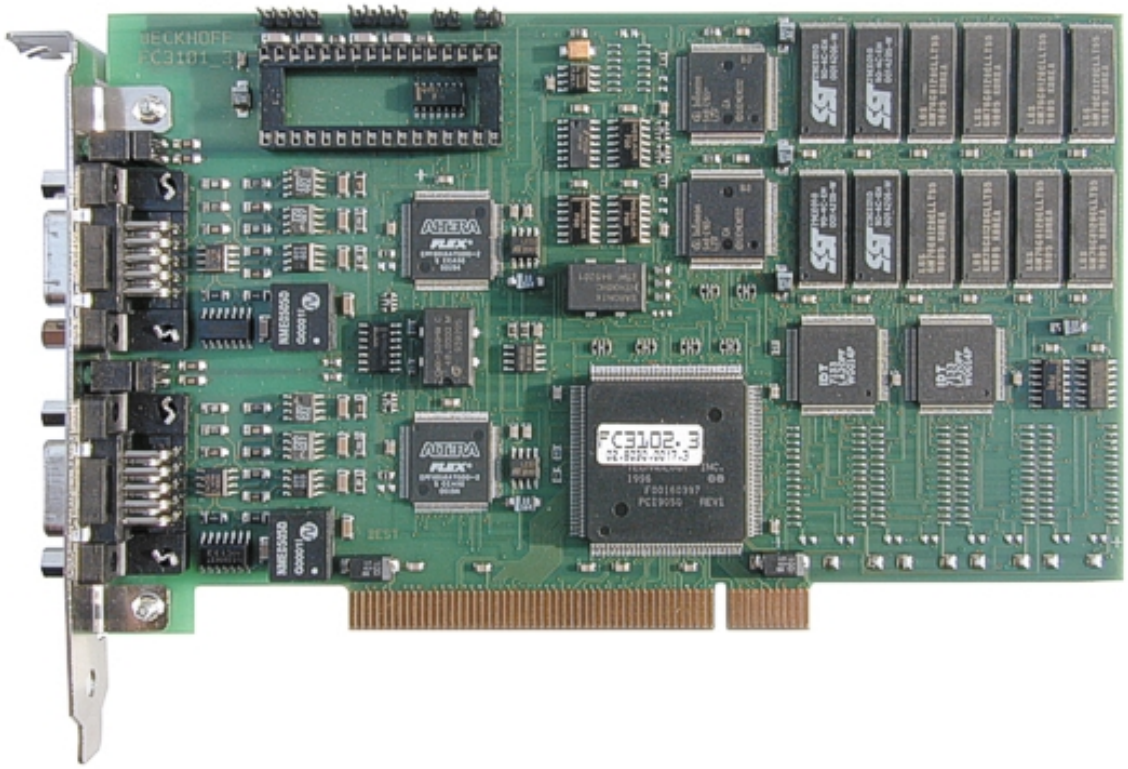

Documentation

FC3101 and FC3102

PCI Cards for PROFIBUS

Version: 3.0
Date: 2017-11-17

BECKHOFF

Table of Contents

1 Foreword

1.1 Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, Safety over EtherCAT®, TwinSAFE®, XFC® and XTS® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, DE102004044764, DE102007017835 with corresponding applications or registrations in various other countries.

The TwinCAT Technology is covered, including but not limited to the following patent applications and patents: EP0851348, US6167425 with corresponding applications or registrations in various other countries.

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany

Copyright

1.2 Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of symbols

In this documentation the following symbols are used with an accompanying safety instruction or note. The safety instructions must be read carefully and followed without fail!

 **DANGER**	**Serious risk of injury!** Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.
 **WARNING**	**Risk of injury!** Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.
 **CAUTION**	**Personal injuries!** Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.
 **Attention**	**Damage to the environment or devices** Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.
 **Note**	**Tip or pointer** This symbol indicates information that contributes to better understanding.

1.3 Documentation Issue Status

Version	Comment
3.0	• Migration
2.2	• Layout and foreword updated
2.1	• Technical data and safety instructions amended
2.0	• Technical description amended
1.0	• First release

2 Product Overview

2.1 Introduction

The functionalities of the Beckhoff PCI card FC310x (as Master [▶ 10] and Slave [▶ 11]) for use under TwinCAT (NCI, PTP, PLC and IO) is described below.
TwinCAT 2.7 does not support all the functions described below. TwinCAT 2.8 on the other hand supports all functions described below, with the exception of redundancy and of the ADS server interface of the FC310x as a slave (these will both be supported as from TwinCAT 2.9).

The following chapters also apply to the PROFIBUS connection for the CX1000 (CX1500-M310 (master) or CX1500-B310 (slave)); the name FC310x then also refers to the CX1500-M310 master or CX1500-B310 slave connection.

2.2 FC310x - Technical Data

Technical Data	FC3101	FC3102
Bus system	PROFIBUS DP (standard), PROFIBUS DP-V1 (Cl. 1+2: acyclic services, alarms), DP-V2, PROFIBUS MC (equidistant)	
Number of fieldbus channels	1	2
Data transfer rate	9.6 kbit/s – 12 Mbit/s [1)]	
Diagnostics	2 LEDs per channel	
Interface to the PC	Plug-and-play PCI interface, 32 bit with 4 kbytes DPRAM per channel	
Bus interface	1 x D-Sub socket, 9-pin, galvanically isolated	2 x D-Sub socket, 9-pin, galvanically isolated
Communication	Master and slave functionality (also mixed)	
Bus devices	per channel: max. 125 slaves with up to 244 bytes input, output, parameter, configuration or diagnostic data per slave	
Process image	Sum max.: 3 kbytes input and output data	
Cycle time	differing DP cycle times per slave are possible using the CDL concept	
Standard driver	standard operating system driver for Intel-compatible NIC	
Real-time driver	TwinCAT driver for real-time Ethernet	
Supply voltage (PCI bus)	5 V	
Current consumption (PCI bus, 5 V)	typ. 600 mA	typ. 1000 mA
Power consumption from PCI bus	< 5 W	
Weight	approx. 45 g	approx. 75 g
Dimensions (W x H x D, without slot panel)	approx. 14 mm x 106 mm x 175 mm	
Permissible ambient temperature range during operation	0°C ... + 55°C	
Permissible ambient temperature range during storage	-25°C ... + 85°C	
Permissible relative air humidity	95 %, no condensation	
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27	
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4	
Installation position	variable	
Approval	CE	

[1)] as of hardware version 10, 6 MBit/s of the FC3101 or FC3102 and their derivatives are no longer supported.

