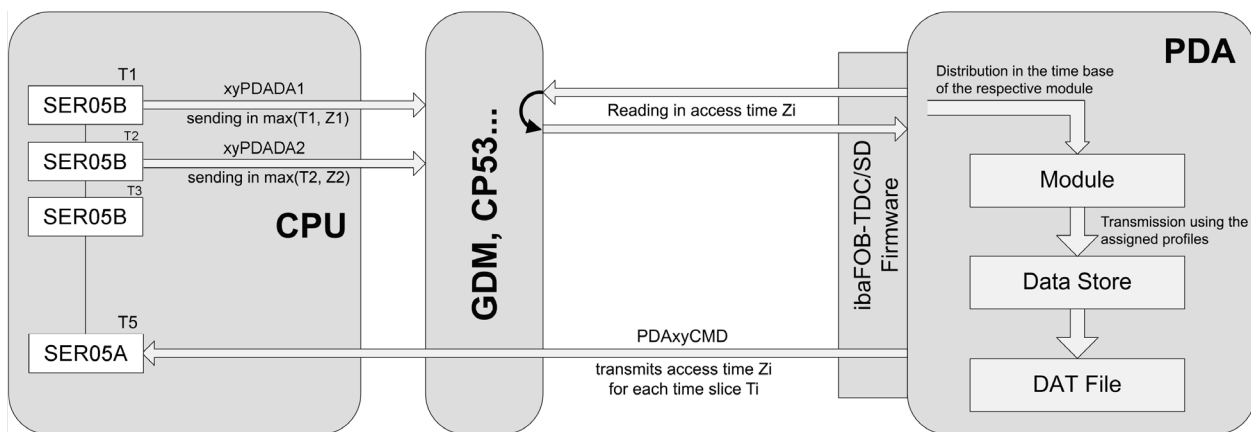
A maximum of 50 channels per connection are possible. In each channel, you can measure a maximum of 32 analog values and 32 digital values. If this is not sufficient, project another PDA channel on this CPU.

### 5.3.4 Temporal sequence of the measurement for SER05

The following time values are used:

$T_{pda}$	<i>ibaPDA</i> sampling time base
$T_{mod}$	Simadyn-D Request/TDC Request module time base
$T_i$	Configured time class of SER05B in SIMATIC TDC/SIMADYN D
$Z_i$	Read cycle time for each time class 1 to 5
$T_s$	Storage cycle, assigned to each measured value in the data storage profile

The connections between these time values are shown in the following overview:



SER05A transfers the associated time slice  $T_i$  ( $T_1 \dots T_5$ ) for each block in the address book.

In case of a data request, *ibaPDA* passes the access time  $Z_i$  to the CPU (block SER05A).  $Z_i$  is the smallest configured time slice of all CPUs and is valid for all time classes. You can configure the access time depending on the requirements – even individually for each time slice ( $T_2$  to  $T_5$ ), see [Timing tab](#), page 66.

The SER05B function block that is called in every time slice  $T_i$  takes over the transmission of the data. The coupling memory is one of the interface components presented in section [Overview](#), page 9. SER05B sends in the cycle of the access time  $Z_i$  (in some cases rounded up to a multiple of  $T_i$ ), or, if  $Z_i$  is faster than  $T_i$ , in every  $T_i$  cycle.

*ibaPDA* reads the data of the required channels from the interface component in the basic reading cycle  $Z_i$ . Processing of the reading assignments is done by the *ibaFOB-TDC/ibaFOB-SD* card. The capacity utilization of the *ibaFOB-TDC/ibaFOB-SD* card depends on the number of channels and the basic sample time, see [Timing tab](#), page 66.

The signals thus received are distributed to the various modules using the respective module time base  $T_{mod}$ .

In the next step, the data are transmitted to the data stores and then to the data files (DAT files) using the configured/assigned recording profiles.

#### Note



A maximum of 50 channels per connection are possible. In each channel, you can measure a maximum of 640 analog values and 640 digital values (128 per time slice). If this is not sufficient, configure another PDA channel on this CPU.

### 5.3.5 Address books

The address books are the database for the symbol browser. An address book contains the names of all charts, function blocks and connections.

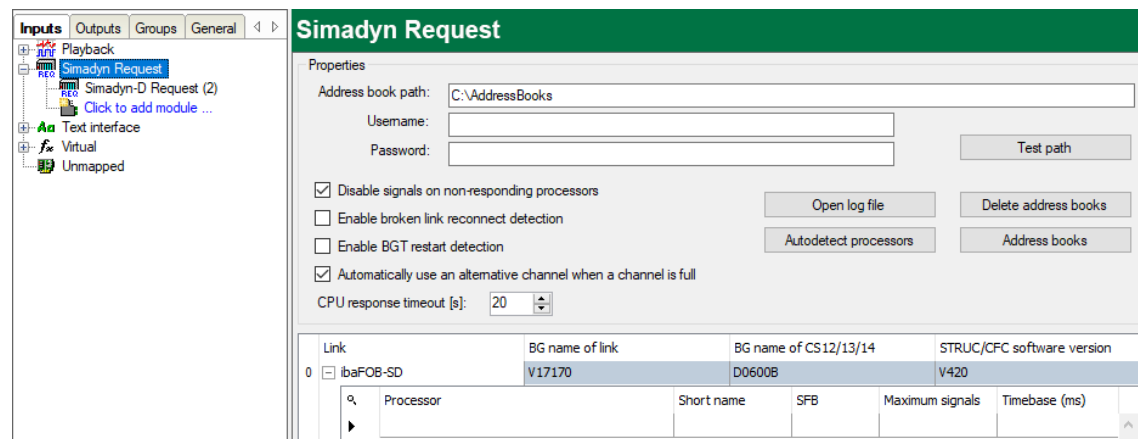
See also the explanations in chapter [➤ Address books](#), page 26.

#### Address book import under STRUC

The address book files generated by the STRUC editor must be imported in the following manner:

1. In the *Simadyn request* interface, enter the path and user name/password or the folder where the address book files are stored.

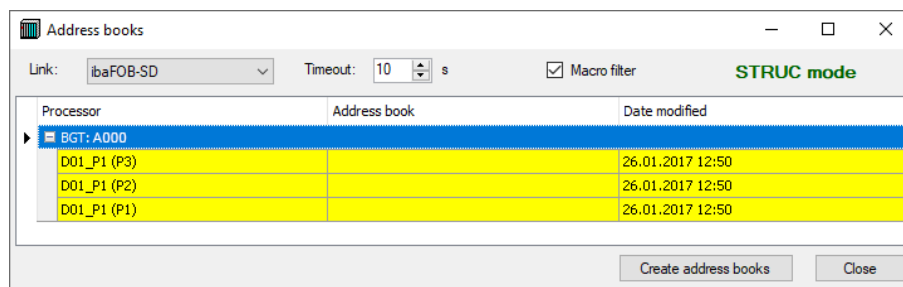
Click on <Test path>.



2. If the path is correct, click on <Address books>.

→ The *Address books* dialog opens.

The channels of the first connection are displayed, sorted by BGT name.



3. To see the channels of the other connections, select the connection in the *Link* drop-down list.
4. Change the settings as needed:
  - *Link*: If you have several parallel connections, you can select the connection here.
  - *Timeout*: Timeout for the response to the *Create address books* command. Default is 20 seconds. With very large address books, you might have to increase the time.
  - *Macro filter*: No significance here, see [➤ Macros](#), page 45
5. Click on <Create address books>.

- The address books are imported.
- The *Address book* column contains the name of the address book. The *Date modified* column contains the date of the address book file.
- The background color of the channel table becomes white.

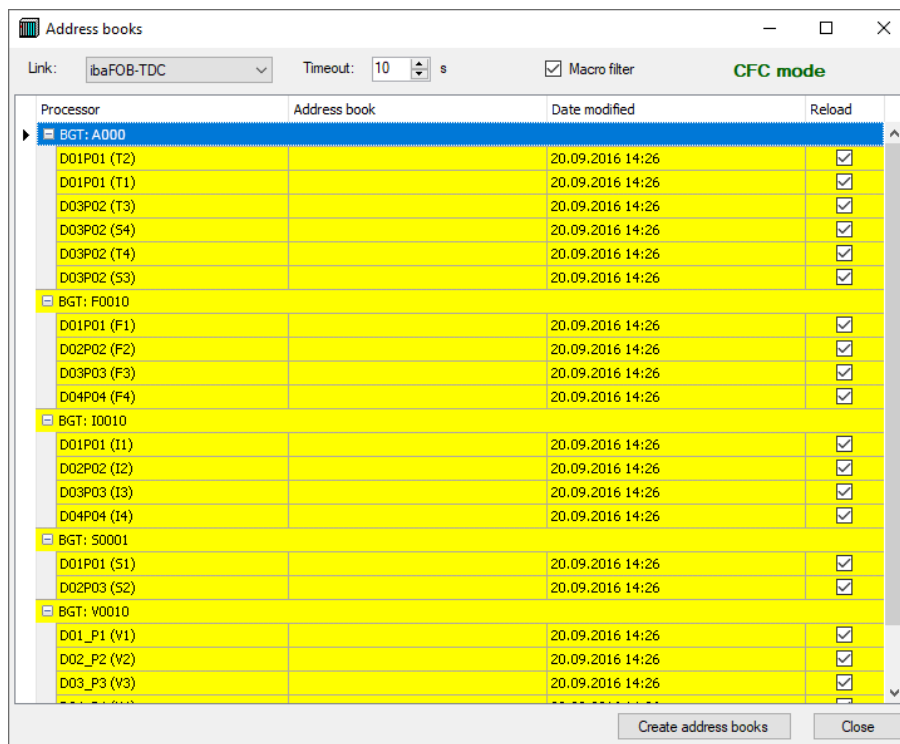
### Address book import under CFC

The CFC address books are created by *ibaPDA*. Proceed as follows:

1. In the *TDC request* interface, click on <Address books>.

→ The *Address books* dialog opens.

The channels of the first connection are displayed, sorted by BGT name.



2. To see the channels of the other connections, select the connection in the *Link* drop-down list.
3. Change the settings as needed:
  - *Link*: If you have several parallel connections, you can select the connection here.
  - *Timeout*: Timeout for the response to the *Create address books* command. Default is 20 seconds. With very large address books, you might have to increase the time.
  - *Macro filter*: If the macro filter is activated, then the connectors within macros are not transmitted. This leads to a shorter transmission time and smaller address books, see [Macros](#), page 45.
4. In the *Reload* column, select the PDA channels whose address books you want to load or reload.  
If no address book is available yet (yellow entry), then this entry is activated by default.

5. Click on <Create address books>.

- The address books are loaded in sequence by the CPU and deposited in files with the name `xxx_yyy.adr` (xxx = BGT name, yyy = component name). In some circumstances, the process can take several minutes.
- The *Address book* column contains the name of the address book. The *Date modified* column contains the date of the address book file.
- The background color of the channel table becomes white.

---

**Note**

All PDA channels of a CPU use the same address book. Therefore, only one address book is created per CPU. If a Request Agent SER05 is configured on this CPU, an address book is queried by this agent.

---

**Note**

In contrast to COROS or WinCC, *ibaPDA* uses the address book only for support in selecting the variables. Internal addresses are not calculated. Therefore *ibaPDA* can basically work with an address book that is not up to date. Consider the marginal conditions explained in the following when using address books that are not up to date.