2.3 Hardware Description

One FC3102 PROFIBUS fieldbus card behaves in logical terms in the same way as two FC3101 cards, i.e. all components (with the exception of the PCI interface and the optional NOV RAM) are present in duplicate on the FC3102 card. Each channel of the FC310x card consists of the following components:

- 80165 25 MHz micro-controller
- 512 k RAM
- 256 k Flash
- 4 k DP-RAM
- Fieldbus interface (Altera 6016 with 48 MHz, RS485 bus interfacing, 9 pole SUB D plug)
- 2 LEDs (1 x green, 1 x red)
- 1 four pin and 1 two pin plug for the bootstrap loading mode

The following components are present in single form only:

- PCI interface (PLX9050: 1 interrupt input per channel, 1 interrupt output and 1 chip select signal for DP-RAM and 1 reset line for both channels)
- optional 32 k plug-in NOV RAM, addressed via an additional chip-select line

LED indicators - meanings

State of the FC310x	LED display
RESET, OFFLINE	Red LED on, green LEDs off
STOP	Green LED off, red LED flashing (at 10 Hz) While uploading the bus configuration: red and green LEDs flash (at 10 Hz)
RUN	If the associated TwinCAT task was started, the green LED is on, otherwise the green LED will flash (with 1 Hz) When all boxes are error-free, the red LED will be off, otherwise it will flash (with 1 Hz)

States of the FC310x

RESET, OFFLINE

After power-on, the FC310x is in the RESET state. It exits the RESET state when TwinCAT is started and will enter the OFFLINE state after TwinCAT is stopped (or after a severe bus error). In the OFFLINE state, the FC310x is not active on the bus.

STOP

After TwinCAT has started, or during reading of the bus configuration from the System Manager, the FC310x will enter the STOP state, in which it is active on the bus, but does not yet carry out data exchange (Data_Exchange) with the DP slaves. During the TwinCAT start, and also during an IO reset, the FC310x will be in the STOP state. It will exit the STOP state automatically both during TwinCAT start and during IO reset and will enter the RUN state.

RUN

In the RUN state, the FC310x will automatically establish the DP connections with all configured DP slaves. As soon as the associated TwinCAT task was started, it will communicate with the DP slaves via Data_Exchange. As long as the associated TwinCAT task was not yet started, it will only query the diagnostics. If the associated task is stopped (e.g. PLC STOP, breakpoint in the PLC), the FC310x will automatically enter CLEAR mode (outputs to 0 or slave-specific response, if FailSafe mode is supported). Once the associated task is running again, the FC310x will automatically re-enter the OPERATE mode (all outputs at the values set by TwinCAT).

3 FC310x as master

3.1 Master

As master, the PROFIBUS DP, PROFIBUS DPV1, PROFIBUS DPV2, S5-FDL-AGAG communication and the PROFIDRIVE-PKW Interface protocols are supported.

PROFIBUS DP

A summary of the PROFIBUS-DP master functions follows:

Function	Description
Standard DP	The PROFIBUS DP [▶ 11] chapter describes the necessary steps for establishing a DP connection (Set_Prm - parameter, Chk_Cfg - configuration) and for the exchange of user data (Data_Exchange).
Task synchronization	The Synchronization [▶ 12] chapter describes how the TwinCAT task is synchronized with the PROFIBUS cycle.
Slave priorities	The slaves can receive telegrams with differing cycle times. The necessary settings are described in the Slave Prioritization/Multiple DP Cycles [▶ 14] chapter.
Multiple DP cycles	In order to receive the most recent possible inputs when the task cycles are long, a number of DP cycles can be carried out for each task cycle, as described in the chapter on Slave Prioritization/Multiple DP Cycles [▶ 14].
Diagnostics	The diagnostic [▶ 16] facilities are described in this chapter.
Error Reactions	It is possible for different error reactions [▶ 16] to be set in the event of a fault (a slave fails or the task is stopped).
Sync/Freeze	Activation of the sync and freeze commands is described in the Sync/Freeze [▶ 15] chapter.
Upload Configuration	The slaves connected to the PROFIBUS can be read via Upload Configuration [▶ 34].
Master redundancy	The Master redundancy [▶ 38] chapter describes the settings required to have a second master with the same configuration configured as a standby master (as from TwinCAT 2.9).

PROFIBUS DPV1

A summary of the PROFIBUS-DPV1 master functions follows:

Function	Description
MSAC_C1	The MSAC_C1 [▶ 31] connection is established along with the cyclic connection. The Read, Write and Data_Transport services are supported.
MSAC_C2	The MSAC_C2 [▶ 31] connection is established independently of the cyclic connection, and can also be used by a second master (while the first one is communicating with the slave over the cyclic MSAC_C1 connection). The Initiate, Abort, Read, Write and Data_Transport services are supported.

PROFIBUS DPV2

A summary of the PROFIBUS-DPV2 master functions follows:

Function	Description
Equidistance	The DPV2 equidistance functionality is described in the PROFIBUS MC [▶ 25] chapter.

S5-FDL-AGAG Communication

S5-FDL-AGAG communication is described in the S5-FDL [▶ 36] chapter.

PROFIDRIVE-PKW Interface

The PROFIDRIVE PKW protocol [▶ 35] is implemented in the PROFIBUS master, and can be used by means of acyclic ADS calls.

3.2 PROFIBUS DP

Standard DP Operation

In order to configure standard DP operation, proceed as follows in the TwinCAT System Manager:

Configure the DP Master

It is first necessary to configure a "PROFIBUS Master FC310x, PCI" I/O device (selecting "I/O devices" with right click, and then selecting "Append Device"). Find the corresponding channel on the "FC310x" tab (see FC310x tab under TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) ("Search" button) and adjust the baud rate, if necessary (the default value is 12 Mbit/s).

Add DP slaves

The Beckhoff slaves or third-party devices are to be configured. (All slaves whose GSD file is stored in subdirectory PROFIBUS of the System Manager are displayed automatically (sorted by manufacturer). Select the "General PROFIBUS Box (GSD)" under Miscellaneous to integrate other GSD files.