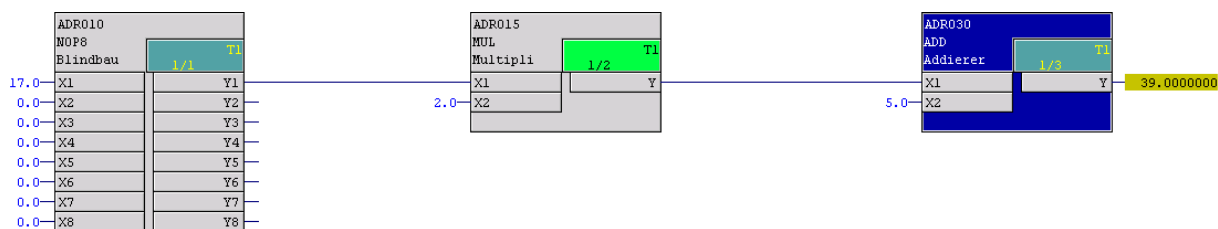
---

## Shifted function block names

Often changes in CFC are executed whereby already existing function block names are transferred to other function blocks, e.g., because a CFC function block is inserted in a series of function blocks and a specified numbering system should be retained for the function blocks. Thus it can happen that a connector address projected in the I/O Manager still exists, but at a completely different CFC function block and with another technological meaning.

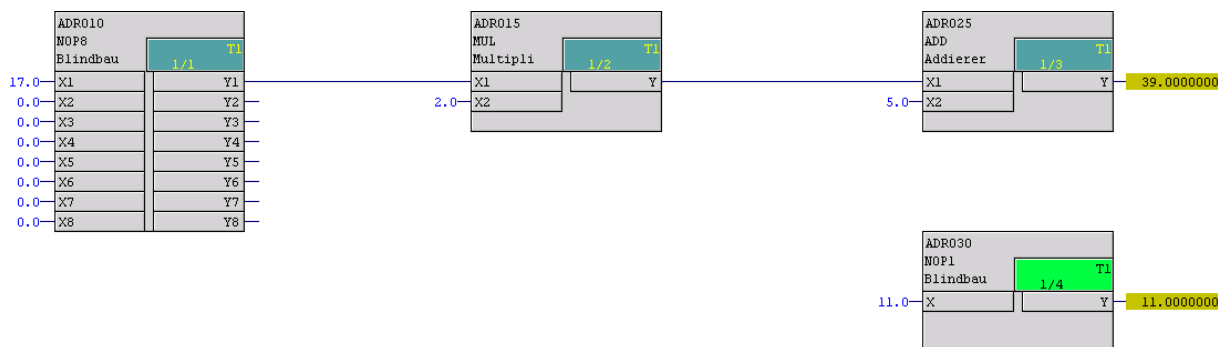
### Example

The CFC program with the following function blocks is running on the CPU. *ibaPDA* records the ADR030/Y connector (value = 39.0) with a current address book.



The ADR030 function block is renamed in a modification step and another function block which is newly inserted receives the function block number ADR030 (see below). The modifications are compiled and loaded.

*ibaPDA* now records the "new" ADR030/Y connector (value = 11.0). Make sure to configure the corresponding adjustments in the I/O configuration.

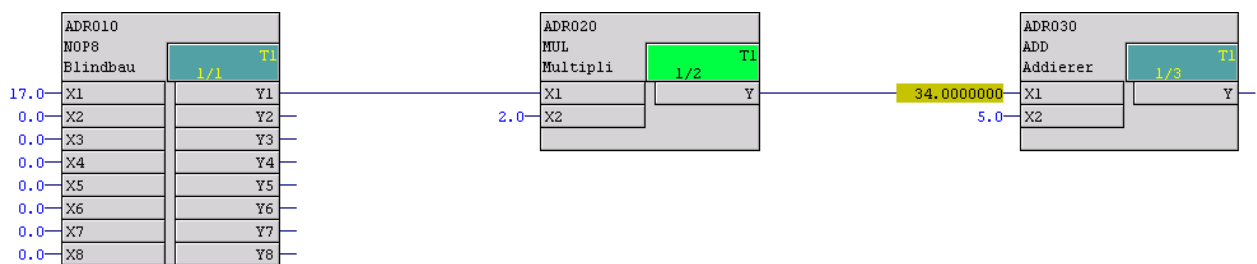


## Online modifications

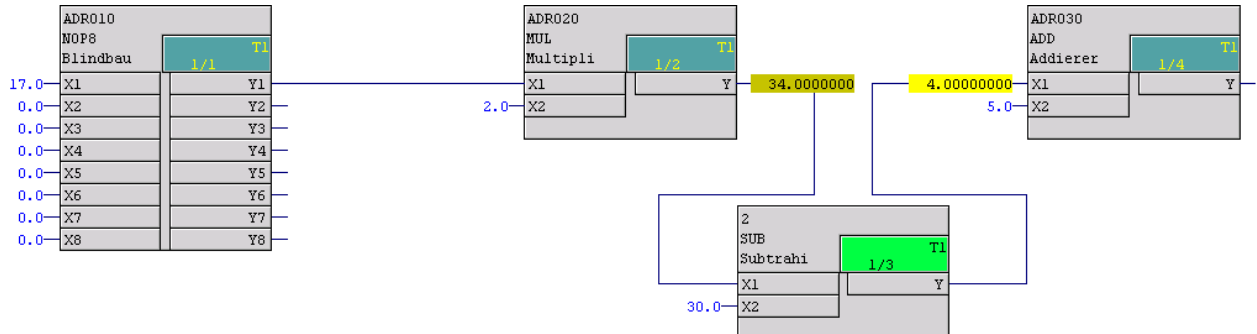
With online modifications (TDC), TDC-internal modifications of storage addresses for connectors result. Data which a CFC function block outputs (output connectors) are stored under storage addresses of the CPU. With online modifications, these addresses are modified during the running time of the program.

### Example

The CFC program with the following function blocks is running on the CPU. *ibaPDA* records the ADR030/X1 connector (value = 34.0) with a current address book. The value of connector ADR030/X1 is under the storage address in which ADR020/Y is stored.



The function block of the type SUB is inserted online. The system assigns the name of the function block (here "2"). The function block is deposited in the CPU under this name. You can only assign an individual name offline and load into the CPU by compiling/loading. The connection between ADR020/Y and ADR030/X1 is modified online.



The value of connector ADR030/X1 is now in the storage address of connector 2/Y. But the old address book in *ibaPDA* still accesses the address of connector ADR020/Y. As long as the address book is not updated, *ibaPDA* records the value 34.0 instead of 4.0.

<sup>3)</sup> After the definition of the communication protocol, the first partner that logs into the communication memory sets up the channel. This is the case if *ibaPDA* starts acquisition before the CPUs have started up on the Siemens side. For assignment of the address books to the PDA channel names, *ibaPDA* needs information from the communication channel that has been set up, information which is present only if these channels were set up by the SER04A/B service function blocks.

### 5.3.6 Manual assignment of address books

In some cases<sup>3)</sup> automatic assignment of address books to the PDA channels does not work properly:

After execution of the *Autodetect processors* function, BGT or BG names are not displayed in the channel table, but only the channel names.

Link	BG name of link	BG name of CP52M0 (GDM)	CFC software version (GDM)
0 ibaFOB-TDC	DPDC1A	D01_P1	V609

Processor	Short name	SFB	Maximum signals	Timebase (ms)
0 \	C1	SER04	32	0
1 \	B8	SER04	32	0
2 \	B7	SER04	32	0

In this case, after loading the address books, you can manually assign the PDA channel names to the address books.

Click a field in the Processor column and open the drop-down list. Select a CPU from this list.

Link	BG name of link	BG name of CP52M0 (GDM)	CFC software version (GDM)
0 ibaFOB-TDC	DPDC1A	D01_P1	V609

Processor	Short name	SFB	Maximum signals	Timebase (ms)
0 \	C1	SER04	32	0
1 A000	B8	SER04	32	0
2 D01P01	B7	SER04	32	0
3 B0010	C4	SER04	32	0
4 D01P01	C3	SER04	32	0
5 D02P02	C2	SER04	32	0
6 D03P03	B6	SER04	32	0
7 D04P04	B2	SER04	32	0
8 D05P05	B1	SER04	32	0
9 D06P06	AA	SER04	32	0
10 D07P07	B5	SER04	32	0
11 D08P08	Ba	SER04	32	0
12 C0010	B3	SER04	32	0
13 D01P01	AB	SER04	32	0
14 D02P02	D1	SER04	32	0
15 D03P03				
16 D04P04				
17 Löschen				

After you assigned the short name to the processor, the row is then shown with a white background.

### 5.3.7 Macros

There are also macros available in CFC. Macros are charts, which have been translated to blocks (not nested charts!).

Blocks within the macros are identified by "blockname#nnn".

In order to optimize the address book size, a macro filter can be used. The macro filter filters out all blocks whose name include an #-sign.

This filter is active by default.

But the #-sign may also be used in block names in the chart level. Then, blocks do not appear in the address book. In this case, you have to deactivate the macro filter.

Address books

Link: ibaFOB-TDC

Timeout: 10 s

☐ Macro filter

CFC mode

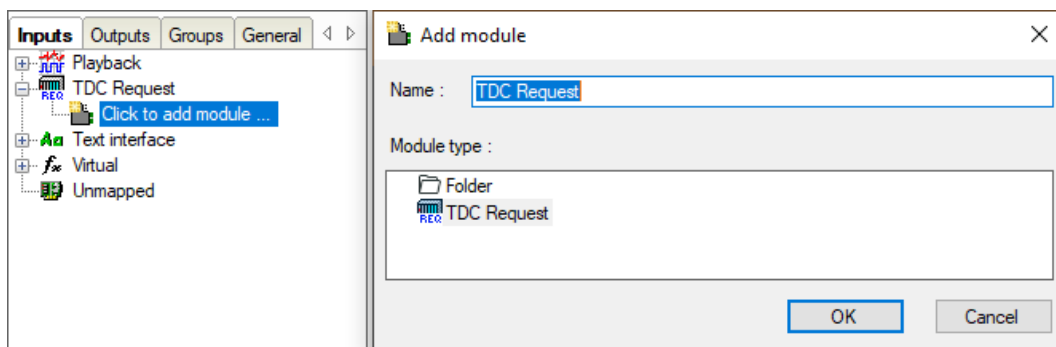
Processor	Address book	Date modified	Reload
-----------	--------------	---------------	--------

## 5.4 Request modules

Under the "SIMADYN request" or "TDC request" data interfaces, you can add a maximum of 64 data modules.

### Adding modules

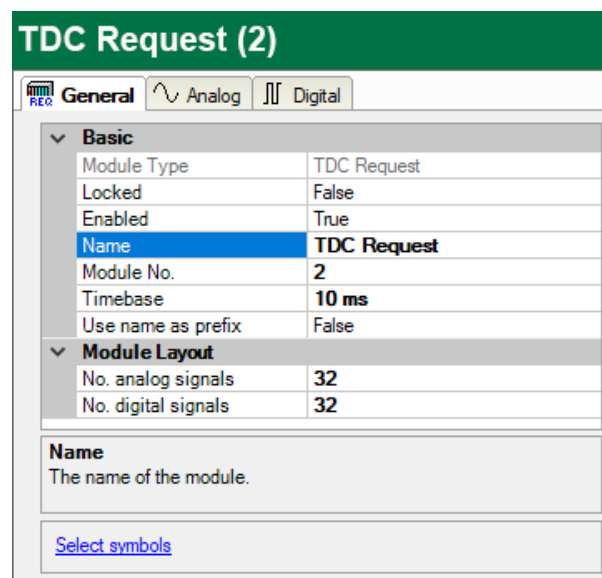
1. Click on the blue command *Click to add module...* located under each data interface in the *Inputs* or *Outputs* tab.
2. Select the desired module type in the dialog box and assign a name via the input field if required.
3. Confirm the selection with <OK>.