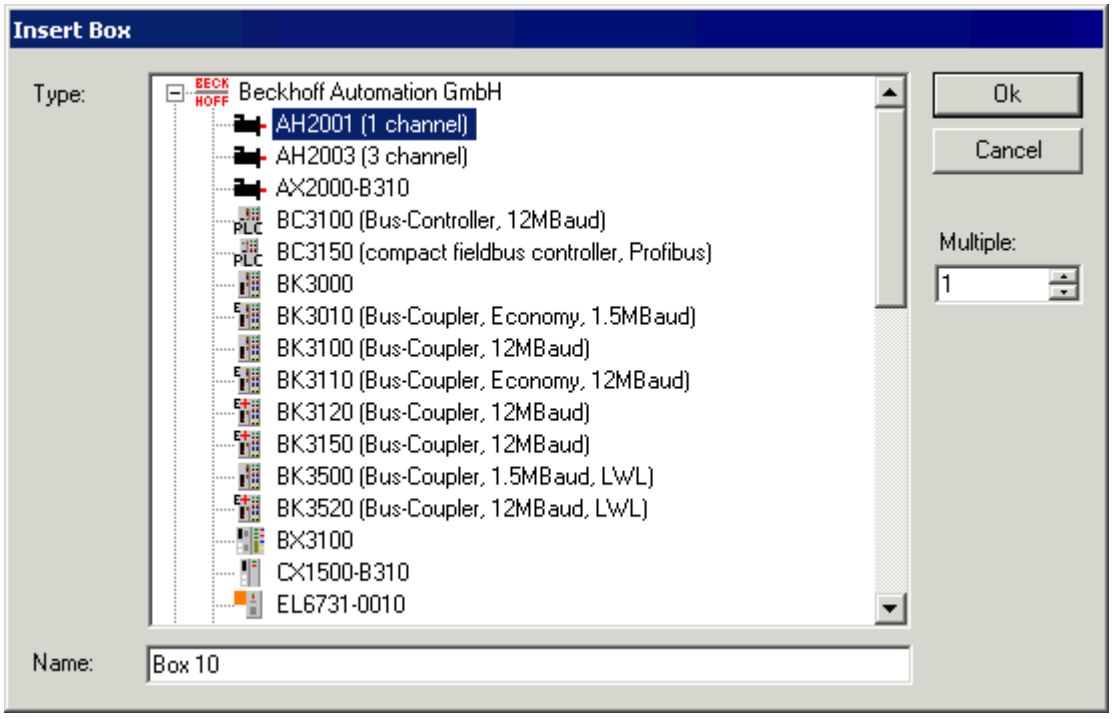

Fig. 1: Adding a DP slave

In the case of modular slaves, it is then still necessary to add the terminal/IL modules (for Beckhoff slaves) or the DP module (for third-party devices).

System Start

TwinCAT Configuration Mode (from TwinCAT 2.9)

For the TwinCAT Configuration Mode, it is only necessary to exchange data with the configured slaves. To do this, TwinCAT Configuration Mode is started, and configuration of the DP master is activated using the "Reload Devices" button on the toolbar. After this, the data relating to the configured slaves can be read and written on the associated Variables tab from the System Manager.

TwinCAT Run-Mode

For the TwinCAT Run-Mode, it is now necessary for at least one variable of the PROFIBUS master or of the configured slave to be linked to a task. The project is then to be stored in the registry, and the TwinCAT system then started in Run-Mode. Data exchange with the slaves is not carried out until the associated task is started. If a number of tasks are linked with the PROFIBUS master or with the configured slave, then whichever task has the highest priority must be started in order for data to be exchanged with the slave.

Bus parameters

TwinCAT 2.8: The PROFIBUS DP Bus parameters are to be found on the device's PROFIBUS [▶ 42] tab, and should only be modified by experienced users.

TwinCAT 2.9: The PROFIBUS DP bus parameters can be found in the Bus Parameters [▶ 49] dialog, which can be selected via the FC310x [▶ 48] tab (Bus Parameter (DP) button). They should only be modified by experienced users.

3.3 Synchronization

3.3.1 Overview

In TwinCAT RunMode, the DP master is always synchronized with the highest priority task with which the variables are linked. Once the mapping was created, the cycle time of the corresponding task is displayed under **Cycle Time** on the **"FC310x"** tab (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the master. The task has a setting to indicate whether the "I/O at the task start" should be updated or not.

I/O at task start

If the setting "I/O at task start" has been selected using the checkbox, which is the default setting for the NC task, then a check is made before the task is started as to whether the previous DP cycle has been completed. The inputs and outputs (the outputs being those from the previous task cycle) are copied, and the DP cycle is started. In the example, the task cycle time is 2 ms, and real-time resources are 80%:

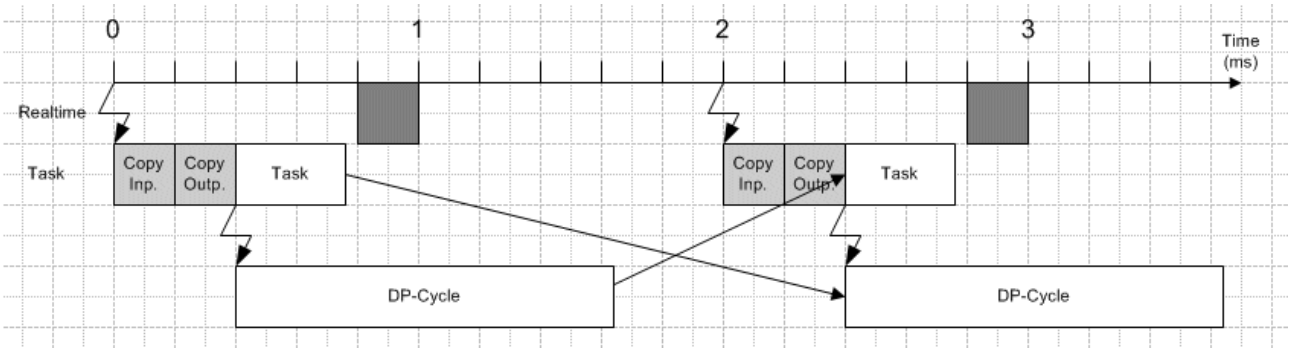

Fig. 2: I/O at task start and real-time resources not exceeded

If, in the previous example, the copying of the inputs and outputs and the task computing time exceeds 0.8 ms, then NT will interrupt execution of the task, because 80% of real-time resources has been reached:

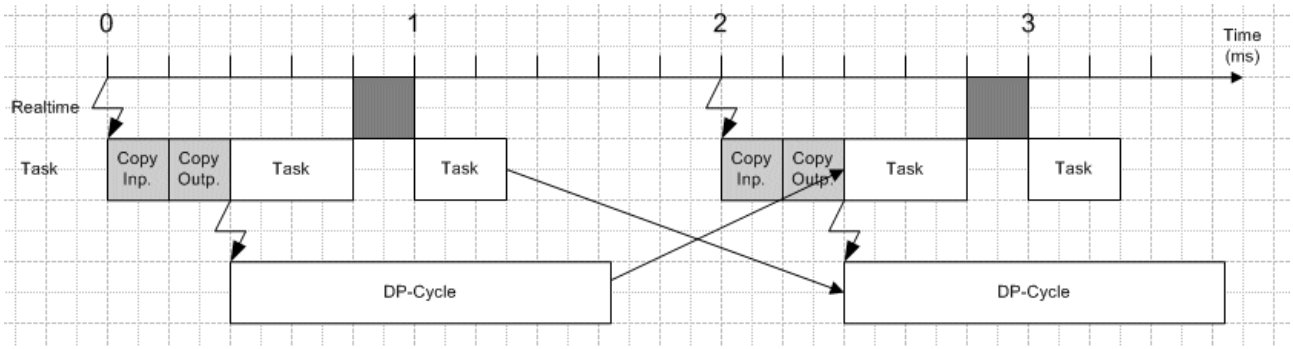

Fig. 3: I/O at task start and real-time resources exceeded

This case would still not be a problem, because the DP cycle was completed within the available time. If "I/O at task start" is not selected, then the process is somewhat more critical, as is described below.

I/O not at task start

If the setting "I/O at task start" is not selected (checkbox) for the task (default for PLC task), the system checks before the task starts whether the previous DP cycle was completed, and the inputs are copied. After this the task is processed, and at the end of the task the outputs are copied and the DP cycle is started. In the example, the task cycle time is 2 ms, and real-time resources are 80%:

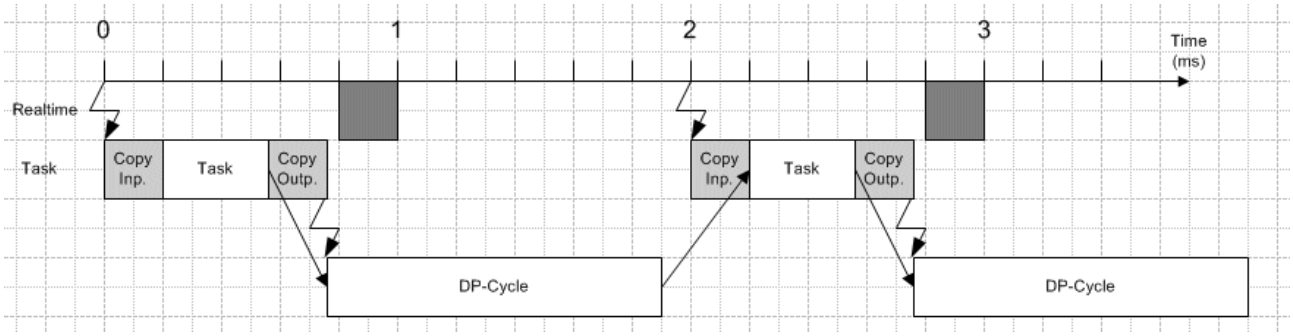

Fig. 4: I/O not at task start and real-time resources not exceeded

In the event of "I/O not at task start" the task and the PROFIBUS have to share the bandwidth. Exceeding of the real-time resources therefore has a much stronger effect than for "I/O at task start":

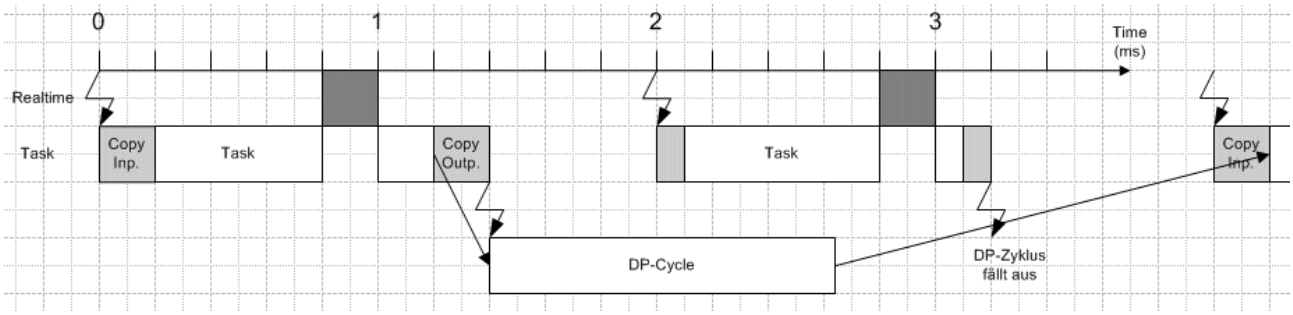

Fig. 5: I/O not at task start and real-time resources exceeded

In the case described, the DP cycle starts later, and is no longer finished within the time available before the following task cycle begins. The effect of this is that it is seen before the task is executed that the previous DP cycle has still not been completed. It follows that inputs are not copied before starting the task, so that the task computes with the old inputs; after the task has been processed no outputs are copied, nor is the DP cycle restarted, so that a DP cycle is omitted. The omission of a DP cycle can be detected with the **CycleCounter**, as described in the Master Diagnostics [▶ 20] chapter.

Comparison of I/O at task start and I/O not at task start

An advantage when "I/O at task start" is selected is that the task and the DP cycle do not have to share the available bandwidth, and that the DP cycle starts very regularly, any jitter being the TwinCAT jitter. If "I/O not at task start" is selected, then it is more likely that a DP cycle will be omitted; the regularity of the start times of the DP cycles depends additionally on the jitter in the task processing. The disadvantage of the setting "I/O at task start" is that the dead time (system response time) increases.

3.3.2 Slave Prioritization/Multiple DP Cycles

Distribution of the DP slaves over a number of DP cycles (prioritization of the slaves)

Slaves can be prioritized so that the DP cycle time can be kept as short as possible in systems in which a few slaves must be polled very rapidly, whereas a larger cycle time would be adequate for other slaves. It is possible to specify for each slave in what multiple of cycles (**Divider** under **Data-Exch Poll-Rate** on the slave's Features [▶ 59] tab) it will be polled. Distributing the polling is then helpful, as, for instance, in the case illustrated below where there are four slaves, each of which is only to be addressed in every second cycle. It is possible to make settings so that two slaves will be polled in one cycle and the other two slaves in the other cycle, so that the overall DP cycle time is kept as constant as possible. For this purpose, the Features [▶ 59] tab for the slave offers the **Modulo** setting under **Data-Exch Poll-Rate**. In the illustrated case, slaves 3 and 5 have **Modulo** 0 while slaves 4 and 6 are given **Modulo** 1. The current modulo can be read in the **ActualModulo** variable which is described in the Master Diagnostics [▶ 20] chapter.