### 5.4.1 General settings

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

All modules have the following setting options.



**Basic settings****Module Type (information only)**

Indicates the type of the current module.

**Locked**

You can lock a module to avoid unintentional or unauthorized changing of the module settings.

**Enabled**

Enable the module to record signals.

**Name**

You can enter a name for the module here.

**Module No.**

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

**Timebase**

All signals of the module are sampled on this timebase.

**Use name as prefix**

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

---

**Note**

Because a module is not assigned to a PDA channel, you cannot modify the PDA channel time base here, i.e., the access cycle to the Siemens CPUs. For this, go to the PDA channel time base in the connection table of the data interface.

---

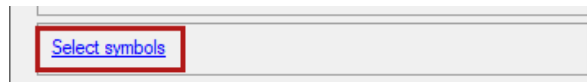
**Module Layout****No. of analog signals/digital signals**

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

### 5.4.2 Signal selection with symbol browser

You have two options to select the signals to be measured via the symbol browser:

- In the module's *General* tab, click on the *Select symbols* link.



The symbol browser opens.

In the symbol browser you can select all variables of the address book. *ibaPDA* enters the selected signals automatically in the appropriate table *Analog* or *Digital*. You can add several variables successively.

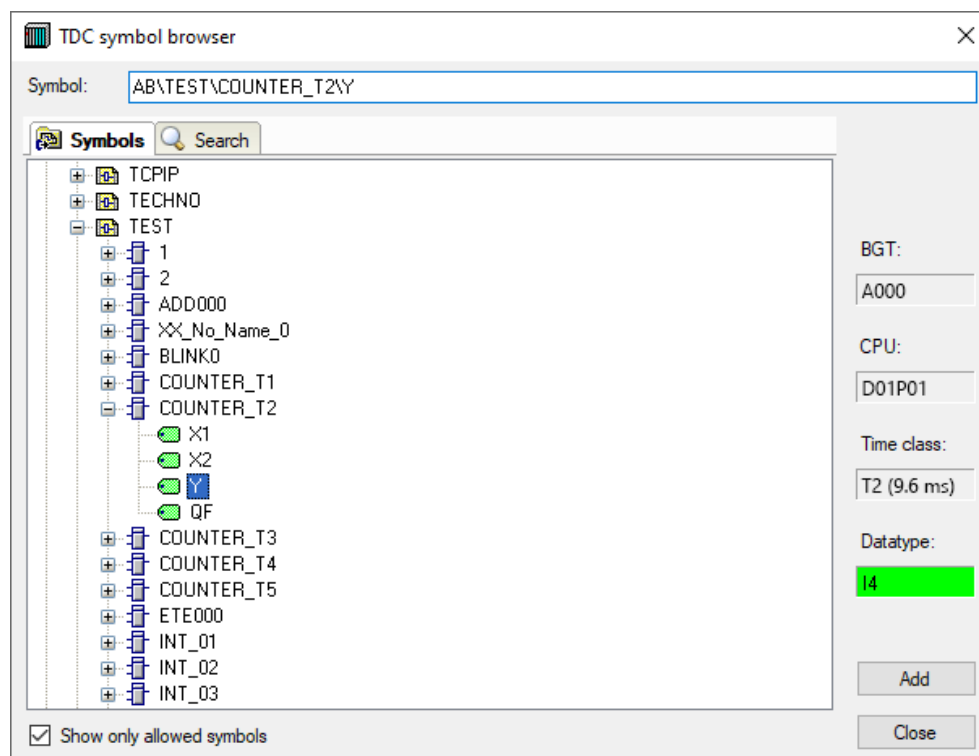
- On the *Analog* or *Digital* tab, click in a cell of the *Symbol* column.

The button <...> appears. Click this button to open the symbol browser.

In the symbol browser, you can only select variables with the data type matching the table. *ibaPDA* enters the variable in the previously selected cell. After every selection, the symbol browser closes.

#### TDC symbol browser

You see all variables arranged in a *PDA channel – chart – function block – connector* tree structure.



After selecting the variable, the symbol browser shows the associated data (BGT, CPU, data type and time class).

To transfer the selected variable is taken, double-click on the connector or click on <Add>.

**Show only allowed operands**

If this option is enabled, the symbol browser shows only variables that *ibaPDA* supports or that match the signal table from which you opened the browser.

If the *Show only allowed symbols* option is not activated, then the connectors are marked in color:

Green	The variable is valid. You can transfer it to the signal table with <Add>.
Yellow	The variable has a data type that does not match the selected column.
Red	The variable has a data type that is not supported by <i>ibaPDA</i> .

**Note**

If certain function blocks do not appear in the address book, check if the names of these blocks include an #-sign. In this case, disable the macro filter when generating the address book, see chapter [➤ Macros](#), page 45.

### 5.4.3 Signal selection by Drag & Drop (only with CFC)

#### Prerequisites

- An *ibaPDA* client must be installed on the same computer as the SIMATIC manager.
- The CFC program of this CPU must be flawlessly compiled.
- The address book must be present in *ibaPDA*.
- Drag & drop is not possible with STRUC programs.

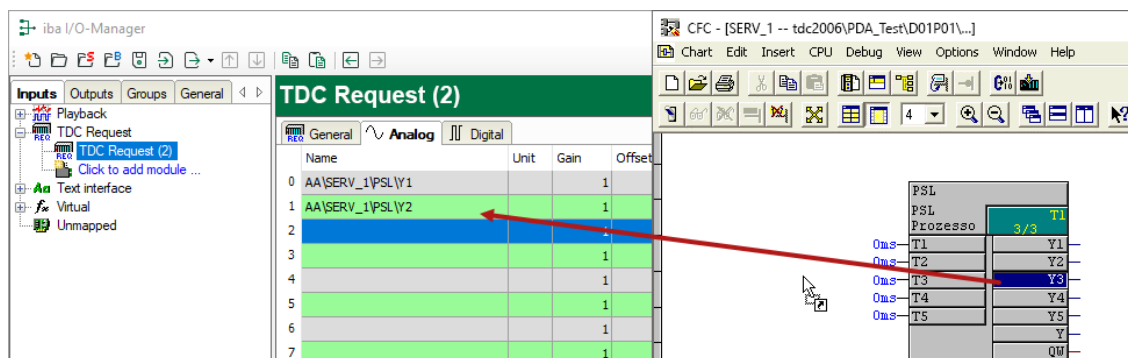
#### Note



For correct generation of the connector address with the drag & drop operation, *ibaPDA* requires an up-to-date address book. Only in this way the path with processor and chart name can be generated. Via drag & drop, you cannot project connectors that are not included in the address book.

#### Procedure

1. Open the I/O Manager of *ibaPDA* and the signal table of the required request module.
  2. Open the CFC chart in the SIMATIC manager and minimize the window so that both the CFC chart and the *ibaPDA* window can be seen.
  3. Drag the connector from the CFC chart into the desired line of the signal table in the I/O Manager of *ibaPDA*. *ibaPDA* automatically opens the correct signal table in the process.
- The connector is now a measured signal in the signal table of the module.



4. When you have selected your data, close the I/O Manager with <OK> and thus hand over the configuration to the *ibaPDA* server.

The data are requested. If no error occurs in the process, then acquisition starts and you can drag the signals into the display.

### 5.4.4 Data modules

At the start of acquisition, *ibaPDA* applies a data module for each active PDA channel. They can be seen in the hardware tree structure of the I/O Manager under the respective ibaFOB-SD/-TDC card.

Select a data module in the interface tree to display the current variable raw values in numerical representation.

The screenshot shows the 'iba I/O Manager' window. On the left, the 'Inputs' tab is active, showing a tree structure under 'Link 0' with modules 'A000\D01P01', 'B0010\D01P01', and 'B0010\D02P02'. The main panel displays the 'AB T3 (A000\D01P01)' data module. The 'Analog' tab is selected, showing a table of data points.

ID	Name	Symbol	Actual
0	[1:3] AA\SERV_1\PSLY4	AA\SERV_1\PSLY4	0,003953125
1	[1:2] AA\SERV_1\PSLY3	AA\SERV_1\PSLY3	0,015375
2	[1:5] AA\SERV_1\PSLY	AA\SERV_1\PSLY	0,4542032
3	[1:4] AA\SERV_1\PSLY5	AA\SERV_1\PSLY5	0,003585937
4	[0:1] AA\SERV_1\PSLY2	AA\SERV_1\PSLY2	0,181
5	[0:0] AA\SERV_1\PSLY1	AA\SERV_1\PSLY1	0,354

The modules are arranged according to BG name \ BGT name. If several PDA channels are projected on a CPU, they can be seen within a data module.

If you have requested more than 32 analog values or digital values for a PDA channel and other PDA channels are projected for the assigned CPU, then *ibaPDA* automatically moves these measurement values into the other PDA channel.

When requesting with SER05, the data modules are sorted according to the time slices.

The screenshot shows the 'iba I/O Manager' window. On the left, the 'Inputs' tab is active, showing a tree structure under 'Link 0' with modules 'AB T1 (A000\D01P01)', 'AB T2 (A000\D01P01)', 'AB T3 (A000\D01P01)', 'AB T4 (A000\D01P01)', 'AB T5 (A000\D01P01)', 'BB T1 (A000\D01P01)', 'BB T2 (A000\D01P01)', 'BB T3 (A000\D01P01)', 'BB T4 (A000\D01P01)', and 'BB T5 (A000\D01P01)'. The main panel displays the 'AB T3 (A000\D01P01)' data module. The 'Analog' tab is selected, showing a table of data points for multiple channels.

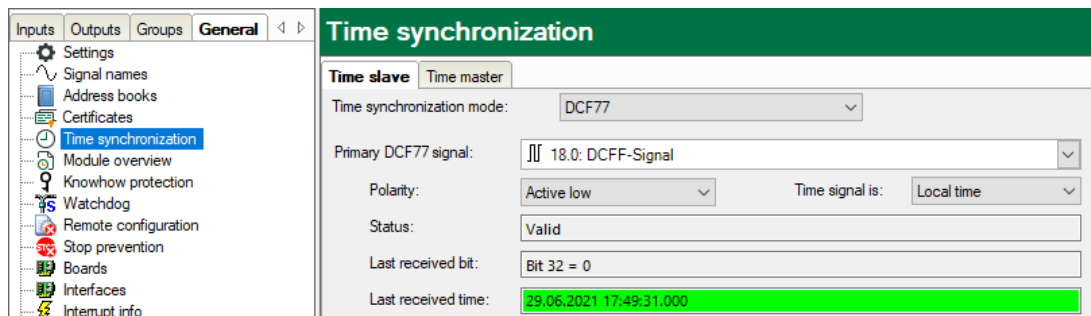
ID	Name	Symbol	Actual
Channel: AB T1			
Channel: AB T2			
Channel: AB T3			
0	[3:0] AB\TEST\COUNTER_T3\Y	AB\TEST\COUNTER_T3\Y	166060
1	[3:1] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	-1
2	[3:2] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	3
3	[3:3] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	30
4	[3:4] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	333
5	[3:5] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	161,19
6	[3:6] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	218,9
7	[3:7] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	1,04301
8	[3:8] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	6,11857
9	[3:9] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	-1
10	[3:10] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	3
11	[3:11] AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u...	AB\LITE-GDM.LITE_M1-4_T2-3.LITE_3u4_T3\...	30

## 5.5 Time synchronization in ibaPDA

There are two mechanisms available for time synchronization with the Siemens control. For this, see the configuration instructions in chapter [Time synchronization on Siemens side](#), page 25.