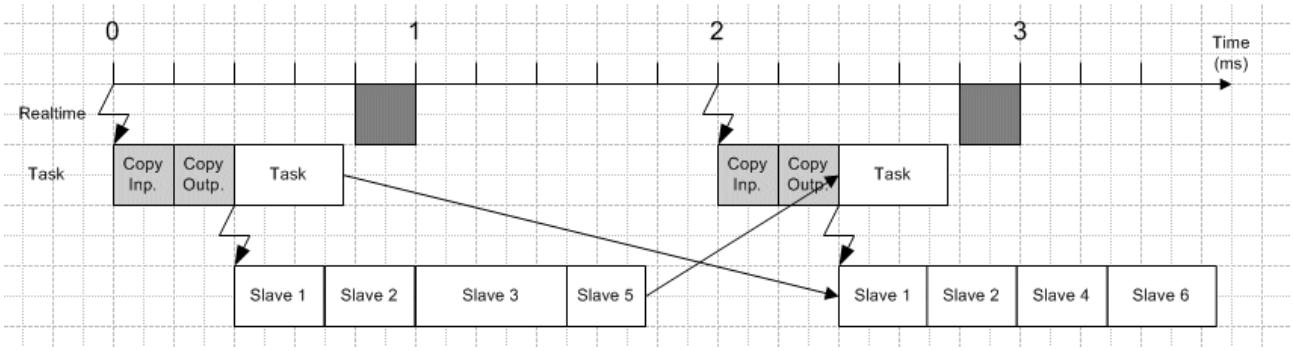

Fig. 6: Allocation of DP slaves to several DP cycles

Multiple DP Cycles in one Task Cycle

If the task cycle time is more than twice as long as the DP cycle time, it is possible for a number of DP cycles to be carried out within a single task cycle, in order to acquire the most up-to-date input data possible. Based on the **Number of DP cycles per task cycle** factor, which can be set on the **"FC310x"** tab of the master (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]), a timer with the cycle time (task cycle time/number DP cycles per task cycle) is started when the first DP cycle starts, through which further DP cycles can then be started. It is, however, necessary to check that the last DP cycle is completed in good time (before the next task start), since otherwise one DP cycle will fail (or possibly more than one, depending on the ratio expressed in the number of DP cycles per task cycle), as described in the Synchronization [▶ 12] chapter.

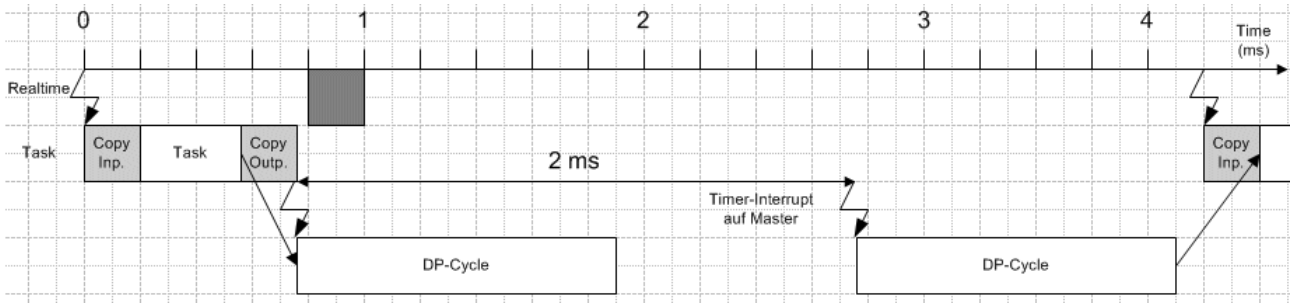

Fig. 7: Multiple DP Cycles in one Task Cycle

Multiple data samples within one task cycle

The two functionalities just described can now be combined in order, for instance, to give one or more slaves in a 2 ms cycle new data every 1 ms, or to obtain new data from the slave in order to achieve better regulation quality. In this case, settings are made under **Additional Data_Exchange Samples** on the slave's Features [▶ 59] tab instead of under **Data-Exch Poll-Rate** (as described above).

In the sample shown below, first set the factor **Number of DP cycles per task cycle** in the **"FC310x"** tab (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the master to 2. So that the task can send 2 different values to the slave, or is able to receive 2 different values from the slave, the appropriate slave is to be entered into the System Manager twice. All settings, with the exception of **Modulo** under **Additional Data_Exchange Samples** on the slave's Features [▶ 59] tab, must be the same. A 0 is entered here for the corresponding slave in one of the boxes and a 1 in the other box in the System Manager. The **Multiplier** under **Additional Data_Exchange Samples** is to be set for this slave in both boxes to 2, so that each of the two boxes that have been entered is only polled in every second DP cycle (the slave is in fact polled in each DP cycle, as it is entered twice). For all other slaves that are only to be polled once within the task cycle (and which of course are only therefore entered once in the System Manager), the **Multiplier** under **Additional Data_Exchange Samples** is also set to 2. **Modulo** under **Additional Data_Exchange Samples** can now be used to distribute these slaves over the two cycles. A slave that is polled twice but which is only to have one variable image in the task is only to be inserted once; the **Multiplier** would be set to 1, and **Modulo** to 0.

In the present example, slaves 1 and 2 would each be entered into the System Manager twice, with these settings:

- Additional Data_Exchange Samples/Multiplier = 2
- Additional Data_Exchange Samples/Modulo = 0 or 1

Slaves 3 and 5 would only be entered into the System Manager once, and would have these settings:

- Additional Data_Exchange Samples/Multiplier = 2
- Additional Data_Exchange Samples/Modulo = 0

Slaves 4 and 6 would also only be entered into the System Manager once, with these settings:

- Additional Data_Exchange Samples/Multiplier = 2
- Additional Data_Exchange Samples/Modulo = 1

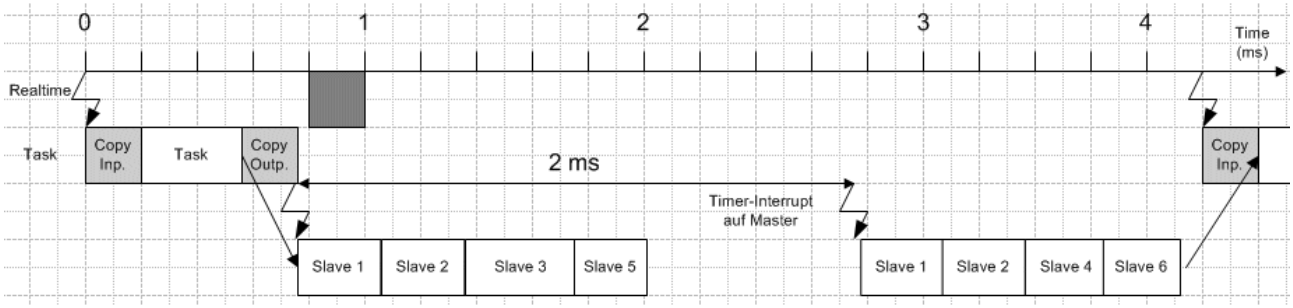

Fig. 8: Multiple data samples within one task cycle

For slaves 1 and 2, the variables are present twice (in each case 2 boxes in the System Manager). The variables associated with the box where Additional Data_Exchange Samples/Modulo is set to 0 are sent or received first.