### 5.5.1 Time synchronization with DCF77 signal

1. In the I/O Manager's *General* tab, select the *Time synchronization* node.
2. Select the *DCF77* synchronization mode.
3. In the *Primary DCF77 signal* drop-down list, select the DCF77 signal previously added to the measurement.
4. Set the polarity to *Active low* (presetting).
5. In the *Time signal is* drop-down list, set the time mode of the DCF77 signal.



6. Apply the settings with <OK> or <Apply>.
  - Initially, the *Last received time* field is shown in red. After at most two minutes the time is synchronized.
  - Then the field is shown in green. Synchronization takes place every minute.
  - If the field stays or turns red, then the time is not synchronized. The DCF77 signal may be faulty or too inaccurate.
7. To ensure the synchronization, you can specify a second source for synchronization. The secondary DCF77 signal only becomes active if the CPU on which you have projected the primary DCF77 signal fails.

#### Note



Make sure that the signal is as accurate as possible, i.e., measured with the fastest possible acquisition time, ideally in 1 ms.

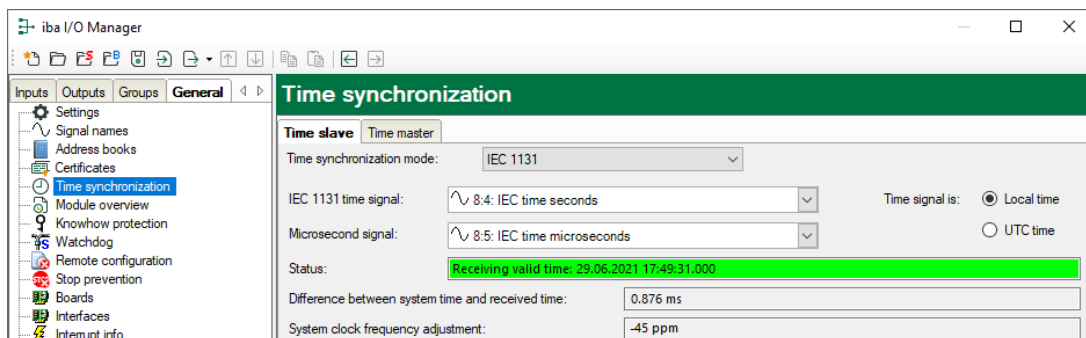
### 5.5.2 Synchronization with IEC time signal

1. Acquire the 3 integer values CSH, CSL and CSM of the time stamp from the function block RTCREL (in CFC) or RTC005 (in STRUC), see chapter [Time synchronization on Siemens side](#), page 25.
2. Calculate the IEC 1131 time stamp (relative to 01.01.1972) from the time stamp of the controller (relative to 01.01.1988) by means of virtual signals:
  - Add "567993600" to the seconds.
  - Multiply the 1/10 milliseconds with "100".

Virtuell (8)

f <sub>x</sub> General ~ Analog Digital		
Name	Expression	
0 Seconds high	f <sub>x</sub> [RTCREL.CSH]	?
1 Seconds low	f <sub>x</sub> [RTCREL.CSL]	?
2 1/10 Milliseconds	f <sub>x</sub> [RTCREL.CSM]	?
3 Seconds since 1988	f <sub>x</sub> DWORD([Seconds low], [Seconds high])	?
4 IEC time seconds	f <sub>x</sub> [Seconds since 1988] + 567993600	?
5 IEC time microseconds	f <sub>x</sub> [1/10 Milliseconds] * 100	?

3. In the I/O Manager's *General* tab, select the *Time synchronization* node.
4. Select the *IEC 1131* synchronization mode.
5. Select the calculated virtual signals "IEC time seconds" and "IEC time microseconds" in the drop-down lists.



6. For *Time signal is*, set the time format generated by the time source.
  7. Apply the setting with <OK> or <Apply>.
- The *Status* field is shown in green immediately and the time is synchronized
- If the field remains red, then the time values have the wrong format or the received time is outside the plausible range.

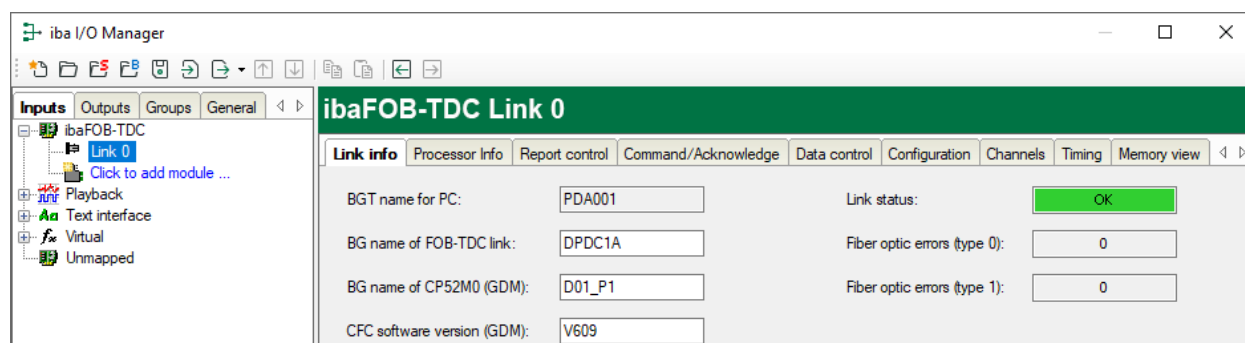
## 5.6 ibaFOB-SD/ibaFOB-TDC diagnostics

When the Link level is highlighted in the tree, you can find some diagnostic information about communication and processors of the card.

- **Link Info:** Set connection parameters and status  
see ↗ *Link Info tab*, page 54
- **Processor Info:** Information about the DPR assignment interface, hardware version and firmware version  
see ↗ *Processor Info tab*, page 55
- **Report control, Command/Acknowledgement, Data control:** Information about the telegram traffic of the PDA channels  
see ↗ *Report control tab*, page 56, ↗ *"Command/Acknowledge" tab*, page 57 and ↗ *Data control tab*, page 58
- **Configuration:** Information about the area in the rack link in which all connected racks are registered  
see ↗ *Configuration tab*, page 59
- **Channels:** Information about all communication channels configured in the rack link  
see ↗ *Channels tab*, page 59
- **Timing:** Information about utilization of the *ibaFOB* card and access statistics  
see ↗ *Timing tab*, page 61
- **Memory view:** Content of DPR memory in the *ibaFOB* card  
↗ *Memory view tab*, page 63

### 5.6.1 Link Info tab

General information on the connection is displayed here.



#### BGT name of the PC

With this name, *ibaPDA* logs into the administration area of the rack link component as BGT. Default setting is "PDA001" (from registry).

#### BG name of the FOB-TDC connection

With this name, the *ibaFOB* card logs into the rack link. The setting is taken from the connection list of the specific request interface.

#### BG name of the rack link

Is read by <Autodetect processors> on the request interface from the rack link

## STRUC/CFC software version of the control

Is read by <Autodetect processors> on the request interface from the rack link

### Connection status

- green: "OK"
- red: "Interrupted"

### Fiber optic error counters

Using the analysis of error interrupts by the optical fiber hardware driver, *ibaPDA* detects a connection interruption. In normal operation, the counters have to remain constant. The long-term monitoring of the counters can provide additional information on the quality of data transmission.

## 5.6.2 Processor Info tab

General information on the card and its firmware are displayed here.

The screenshot shows the 'Processor Info' tab in the ibaPDA software. The interface includes several sections:

- Command info:** Fields for Reset command (0xFF00), Reset comm. timeout (0 s), General command (0xFF00), General comm. timeout (0 s), Diagnose command (0xFF00), Diagnose comm. timeout (0 s), and Interrupt cycle time (0 ms).
- Processor Info:**
  - Life counter: 126
  - Address space size: 128 kB
  - Micro clock: 24000
  - Micro ID: C165
  - Reset Processor button
  - Interrupt Control: Slave, Generated Internally, Interrupt is activated on PCI bus
  - Hardware designation and version: FOB\_TDC-PCI\_ H1.2
  - iba standard designation and version: iba-FOB\_TDC\_ S1.1
  - Firmware designation and version: FOB\_TDC\_ F2.5
  - Version info: FTCB5-V2.5 FOB\_TDC iba / UserPDA001 : 20.08.2008/2 C165 24.000MHz >B0TP< FTCB5.PRM: 274485.21.08.08/11.47 UserPDA001 iba - fue -07.02.08/11.29.21?Z<??????????

### Command info

- Display of current values for command and monitoring times

### Processor info

- Own life counter
- Processor information (type, clock rate, DPR size)
- Interrupt information
- Information about hardware and firmware version

### <Reset processor>

Click this button to reset the card. (Only visible if no recording is running.)

### 5.6.3 Report control tab

The results of logging into the rack link are displayed here.

Link info	Processor Info	<b>Report control</b>	Command/Acknowledge	Data control	Configuration	Channels	Timing	Memory view
Nr of valid PN:		<b>Valid mask</b>						
<input type="text" value="7"/>								
		<b>Ok mask</b>						
		<b>Error mask</b>						
PN: <input type="text" value="F1"/>		Info: <input type="text" value="0x0000"/>						
Mode: <input type="text" value="Request Version"/>								
BGT name of PC: <input type="text" value="PDA001"/>		CFC software version (GDM): <input type="text" value="V609"/>						
BG name of FOB-TDC link: <input type="text" value="DPDA1A"/>		BG name of CP52M0 (GDM): <input type="text" value="D01 P1"/>						
ZB connectors								
CDM: <input type="text" value="0x00FF"/>		NCP: <input type="text" value="0x0003"/>		ENO: <input type="text" value="0x0000"/>		BZS: <input type="text" value="0x0008"/>		
CDA: <input type="text" value="0x0000"/>		QTS: <input type="text" value="0x00FF"/>		EZU: <input type="text" value="0x0000"/>		KZS: <input type="text" value="0x0000"/>		

**Nr of valid PN**

Number of CPUs (PN) from the processor list

## Bitmasks

<i>Valid mask</i>	displays which processors were activated
<i>OK mask</i>	displays processors that responded with OK
<i>Error mask</i>	displays processors that responded with Error

If you move the slider under the bitmasks with the mouse, the following associated data are displayed for the current processor.

The bit numbering (0 is right) corresponds to the sequence of processors in connection list.

- *PN, Info*: PDA channel name and error info
  - *Mode*: Request Version or Lite Version
  - Connection data (as in link info, chapter [↗ Link Info tab](#), page 54)
  - *ZB connectors*: diagnostic fields initialization function block
- After correct log-in, the connectors have the following states:

<i>CDM</i>	0x00FF	Coupling OK
<i>CDA</i>	0x0000	Coupling released
<i>NCP</i>	0x0001...n	Number of connected racks (incl. <i>ibaPDA</i> )
<i>QTS</i>	0x00FF	Coupling initialized
<i>BZS</i>	0x0008	Life counter monitoring released
<i>ENO</i>	0x0000	No error

5.6.4 Command/Acknowledge tab

The results of the "Command/acknowledgement" handshake are displayed here.

Link infoProcessor InfoReport control**Command/Acknowledge**Data controlConfigurationChannelsTimingMemory view

Nr of valid PN:

7

**Valid mask**

0000000000000000000000000000100000001111000011

**Ok mask**

0000000000000000000000000000100000001111000011

**Error mask**

00

PN:

F1

Command info:

0x0000

Acknowledge info:

0x0000

Nr of bits requested:

1

Nr of bits acknowledged:

1

Nr of floats requested:

3

Nr of floats acknowledged:

3

Nr of valid PN

Number of CPUs (PN) from the processor list

Bitmasks

<i>Valid mask</i>	displays which processors were activated
<i>OK mask</i>	displays processors that responded with OK
<i>Error mask</i>	displays processors that responded with Error

If you move the slider under the bitmasks with the mouse, the following associated data are displayed for the current processor.  
The bit numbering (0 is right) corresponds to the sequence of processors in connection list.