3.3.3 Sync/Freeze functionality

Sync is used for the simultaneous outputting of outputs for several slaves. Freeze is used for reading in inputs from several slaves simultaneously.

The process in TwinCAT with FC310x and Bus Couplers (in K-bus synchronous mode) would therefore be as follows (see the Synchronization [▶ 12] chapter):

- The outputs are written at the beginning (I/O at the start of the task) or the end (I/O not at the start of the task) of the task cycle

- This will start the PROFIBUS cycle
- A Sync/Freeze telegram is sent at the start of the PROFIBUS cycle
- This will cause the Bus Couplers to start a K-bus cycle with the outputs from the last task cycle and transfer the inputs from the last K-bus cycle
- The master will then send the current outputs to each slave and pick up the transferred inputs
- The inputs are read at the start of the next task cycle
- etc.

Outputs and inputs are therefore always one cycle old.

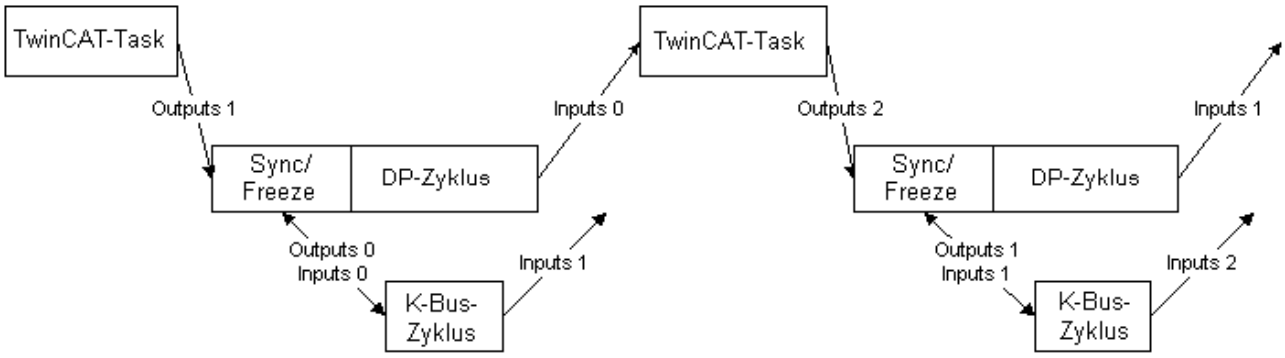

Fig. 9: Sync/Freeze functionality

On the FC310x, set the **Operation Mode** on the **"FC310x"** tab (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the master to "DP/MC (equidistant)". At the boxes to be operated via Sync/Freeze, click the flag **Sync/Freeze enable** on the Profibus [▶ 58] tab. The master always uses group 1 for the Sync/Freeze synchronization.

3.4 Diagnostics

3.4.1 Overview

The Error reactions [▶ 16] section describes the reactions that will be given to slaves that do not answer or that answer incorrectly, to a PLC stop or at start-up. The Slave Diagnostics [▶ 23] chapter explains how diagnostic data provided by the slave and slave statistics can be read, while the Master Diagnostics [▶ 20] chapter describes general diagnostic information and statistics.

3.4.2 Error Reactions

Failure of a slave

If a slave does not respond or the response is faulty, the master repeats the telegram several times until the **Max Retry limit** is reached (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Bus Parameter [▶ 49] dialog). If a faulty telegram is received, the master repeats immediately, in the event of a timeout the master waits for a response from the slave until the **Slot time** has elapsed (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Bus Parameters [▶ 49] dialog). At 12 Mbit/s, a slot time of 1000 bit-periods and a max retry limit of 4 (default values) then a Data_Exchange telegram will delay sending the following telegram by

TDelay = (4 x ((15 + number of outputs) x 11 + 1000) - (15 + number of inputs) x 11)/12 µs

The DpState [▶ 25] of the slave is set to 0x02 (timeout) or 0x0B (faulty telegram). The effect on the DP connection can be set (see below).

Normal DP cycle (12 Mbit/s, 5 slaves, 20 bytes I, 20 bytes O per slave on average)

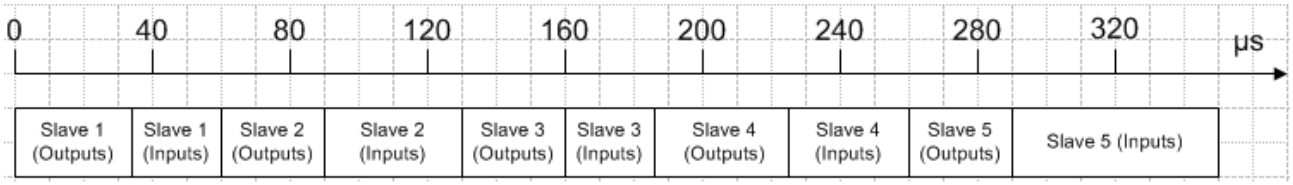

Fig. 10: Normal DP cycle

first occurrence of a faulty DP cycle (slave 3 does not answer)

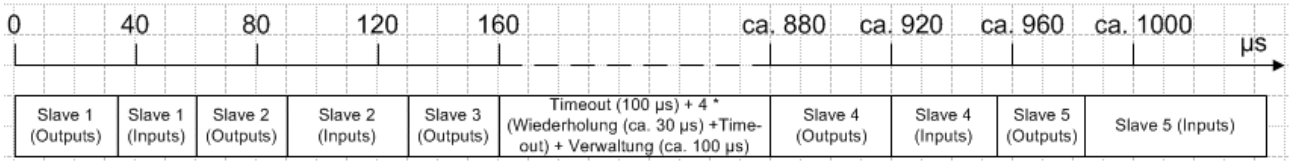

Fig. 11: First occurrence of a faulty DP cycle

subsequent DP cycles (slave 3 no longer in the polling list)