- *PN*: PDA channel name
- *Command info*: error number with command telegram
- *Acknowledge info*: error number with acknowledgment telegram
- Number of requested and confirmed digital and analog values

iba

5.0

57

### 5.6.5 Data control tab

The results of the measurement data transmission are displayed here.

The screenshot shows the 'Data control' tab in the ibaPDA software. It features three bitmaps: 'Valid mask', 'Ok mask', and 'Error mask', each with a bit numbering slider. Below the bitmaps, there are input fields for 'PN' (T1), 'Access counter' (45533), 'Complete data transfers' (45525), 'Info' (0x0000), 'Timebase (ms)' (1), 'Incomplete data transfers' (94), and '% completed data transfers' (100). A 'Reset counters' button is located at the bottom.

#### Nr of valid PN

Number of CPUs (PN) from the processor list

#### Bitmasks

<i>Valid mask</i>	displays which processors were activated
<i>OK mask</i>	displays processors that responded with OK
<i>Error mask</i>	displays processors that responded with Error

If you move the slider under the bitmasks with the mouse, the following associated data are displayed for the current processor.

The bit numbering (0 is right) corresponds to the sequence of processors in connection list.

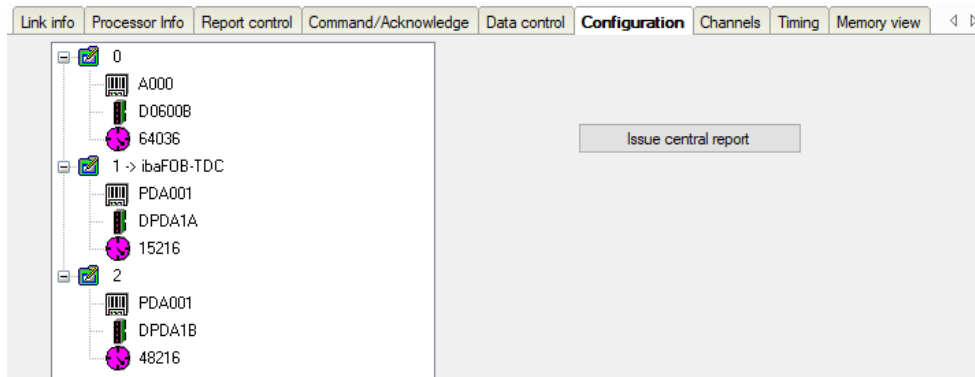
- *PN, Info*: PDA channel name and error info
- Access counters
  - *Access counters*: Number since the start of the measurement (overrun at 65535)
  - *Time base*: access cycle (set acquisition time base)
  - *Data transfer rate*: Percentage since the last reset

#### <Reset counters>

Click this button to reset the calculated times and error counters in the table to 0.

## 5.6.6 Configuration tab

The data from the log-in area of the rack link are displayed here.



All racks logged in the communication memory of the rack link, are shown. For each rack, the following data are displayed:

- BGT name of the logged-in rack (e.g. of the *ibaPDA*-systems)
- BG name of the connection, e.g., CS22, CP52A0, CP53 or name of the *ibaPDA* connection
- Running life counter of the connection

### <Issue central report>

Log in of the *ibaFOB-SD/ibaFOB-TDC* at the coupling partner. (Not necessary, as this is done automatically at the first start of the acquisition by *ibaPDA*.)

## 5.6.7 Channels tab

Information from the data area of the rack link (communication channels) are displayed here.

Channel name	BGT name	PN	Length	Mode	Status
F1PDADAT	F0010	D01P01	136	Refresh	Buffer full
F2PDADAT	F0010	D02P02	136	Refresh	Buffer full
F3PDADAT	F0010	D03P03	136	Refresh	Buffer full
F4PDADAT	F0010	D04P04	136	Refresh	Buffer full
G1PDADAT	G0010	D01P01	136	Refresh	Sender initialized
G2PDADAT	G0010	D02P02	136	Refresh	Sender initialized
G3PDADAT	G0010	D03P03	136	Refresh	Sender initialized
G4PDADAT	G0010	D04P04	136	Refresh	Sender initialized
I1PDADAT	I0010	D01P01	136	Refresh	Sender initialized
I2PDADAT	I0010	D02P02	136	Refresh	Sender initialized
I3PDADAT	I0010	D03P03	136	Refresh	Sender initialized
I4PDADAT	I0010	D04P04	136	Refresh	Sender initialized
H1PDADAT	H0010	D01P01	136	Refresh	Sender initialized
H2PDADAT	H0010	D02P02	136	Refresh	Sender initialized
H3PDADAT	H0010	D03P03	136	Refresh	Sender initialized
H4PDADAT	H0010	D04P04	136	Refresh	Sender initialized
H5PDADAT	B0010	D05P05	136	Refresh	Sender initialized

When you enter the search term (with wildcard "?") and click on <Search>, *ibaPDA* searches the rack link for telegrams that match the search term. For example:

"?????DAT": all *ibaPDA* data telegrams (Siemens → *ibaPDA*)

"?????CMD": all *ibaPDA* assignment telegrams (*ibaPDA* → Siemens)

"?????ACK": all *ibaPDA* acknowledgment telegrams (Siemens → *ibaPDA*)

### BGT diagnostics

- Log-in data read from the log-in area of the rack link
- *Error code*: error number in the case of access error, see chapter [Errors with diagnostic functions](#), page 73

### List of channels found

- *Channel name*: telegram name
  - *BGT name*: only with PDA channels, otherwise address level 1 or empty
  - *PN*: only with PDA channels, otherwise address level 2 or empty
  - *Length*: channel user data length in bytes
    - DAT telegram: 136
    - ACK telegram: 128
    - CMD telegram: 1020 with CFC, 964 with STRUC
  - *Mode*: handshake with CMD and ACK, refresh with DAT
  - *Status*: shows buffer state: initialized, buffer full or buffer empty
- The status is only a snapshot and is not dynamic.

### Note



The values are not dynamic, but show only a snapshot at the time of access.

Click on a column heading to sort the table can be sorted in ascending/descending order.

Drag a column heading onto the free area above to group the table (see figure).

BGT name / Length /			
Channel name	PN	Mode	Status
+ BGT name:			
- BGT name: A000			
+ Length: 136			
S1PDADAT	D01P01	Refresh	Sender initialized
T1PDADAT	D01P01	Refresh	Sender initialized
U1PDADAT	D01P01	Refresh	Sender initialized
V1PDADAT	D01P01	Refresh	Sender initialized
S2PDADAT	D01P01	Refresh	Sender initialized
T2PDADAT	D01P01	Refresh	Sender initialized
S3PDADAT	D01P01	Refresh	Sender initialized
T3PDADAT	D01P01	Refresh	Sender initialized
- Length: 532			
ABPDADA1	D01P01	Refresh	Buffer full
BBPDADA1	D01P01	Refresh	Buffer full

### 5.6.8 Timing tab

A statistical analysis of all accesses is done in the firmware of the *ibaFOB-SD/ibaFOB-TDC*. It is primarily intended as an aid for software development and optimization. However, it is also accessible to the user for optimal configuration of TDC and *ibaPDA*.

#### Note



Some values are not mathematically reproducible because certain marginal conditions are not taken into account. As tendencies and benchmarks they are nonetheless helpful for optimizing settings.

In order to correctly interpret the values, knowledge of the internal assignment and data transfer interface between *ibaPDA* and *ibaFOB-SD/ibaFOB-TDC* firmware is necessary. Thus, some brief explanations follow.

A basic sample time is set in the *ibaPDA* system. It becomes the basic clock rate of an *ibaFOB-SD/ibaFOB-TDC* access cycle. At the beginning of the cycle, the sequence control starts the acquisition processes for *all* requested processors. At the end of the cycle, the sequence control checks the results of the accesses. There are the following possibilities normally (no access errors):

Case	Cause
New data was read.	The transmission cycle is faster or the same as the reception cycle ( <i>ibaPDA</i> sample time).
No new data was read.	The transmission cycle is slower than the reception cycle ( <i>ibaPDA</i> sample time).
Access to PN has not ended yet.	The <i>ibaFOB</i> card is overloaded; the amount of data or processors cannot be processed within the cycle time.

The results vary within a certain bandwidth due to asynchronous events, e.g., incrementing the life counter in CS14, reading the technosting channel or different time slices of the processors. The values for the statistics are recalculated every 100 basic [sample] times.

Link info	Processor Info	Report control	Command/Acknowledge	Data control	Configuration	Channels	<b>Timing</b>	Memory view	◀ ▶
PDA cycle time [ms]:		1				Reset counters			
Sampling efficiency:		100%							
	PN Name	% of valid datatransfers	Valid datatransfer counter						
0	T2	50	27541						
1	T1	25	13771						
2	??	??	??						
3	??	??	??						
4	??	??	??						
5	??	??	??						
6	F1	25	13770						
7	F2	25	13770						
8	F3	25	13770						
9	F4	25	13770						
10	??	??	??						

#### PDA cycle time

Currently set *ibaPDA* acquisition timebase in ms (basic sample time)

### Sampling efficiency

Load of *ibaFOB-SD/ibaFOB-TDC* card

Sampling efficiency = (number of finished reading calls) / (number of started reading calls) per cycle

The reading calls are generally distributed in the cycle of the basic sample time. The finished reading calls are counted, regardless of whether new data arrive or not. No new data arrive if the data telegram transmission cycle is slower than the *ibaPDA* basic sample time. Calls are regarded as not completed in a cycle if the processing time of the firmware is longer than the *ibaPDA* basic sample time. Increasing the channel time base (see chapter [➤ Creating PDA channels](#), page 35) has no direct influence on the sampling efficiency since the completion of the call is counted, regardless of whether new data arrive. There is an indirect influence because the calls can be processed more quickly without new data than with new data.

#### Example sampling efficiency

PDA cycle time 2 ms, 40 channels. Only 30 channels are concluded within 2 ms.

→ Eff. =  $30/40 = 75\%$

For an effective measurement, a value between 80 % and 100 % should be aimed for.

### PN Name

Channel ID in the order of the connection list

### % of valid data transfers

The data telegram transmission clocking is the maximum from the two "Cycle SER04B function block" and "Channel time base" parameters, and thus is directly dependent on the channel time base, see [➤ Creating PDA channels](#), page 35.

% of valid data transfers = (*ibaPDA* acquisition timebase) / (data telegram transmission clocking)

#### Example % of valid data transfers

Acquisition timebase 2 ms, cycle SER04B 2 ms, channel time base 12 ms

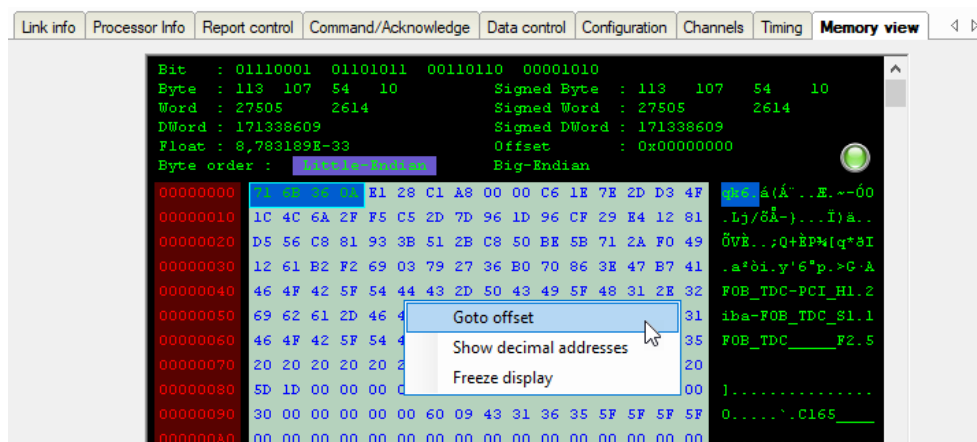
→ % of valid data transfers =  $2/12 = 16.6\%$

### Valid data transfer counter

Number of successful data transfers since the last reset.

### 5.6.9 Memory view tab

For service and diagnostic purposes, the *Memory View* tab can be an important aid in troubleshooting communication problems. For normal use, you do not need this view.



The upper area displays the highlighted data in various data formats.

A flashing green light indicates a running system.

#### Handling the memory view via the context menu

Right-click within the memory data display area to open the context menu.

##### Goto offset

Enter a specific memory address that you want to check. Click <OK> to jump to the address you entered and view the contents.

##### Show decimal addresses

When this option is enabled, the memory view displays the decimal address instead of the hexadecimal address

##### Freeze display

When this option is enabled, the data is no longer updated and you can evaluate the current state. The red light indicates that the display is not updated.

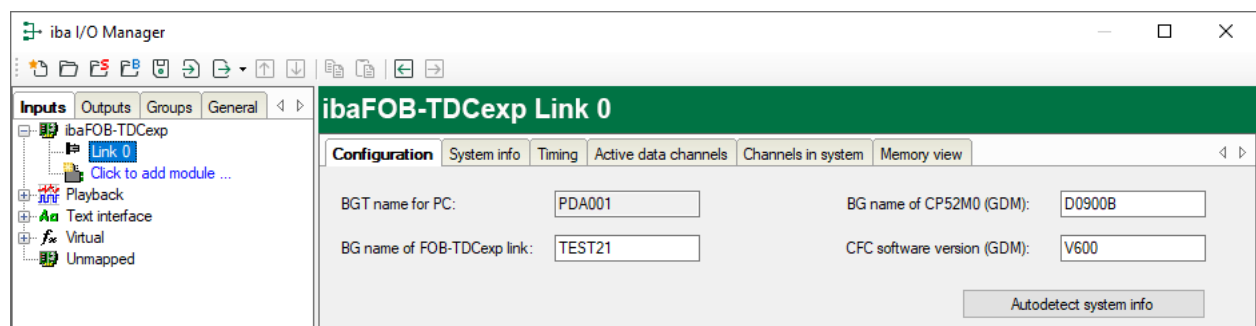
## 5.7 ibaFOB-SDexp/ibaFOB-TDCexp diagnostics

When the Link level is highlighted in the tree, you can find diagnostic information about communication and processors of the card.

- **Configuration:** Configured connection parameters  
see ↗ *Configuration tab*, page 64
- **System info:** Information about the link status, system information of the coupling partner, log-in area in the rack link  
see ↗ *System Info tab*, page 65
- **Timing:** Information about utilization of the *ibaFOB* card and access statistics  
see ↗ *Timing tab*, page 66
- **Active data channels:** Overview of PDA channels, their time classes and access times  
see ↗ *Active data channels tab*, page 68
- **Channels in system:** Information about all communication channels configured in the rack link  
see ↗ *Channels in system tab*, page 68
- **Memory view:** Content of DPR memory in the *ibaFOB* card  
see ↗ *Memory view tab*, page 69

### 5.7.1 Configuration tab

Here you can enter the data that are needed for the connection of *ibaPDA* to the SIMATIC TDC system and the Simadyn SD system.



#### BGT name for PC

With this name, *ibaPDA* logs in on the rack link component group as BGT. The default setting is "PDA001".

#### BG name of FOB-SDexp/ibaFOB-TDCexp link

With this name, *ibaFOB-SDexp/ibaFOB-TDCexp* logs in on the rack link. The name has to be unambiguous, if there exists more than one FO connection to the same coupling partner. Changes are entered in the respective field in the connection list of the interfaces *Simadyn Request* or *TDC Request*.

#### BG name of the rack link

Is read from the control by clicking on the button <Autodetect system info>.

#### CFC software version

Is read from the control by clicking on the button <Autodetect system info>.

## 5.7.2 System Info tab

Information on the connection of *ibaPDA* and the SIMATIC TDC system are displayed here.

### Link diagnostics

#### ■ Link status:

- green: "OK"
- red: "Error"

#### ■ Error bytes: Counter

Access error: In normal operation, the counters need to remain constant. The long-term monitoring of the counters can provide additional information on the quality of data transmission.

#### ■ Last access time/Max. access time: Duration of the last access or the longest access since resetting.

### <Reset counters>

Click this button to reset the error counter and the max. access time.

### System diagnostics

- *BGT name, BG name, Software version*: Values read from the store of the coupling partner
- *Cycle time*: Monitoring cycle of the communication (task of the integrated initialization function block)
- *System free memory*: free/whole memory in the coupling partner (GDM, CP53 or CS13)
- *No of channels*: total number of the configured channels in the communication memory

### Central report area

All racks logged in in the rack link incl. *ibaPDA* are displayed. The following data is displayed for each rack:

- BGT name of the rack logged in (e.g. of the *ibaPDA* system)
- BG name of the connection, e.g. CS22 or name of the *ibaPDA* connection
- Running life counter of the connection

## Central report status

- green: "OK"
- red: Error number

### <Issue central report>

Log in of the *ibaFOB-SDexp/ibaFOB-TDCexp* at the coupling partner. (Not necessary, as this is done automatically at the first start of the acquisition by *ibaPDA*.)

## 5.7.3 Timing tab

Information on the accesses of the card to the SIMATIC TDC CPUs and the accesses of the card to the data memory (DMA) are displayed here.

The screenshot shows the 'Timing' tab of the ibaPDA software. It contains two main sections: 'Timing' and 'Image generation'.

**Timing Section:**

- ☐ Manually set read cycle times
- T1: 1.0 ms
- T2: 4 x T1 4.0 ms
- T3: 16 x T1 16.0 ms
- T4: 64 x T1 64.0 ms
- T5: 256 x T1 256.0 ms

	Actual read cycle time	% valid transfers
T1:	1.2 ms	100%
T2:	1.2 ms	100%
T3:	1.2 ms	100%
T4:	1.2 ms	100%
T5:	1.2 ms	100%

Current transfer duration: 50 µs  
 Max transfer duration: 88 µs  
 Dropped transfers: 5  
 Automatic channel initializations: 0

**Image generation Section:**

Images processed at interrupt:

- Actual: 1
- Min: 1
- Max: 1

Images in DMA buffer: 10

Images copied to interrupt buffer: 80189

DMA buffer empty: 0

Dropped images: 0

Image sample rate: 1 ms

Image size (bytes): 80

DMA buffer size: 4 MB

DMA buffer element size: 4 kB

Reset counters

### Timing

- *Manually set read cycle times*: Possibility of selecting manually the access times of the single time classes.
- *Actual read cycle time*: Currently selected or automatically calculated access times.
- *% valid transfers*: Proportion of the values that can be acquired within the *Actual read cycle time*.
- *Current transfer duration*: Current duration of the data transfer for a sample. The ratio of *Current transfer duration* and the lowest *Actual read cycle time* shows the utilization of the *ibaFOB-SDexp/ibaFOB-TDCexp*.
- *Max. transfer duration*: Max. duration of a data transfer since the last reset.
- *Dropped transfers*: Number of lost samples. Whenever the *Current transfer duration* is higher than the *Actual read cycle time*, this counter is incremented.

- *Automatic channel initializations*: Diagnostics counter for automatic actions.

- Access times

The smallest time slice of all CPUs is taken as default setting for the access times. The result is an 'oversampling' in the time slices T2-T5 on the one hand, but on the other hand the delays during measurement are minimized. If this 'oversampling' results in an overload of the *ibaFOB-SDexp/ibaFOB-TDCexp* card, it is possible to modify the read access times for the single time slices. Please note the following:

- If the read cycle time of a time slice is slower than the corresponding time slice of a CPU, short pulses may be lost.
- The slower the read cycle times, the larger are the delays of the signal acquisition.

The channel list in the *Active data channels* tab offers the possibility to compare the configured time slices with the current access times and thus to set optimized values, see ➤ *Active data channels tab*, page 68.

- Capacity utilization

You can see the current capacity utilization by having a look at the fields *Current transfer duration*, *Max. transfer duration* and *Dropped transfers*:

- As long as the *Max. transfer duration* is lower than the lowest access cycle, all samples are transferred without any losses.
- If the *Max. transfer duration* is higher and the *Current transfer duration* lower, samples get sporadically lost. These samples are counted in *Dropped transfers*.
- If the *Current transfer duration* lies near the access cycle or above, the card is overloaded.
- You can see the percentage of the lost samples in *% valid transfers*.

### Image generation

Diagnostic data for the DMA accesses to the *ibaPDA* data memory.

#### <Reset counters>

Click this button to reset all counters and restart the calculation of *% valid transfers* and *Max. transfer duration*.

### 5.7.4 Active data channels tab

Here, all data channels are displayed in a table with information about the configured time class, the access time and the data volume.

Configuration	System info	Timing	Active data channels	Channels in system	Memory view
Short name	Channel name	Type	Time class	Read cycle time	Data size
0 AB	ABPDADA1	SER05	T1 (1,0 ms)	2,0 ms	12
1 AB	ABPDADA2	SER05	T2 (8,0 ms)	8,0 ms	528
2 AB	ABPDADA3	SER05	T3 (32,0 ms)	32,0 ms	140
3 AB	ABPDADA4	SER05	T4 (128,0 ms)	128,0 ms	100
4 AB	ABPDADA5	SER05	T5 (512,0 ms)	512,0 ms	68
5 BB	BBPDADA1	SER05	T1 (1,0 ms)	2,0 ms	20
6 BB	BBPDADA2	SER05	T2 (8,0 ms)	8,0 ms	20
7 BB	BBPDADA3	SER05	T3 (32,0 ms)	32,0 ms	20
8 BB	BBPDADA4	SER05	T4 (128,0 ms)	128,0 ms	20
9 BB	BBPDADA5	SER05	T5 (512,0 ms)	512,0 ms	20
10					

By double-clicking on a line with an entry, you are forwarded to the memory view of this data channel, see [Memory view tab](#), page 69.

### 5.7.5 Channels in system tab

Here, you find information about the coupling memory and the communication channels configured in this memory.

Configuration	System info	Timing	Active data channels	Channels in system	Memory view
Search string: <input type="text" value="????DA?"/> Search 39 channels found					
Drag a column header here to group by that column					
Channel name	BGT name	PN	Length	Mode	Status
F1PDADAT	F0010	D01P01	136	Refresh	Buffer full
F2PDADAT	F0010	D02P02	136	Refresh	Buffer full
F3PDADAT	F0010	D03P03	136	Refresh	Buffer full
F4PDADAT	F0010	D04P04	136	Refresh	Buffer full

When you enter the search term (with wildcard "?") and click on <Search>, *ibaPDA* searches for channels that match the search term.

#### List of channels found

- **Channel name:** all communication channels in the memory
- **BGT name:** only for PDA channels that have been created by the SER04/SER05 service block
- **PN name:** only for PDA channels that have been created by the SER04/SER05 service block
- **Length:** channel user data length in bytes
- **Mode:** channel mode: Refresh, Handshake, Select or Multiple
- **Status:** Receiver/Sender initialized, channel in operation, buffer empty/full

#### Note



The values are not dynamic, but show only a snapshot at the time of access.