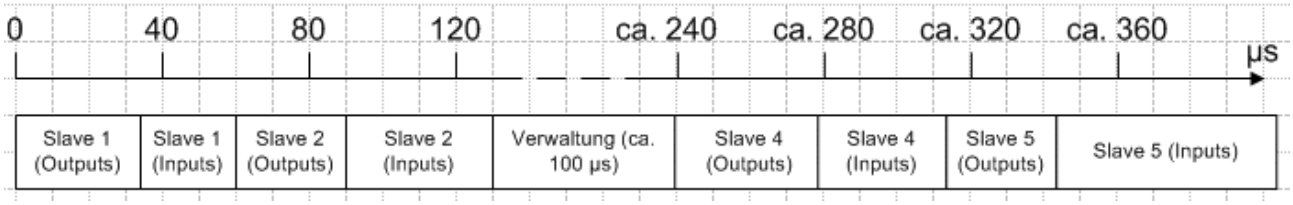

Fig. 12: Following DP cycles

It can still happen that the slave answers incorrectly (e.g. because, as a result of a local event on the slave, the DP connection has been removed). In this case, the telegram is not repeated, but the system continues by sending the next telegram. The DpState [▶ 25] is set to a value other than 0, the slave is removed from the polling list and is no longer addressed in the following DP cycle (which means that the time at which the following telegram is sent changes), until the DP connection can be established again.

Reactions in the master

The master's reactions can be set differently for each slave (see the tab for the slave's Features [▶ 59]).

Effect on the DP connection (NoAnswer reaction) if the slave either does not answer or does not answer correctly

This specifies whether the DP connection to the slave should be removed immediately in the absence of a correct reception telegram, or only after the DP watchdog time has elapsed (see the slave's PROFIBUS [▶ 58]tab).

1. If the DP connection is to be removed immediately (Leave Data Exch, default setting) the slave is removed from the polling list and is no longer addressed in the following DP cycles until the DP connection is established once again. In order to re-establish the DP connection to the slave, at least 7 telegrams are sent, and the process generally requires at least 10-20 ms.
2. If the DP connection is only to be removed when the slave has not answered (or not answered correctly) within the DP watchdog time (Stay in Data-Exch (for WD-Time)), a further attempt is made in the next polling cycle to address the slave, but if the slave does not answer, a repeat is not sent.

The "Stay in Data-Exch (for WD-Time))" (2.) setting makes sense if the PROFIBUS cycle is to continue to operate at the most regular possible period even if a slave fails, and if the failure of a slave for one or more cycles can be tolerated (e.g. in the DP/MC (Equidistant) [▶ 25] operation mode). In this case the DP watchdog time for the slave should be set according to the tolerable outage time of the slave, and the **Max Retry limit (DX)** (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Bus Parameters [▶ 49] dialog) should be set to 0.

Normal DP cycle (12 Mbit/s, 5 slaves, on average 20 bytes I, 20 bytes O per slave) in mode "Stay in Data Exch (for WD time)"

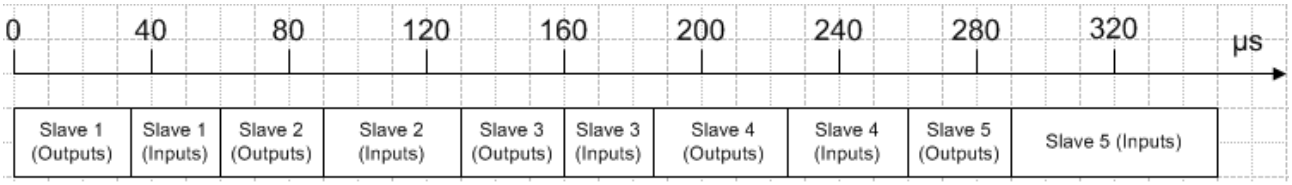

Fig. 13: Normal DP cycle for *Stay in Data Exch (for WD time)*

first faulty and subsequent DP cycles in the "Stay in Data-Exch (for WD-Time)" mode (slave 3 does not respond)

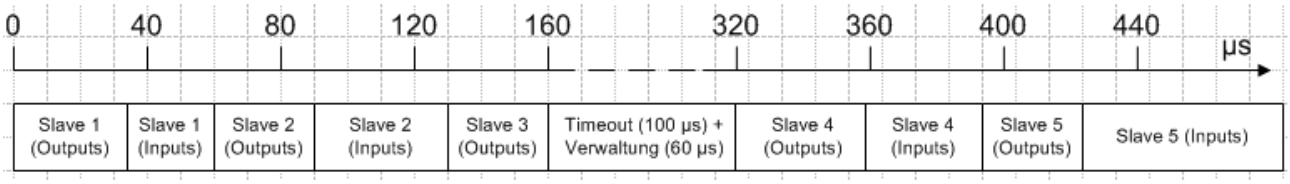

Fig. 14: First faulty and subsequent DP cycles for *Stay in Data Exch (for WD time)*

Changes of the slave's input data if the slave does not respond correctly

Here you can specify whether the input data of the slave are set to 0 if the slave fails ("Inputs will be set to 0", default setting) or whether the old value should be retained ("No changes"). In either case the DpState [▶ 25] of the slave is set to value other than 0, so that the task can always recognize whether or not the data is valid. If a slave gives a faulty answer, the input data is always set to 0, independently of the setting of **Changes of the Input Data**.

Setting the slave's restart behavior if the DP connection to the slave is removed

This specifies whether the DP connection to slave whose DP connection has been removed is automatically re-established, or whether this should be done manually as a result of a call to ADS-WriteControl (see ADS-Interface [▶ 28]).

The reaction of the master if the DP connection to the slave is removed

This specifies whether removing the DP connection to a slave has no other effects (No Reaction, the default setting), or whether the master should enter the STOP state, thus removing the DP connections to all the slaves.

Effect on the state of the master (Clear mode), if the DP connection to the slave is removed

Clear mode (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog) can be used to specify that the master should switch to or remain in "Clear" state, as long as at least one MC slave (setting: "Only MC slaves") or any slave (setting: "All slaves") does not respond correctly (i.e. has a DpState [▶ 25] not equal 0).

The **Reaction of the Master** setting (see the slave's Features [▶ 59] tab), which was described in the previous chapter, has priority over the **Auto-Clear mode**, so that when an appropriately set slave fails, the Master enters the STOP state.

Failure of the master

Monitoring in the PLC/IO task

In the event of persistent bus faults, the DP cycle also may extend up to 100 ms, even with 12 Mbit/s. In order to monitor the DP master, there is a status variable CycleCounter, and this can be linked in the PLC (see the Master Diagnostics [▶ 20] chapter). This variable is incremented by the master after each DP cycle, so that failure of the master can be detected by monitoring this variable in the PLC.