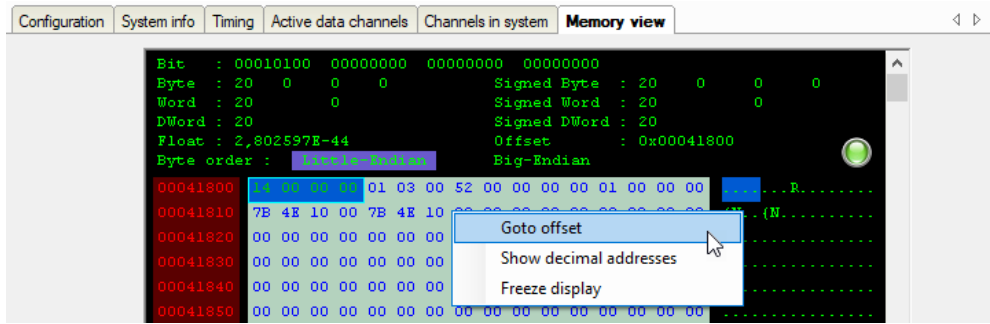
Click on a column heading to sort the table can be sorted in ascending/descending order.  
Drag a column heading onto the free area above to group the table (see figure).

BGT name / Length /

Channel name	PN	Mode	Status
BGT name:			
BGT name: A000			
Length: 136			
S1PDADAT	D01P01	Refresh	Sender initialized
T1PDADAT	D01P01	Refresh	Sender initialized
U1PDADAT	D01P01	Refresh	Sender initialized
V1PDADAT	D01P01	Refresh	Sender initialized
S2PDADAT	D01P01	Refresh	Sender initialized
T2PDADAT	D01P01	Refresh	Sender initialized
S3PDADAT	D01P01	Refresh	Sender initialized
T3PDADAT	D01P01	Refresh	Sender initialized
Length: 532			
ABPDADA1	D01P01	Refresh	Buffer full
BBPDADA1	D01P01	Refresh	Buffer full

5.7.6 Memory view tab

The DPR memory is displayed here on the *ibaFOB-SDexp/ibaFOB-TDCexp* card.



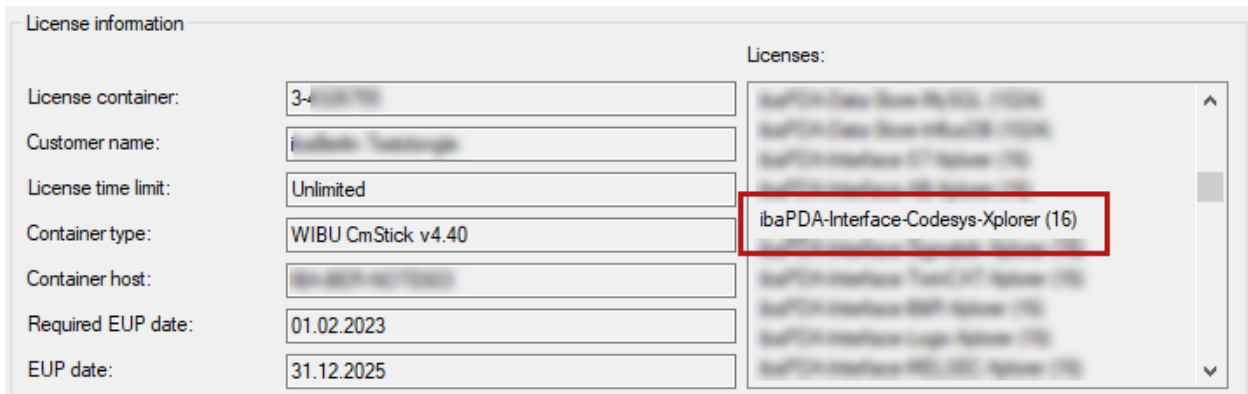
The marked data are displayed in the upper area in various data formats.  
By right-clicking, you can open the context menu.

## 6 Diagnostics

### 6.1 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.



### 6.2 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.

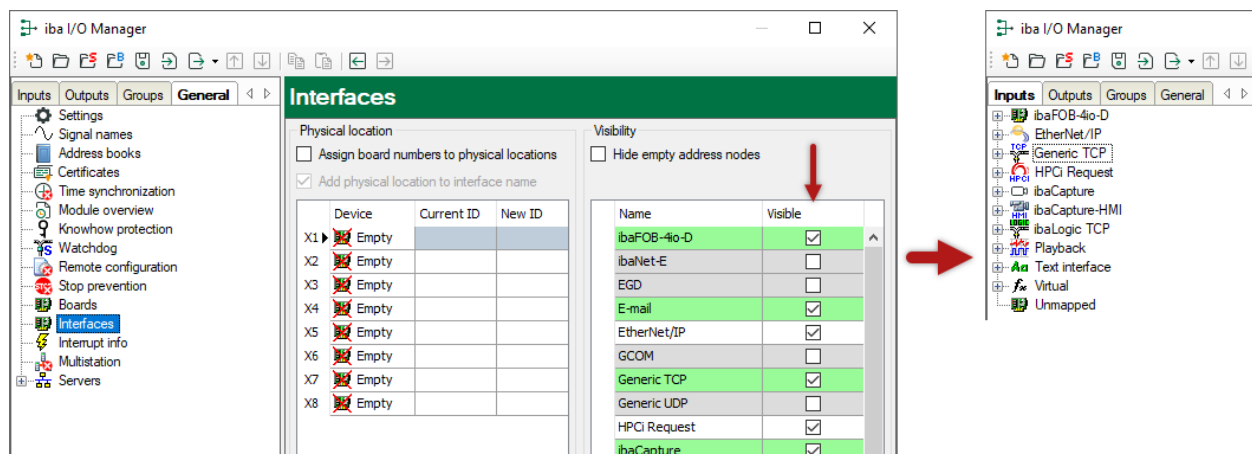
#### Visibility

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:



## 6.3 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.

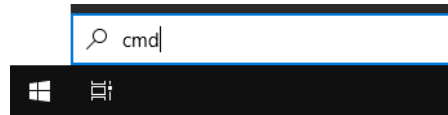
Examples:

- [ethernetipLog.txt](#) (log of EtherNet/IP connections)
- [AbEthLog.txt](#) (log of Allen-Bradley Ethernet connections)
- [OpcUAServerLog.txt](#) (log of OPC UA server connections)

## 6.4 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.

A screenshot of a Windows Command Prompt window titled 'Administrator: Command Prompt'. The window shows the following text:

```
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 TTL=30
Reply from 192.168.1.10: bytes=32 time<1ms TTL=30
Reply from 192.168.1.10: bytes=32 time<1ms TTL=30
Reply from 192.168.1.10: bytes=32 time<1ms TTL=30

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 a Windows Command Prompt window titled 'Administrator: Command Prompt'. The window shows the following text:

```
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.
Reply from 192.168.1.10: Destination host unreachable.
Reply from 192.168.1.10: Destination host unreachable.
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>
```

## 7 Appendix

### 7.1 Error of sequence control

#### 7.1.1 Analyzing the error counters

The two *Fiber optic errors (type 0)* and *Fiber optic errors (type 1)* fields in the *Link info* register of the hardware interface include the statuses of two counters that monitor optical fiber communication on the lowest level. Some errors are always registered in the start-up phase. Afterwards the values have to remain stable.

With an interruption of the FO connection, one or both counters increment persistently. With systems in continuous operation, the long-term monitoring of the counters can provide additional information on the quality of data transmission.

#### 7.1.2 Errors with diagnostic functions

When accessing the rack link memory, the plausibility of the data is checked. Implausible data indicate connection errors, poor or damaged cables or errors in the rack link.

Error	Meaning	Explanation
0	No error	
0x5000	Timeout at access	
0x5010	BG name of the rack link is faulty	No connection to RK, disturbance FO
0x5011	BGT name is faulty	No connection to RK, disturbance FO
0x5012	Version identification faulty	No connection to RK, disturbance FO
0x5013	Sampling time faulty	No connection to RK, disturbance FO
0x5014	Channel name faulty	No connection to RK, disturbance FO
0x5015	Number of channels faulty	No connection to RK, disturbance FO
0x5016	Number of racks faulty	No connection to RK, disturbance FO

#### 7.1.3 Timeout error of ibaPDA

The firmware on the *ibaFOB-SD/ibaFOB-TDC* card is state-controlled. The *ibaPDA* commands are only processed in certain states. If a command is not processed by the firmware due to a coordination error, then *ibaPDA* generates a timeout error message with the command code as error number.

##### Procedure

1. Restart *ibaPDA* acquisition with the activated option *Force reload of driver at next start of the acquisition* (in the I/O Manager's *General* tab – *Settings*).
2. Check whether *ibaFOB-SD/ibaFOB-TDC* is running. (Green LED flashes.)
3. Check whether the connection is OK. (Yellow LED is lit up.)

Error	Meaning
0x04	Timeout with channel log in
0x08	Data request with assignment
0x18	Address book request with assignment
0x0C	Data reading with assignment
0x1C	Technostring reading with assignment

### 7.1.4 Timeout errors of sequence control

The interface within the firmware between the sequence control and the driver for access to the communication memory in the rack link is similar.

The sequence control generates an error message here if the driver is not ready to accept new transmission/reception assignments.

#### Procedure

1. Restart *ibaPDA* acquisition with the activated option *Force reload of driver at next start of the acquisition* (in the I/O Manager's *General* tab – *Settings*).
2. Check whether the connection is OK. (Yellow LED is lit up.)
3. Check whether the optical fiber error counters are constant, see ➤ *Link Info* tab, page 54.

Error	Meaning	Explanation
0x7000	No connection to RK	
0x7001	TO with ZB log-in	No RS after ZB log in (step 5 of sequence control)
0x7002	TO with channel log in	No RS after channel log in (step 7)
0x7003	TO with assignment telegram	No acknowledgment telegram (step 11)
0x7004	TO with data telegram	No acknowledgment telegram (step 11)
0x7005	TO with data telegram	Driver not ready for "data reading" assignment (step 13)
0x7006	TO with data telegram	No RS after start of "data reading" (step 14)
0x7008	TO with data telegram	No RS with cyclic reading (Interrupt routine)
0x700A	TO with assignment telegram	No RS with assignment "Address book request"
0x700B	TO with address book telegram	No RS with reading address book telegram

TO = Timeout, RS = Return signal

## 7.2 Errors of the TDC driver

### 7.2.1 Error classes

Depending on the cause of the error or the necessary reaction on the *ibaPDA* side, the error messages of the SD-/TDC driver can be divided into various classes.

- Temporary status messages, no error;  
if an error is permanently present, then an error of class 2 or 3
- Programming error in the *ibaFOB-SD/ibaFOB-TDC* firmware; please notify iba support
- Error due to inconsistent data, access error via FO to the rack link
- Configuration error on Siemens side
- Configuration on Siemens side does not match iba configuration
- Configuration or programming error on *ibaPDA* side or error in SER04 function blocks

#### Note



The error codes 0x6000 ... 0x6FFF listed in the following sections correspond to those in the SIMADYN D/SIMATIC TDC manual "Diagnostics" section. More precise descriptions and causes can be found there.

The most frequent configuration errors are marked in bold.

### 7.2.2 Errors with ibaPDA log in to the computer coupling

Error	Cl.	Meaning	Explanation	Remedy
0		Not an error, log-in OK		
#0		Login, abort with error, the connectors have the following states: CDM = 0x0000: coupling defective CDA = 0x00FF: coupling blocked NCP = 0x0000: no. of connected racks QTS = 0x0000: coupling not initialized BZS = 0x0000: block switched off ENO = 0xn timer: error number, see below		
0x6AA0	6	<b>Multiple configuration</b>	<b>The connection name is already entered in the computer coupling log-in area. Several connected ibaPDA systems have the same link name.</b>	<b>Change the BG name of the connection in the link list, see ↗ Creating PDA channels, page 35.</b>
0x6AA1	2	CIP name known, but CIP not addressable		

Error	Cl.	Meaning	Explanation	Remedy
0x6AA2	5	CIP name unknown	Names of the rack link component in the Siemens configuration and the <i>ibaPDA</i> configuration do not match.	Check the component name of the rack link component, see <a href="#">↗ Creating PDA channels</a> , page 35.
0x6AA3	2	Impermissible function value of the <code>zbak_cip_suche [search]</code> function (*)	The connector EZU contains the unknown value	
0x6AA5	3	Too many racks logged in	With CS14: 8 slave racks With GDM: 44 TDC racks	
<b>0x6AA6</b>		<b>Component name of the rack link component is unknown</b>	<b>Names of the rack link component in the Siemens configuration and the <i>ibaPDA</i> configuration do not match.</b>	<b>Check the component name of the rack link component, see <a href="#">↗ Creating PDA channels</a>, page 35.</b>
0x6AA7	2	Impermissible function value of the <code>_CS2_anmeldung [login]</code> function	The connector EZU contains the unknown value	
0x6AA8	2	Unknown operating state (*)	The connector EZU contains the unknown value	
0x6AA9	2	SAVE memory too small		
0x6AAA	2	SAVE area unknown		
0x6AAB	2	FOB-SD/-TDC in impermissible sampling time (larger than $2 * @CS1$ sampling time)	Since it is preset in the FOB firmware at 64 ms, the error cannot occur with $@CS1$ sampling time $>32$ ms.	Check configuration at central block <code>@CS1</code> or <code>@SRACK</code> .
0x6AAC	2	No exception memory present		
0x6AAD	2	No reorganization memory present		
0x6AAE	2	Impermissible function value of the <code>zp_fb_init</code> function (*)	The connector EZU contains the unknown value	
0x6AAF	2	Faulty call of the <code>bsspw_korrekturwert [correction value]</code> function		

Error	Cl.	Meaning	Explanation	Remedy
0x6AB0	2	Faulty call of the ampi_asop function		
0x6AB1	2	Wrong coupling type		
0x6AB2	2	3 CS21 already present		
<b>0x6AB3</b>	<b>5</b>	<b>Software version incompatible</b>	<b>Version of the Siemens software does not match the version of the connected racks.</b>	<b>Change the SW version in the "Link info" menu, see ↗ <i>Creating PDA channels</i>, page 35.</b>
0x6AB4	4	Memory sizes do not match	FOB-TDC expects CP52M0 with 2 MBytes	Check TDC configuration.

### 7.2.3 Errors with channel log in

Error	Cl.	Meaning	Explanation	Remedy
0		No error, log-in OK		
0x0031	2	Previous log-in not possible		
0x0032	2	Error with provision of memory		
0x0033	2	Wrong transmission/reception identification		
0x0034	2	Too much partial information		
0x0035	2	Target information not identifiable		
0x0036	2	Target information faulty		
0x0037	2	Channel name too long		
0x0038	2	Address level 1 too long		
0x0039	2	Address level 2 too long		
0x003A	2	Wrong address string		
0x003B	2	@Ziel [target] rack name too long		
0x003C	2	Length of target function block name wrong		
0x003D	2	Length of target interface wrong		
0x003E	2	Target interface information wrong		
0x6031	1	CIP entry found	No connection	Check optical fiber connection

Error	Cl.	Meaning	Explanation	Remedy
0x6032	1	Coupling for initialization released	No connection	Check optical fiber connection
0x6033	3	Coupling release missing	CP52M0 memory is currently being reorganized or is blocked due to access error	New <i>ibaPDA</i> log-in
0x6034	1	DEXDO3 released		
0x6035	3	DEXDO3 release missing	CP52M0 memory is blocked due to access error	New <i>ibaPDA</i> login; check optical fiber connection
0x6036	1	Coupling free, initialization running		
0x6037	1	Semaphore was blocked		
0x6038	1	Semaphore already blocked		
0x6039	1	Channel name known		
0x603A	1	Search not yet concluded		
0x603B	3	Channel name unknown	Subsequent error	
<b>0x603F</b>	<b>1</b>	<b>Channel initialization concluded in faulty manner</b>		
0x6040	3	Data inconsistent	Access error	Check optical fiber connection
0x6050	3	Buffer state invalid		Check optical fiber connection
0x6051	3	Channel state invalid		Check optical fiber connection
0x6052	3	Buffer semaphore already blocked		Check optical fiber connection
<b>0x6053</b>	<b>1</b>	<b>No more space in ZCVWL list</b>	<b>Internal list is full, since with each channel log-in list spaces are occupied but are no longer released. (CIP3 concept!)</b>	<b>Activate <i>ibaPDA</i> "Driver restart" and restart acquisition</b>
0x6054	2	No more memory space in ZCDAT area		
0x6055	2	CIP name known, but CIP not addressable		

Error	Cl.	Meaning	Explanation	Remedy
0x6056	5	CIP name unknown	Component name of the rack link component in the Siemens configuration does not match the <i>ibaPDA</i> configuration.	Check TDC and <i>ibaPDA</i> configuration
0x6057	2	Unknown zbak_cip_suche [search] function value		
0x6058	2	ZINI bus identification invalid		
0x6059	4	Too little memory space	Channel no longer fits in memory	Check TDC projection
0x605A	2	Wrong channel mode	Channel mode is fixedly set in firmware and service building blocks	
0x605B	2	Wrong user data structure	Preset in firmware and service function blocks	
<b>0x605C</b>	<b>4</b>	User data length too short	<b>Possibly after switching STRUC ↔ CFC</b>	<b>Activate <i>ibaPDA</i> "Driver restart" and restart acquisition</b>
0x605D	46	Wrong service type	With TDC-Lite: PROCESS DATA With TDC symbolic request: SERVICE	Check TDC configuration

### 7.2.4 Errors with data transfer

Error	Cl.	Meaning	Explanation	Remedy
0x6000		No error, data transfer OK		
0x6001	5	Transmitter not ready to start	No transmission channel configured with this name	Check SD/TDC configuration
0x6002	5	Receiver not ready to start	No reception channel configured with this name	Check SD/TDC configuration
0x6003	5	No assignment, i.e., buffer empty	Transmitter is too slow or not released.	Check SD/TDC configuration (sampling time)
0x6004	5	Buffer full (with transmitter, hand-shake)	Receiver is too slow or not released.	Check SD/TDC configuration (sampling time)
0x6005	3	DEXDO3 release missing	Initialization function block @CS1/@SRACK missing	Check SD/TDC configuration
0x6006	1	Coupling free, initialization running		
0x6007	1	Semaphore was blocked		
0x6FF0	2	Buffer pointer = 0		
0x6FF1	2	Data length = 0 with writing		

## 7.3 Display of service blocks

### 7.3.1 SER04A/SER05A

ST	Internal block status
0	Waiting for start-up of SER0B
1	Waiting for the synchronization of the channels with <i>ibaPDA</i>
2	Normal state, waiting for request of <i>ibaPDA</i>
3	Stop SER0xB, waiting for feedback
4	Start SER0xB, waiting for feedback
5	Send acknowledgment telegram
6	Fatal error state (see YTS and other connectors)
7	Assignment transfer of address book
8	Assignment read version identification
9	Assignment continuation of address book transmission
10	Assignment repetition of address book transmission
11	Send acknowledgment telegram in case of error

YTS	YTZ	Error
0	0	everything ok
1	frw <sup>4)</sup>	Error when allocating the local storage
2	frw	Error when allocating the common storage (VPORT)
3	frw	Error when reading the own sampling time
4	frw	Error when creating the transmission channel
5	frw	Error when creating the reception channel
6	Status B	Block SER0xB is not in the expected state
7	Sampling time B Ti B	SER04: Non permissible sampling time ( $T_a \leq T_b$ ) SER05: -B connected with wrong sampling time
8	frw	Permanent channel error
9	Mask	Only SER05: At least one SER05B is not connected
10	frw	Temporary channel error
11	No. Bits	More than 32 or 128 Bits requested
12	No. anal.	More than 32 or 128 analog values requested
13	0	No data requested
14	0	Error telegram length request telegram
15	0	Only SER05: Error time slice in the request telegram
20	frw	Error when reading the own rack name
21	frw	Error when reading the own CPU name

<sup>4)</sup> frw = Function return value of a system function

YTS	YTZ	Error
30	frw	Error when creating the address book channel
31	frw	Temporary error with the address book channel
32	Seq.Ctr	Error with address book telegram sequence
33	frw	Permanent error with address book channel
40-45	frw	Error with internal system function
50	0	Timeout, no feedback by SER04B

YTT (dez)	(hex)	Status transmission channel (ACK)
24639	603F	Channel initialized, not in operation, yet
24576	6000	Channel in operation, normal state
		Other status or error, see above

YTR (dez)	(hex)	Status reception channel (CMD)
24677	6001	Channel initialized, not in operation, yet
24576	6000	Channel in operation, reception active
24579	6003	Channel in operation, reception buffer empty
		Other status or error, see above

### 7.3.2 SER04B/SER05B

ST	Internal block status
0	Waiting for initialization
1	Waiting for synchronization of channels with <i>ibaPDA</i>
2	Normal status, no data transfer
3	Data transfer started, start-up phase
4	Data transfer running
5	Fatal error status (see YTS and other connectors)
6	Only SER05B: Waiting for synchronization with SER05A

YTS	YTZ	Error
0	0	Everything ok
1	frw	Error when allocating the local memory
2	frw	Error when reading the own sampling time
3	frw	Error when creating the transmission channel
4	0	Non permissible address on the ADR connection
5	frw	Permanent channel error
6	0	Only SER05B: "Short name" in VPORT
10	frw	Temporary channel error
20	frw	Error when reading the own rack name
21	frw	Error when reading the own CPU name

YTT (dez)	(hex)	Status transmission channel (ACK)
24639	603F	Channel initialized, not in operation
24576	6000	Channel in operation, data transfer active
		Other status or error, see above

## 7.4 Abbreviations

BG	Board, component
BGT	Rack, sub-rack
CFC	Continuous flow chart, graphic programming language
CIP	Communication interface for SD and TDC
CP52M0	GDM – memory component
CP52IO	GDM – interface component
CP52A0	TDC – interface component to GDM
CP53	TDC/SD rack link configurable as master and slave
CS14	SIMADYN D rack link – master
CS22	SIMADYN D rack link – slave
DPR	Dual port RAM, interface <i>ibaPDA</i> ↔ <i>ibaFOB-SD/ibaFOB-TDC</i>
FOB	Fiber optical board, optical fiber interface component
GDM	Global data memory, central communication memory in the TDC system
FO	Fiber optical
PDA	Process data acquisition
PN	SIMATIC TDC processor component, CPU
RK	Rack link, component CS12, CS13, CS14, CP53, GDM
SD	SIMADYN D
STRUC	Graphic programming interface for SIMADYN D
TDC	SIMATIC TDC
TS	Technostring (text channel)
ZB	Central function block, initialization function block for communication

## 8 Support and contact

### Support

Phone: +49 911 97282-14  
Fax: +49 911 97282-33  
Email: [support@iba-ag.com](mailto: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](mailto: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](http://www.iba-ag.com).**