Monitoring in the slave

In order to monitor failure of the master and data transmission on the PROFIBUS, a **watchdog** (see the box's PROFIBUS [▶ 58] tab) can be activated (default setting: watchdog activated with 200 ms). The **Watchdog** must be set to at least twice the maximum **Estimated Cycle Time** and **Cycle Time** (see **"FC310x"** tab (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the master).

Failure of the PLC/IO task

A distinction is made between a PLC stop, reaching a break point and a task stop (the I/O task and NC task are only stopped when the entire system stops). In the case of a PLC stop, the output data is set to 0 by the PLC, whereas when a breakpoint is reached the data initially remains unchanged.

In the master, the task is monitored with a monitoring time (TwinCAT 2.8: according to the setting **Clear Delay** x task cycle time on the PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: according to the setting **Task Watchdog** x task cycle time in the Fault Settings [▶ 51] dialog). If no new data transfer takes place within this monitoring time, the master switches to "Clear" state according to the setting **Reaction on PLC Stop** or **Reaction on Task Stop** (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog) (outputs are set to 0 or safe state (Fail_Safe = 1 in the GSD file, default setting) or remains in "Operate" state (outputs retain the last value). The "Operate" setting is valuable when the outputs should not be cleared when a breakpoint is reached in the PLC. However, if the PLC stops, the outputs will still be set to 0 (by the PLC), even if the master remains in the "Operate" state. It should, however, be noted that the outputs will only be zeroed if the previous DP cycle is completed in time (see the Synchronization [▶ 12] chapter). It should therefore only be set during the commissioning phase.

Failure of the host

To monitor a host crash (e.g. blue screen on a PC), a **watchdog time** can be set (TwinCAT 2.8: see FC310x [▶ 40] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog). If this watchdog timer elapses, the master enters the OFFLINE state, so that the DP connections to all the slaves are removed, and the master logs off from the PROFIBUS, ceasing to carry out bus accesses.

Start-up behavior

The DP connections to all the slaves are established when the TwinCAT system starts up. Until the highest priority task that is involved has not been started, the master still does not send any Data_Exchange telegrams even after the DP connection has been established, and sends only diagnostic telegrams. As soon as the highest priority task has transferred data once, and the DP connection for the corresponding DP slave has been established, the master cyclically (with the highest priority assigned task) sends one Data_Exchange telegram to each of the corresponding slaves.

In addition, the **Operate Delay** and **Clear Mode** settings (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog) can be used to specify when the master switches from "Clear" state (outputs are set to 0 or safe state (Fail_Safe = 1 in the GSD file)) to "Operate" state (outputs correspond to the outputs transferred by the task). The **Operate Delay** specifies the minimum length of time for which the master should remain in the "Clear" state following the first transfer of data. As has been described above, the **Clear mode** specifies whether the master changes into or remains in the "Clear" state if a slave in general or an MC slave in particular fails.

Shut-down behavior

The reaction to the stopping of the TwinCAT system is exactly the same as has been described above in the "Failure of the Host" chapter; the DP connections to all slaves are removed, and the master logs itself off from the bus.

3.4.3 FC310x - Master Diagnostics

Diagnostic Inputs

The master possesses a variety of diagnostic variables that describe the state of the card and of the Profibus. They can be linked in the PLC:

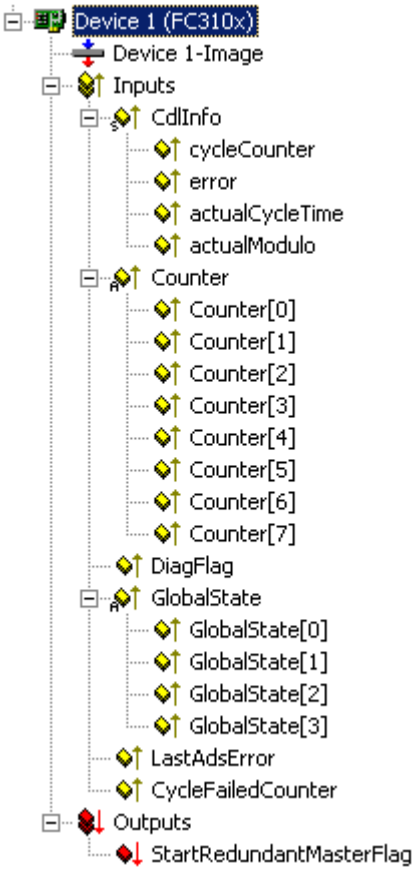

Fig. 15: Diagnostic variables of the PROFIBUS master

CdlInfo:

CdlInfo.error: Shows the number of slaves with which data exchange could not take place in the last cycle. The BoxState of the slaves should only be checked if this value is not equal to 0.

CdlInfo.cycleCounter: Is incremented at the end of each PROFIBUS cycle in order that this variable can indicate whether the last cycle was completed before the task was started.

CdlInfo.actualCycleTime: Shows the current cycle time in 4/25 µs. This variable is updated only when all slaves are involved in the data exchange (also when CdlInfo.error is 0).

CdlInfo.actualModulo: Indicates the current modulo. This variable is only of significance if the slaves are prioritized (see the Slave Prioritization/Multiple DP Cycles [▶ 14] chapter)

Counter: Used for the Redundancy mode [▶ 38]

DiagFlag: Indicates whether the card's master diagnostic information has changed. It can then be read via ADS [▶ 28] by the control program, after that the "DiagFlag" variable is reset.

GlobalState: GlobalState[0] indicates the state of the FC310x, GlobalState[1-2] indicate global bus statuses, while GlobalState[3] is reserved for extensions:

RESET (1): Card router not started (after start-up of the PC).

INIT (2): Router started but card not active on PROFIBUS.

STOP (3): Card active on PROFIBUS, but no cyclic data exchange.

STOPPING (4): Card ends cyclic data exchange.

RUN (0): Card in cyclic data exchange.

GlobalState[1] counts the detected bus errors (as from FC310x, version 1).

GlobalState[2]: Bit 0 is set, if no 11 bit idle time is detected on the PROFIBUS (-> check cabling), bit 1 contains the operation mode CLEAR (bit 1 = 1) or OPERATE (bit 1 = 0), the other bits (2..7) are reserved for expansions (from FC310x, version 1).

GlobalState[3] is reserved for expansions.

CycleFailedCounter: This counter shows how often the FC310x PROFIBUS cycle was unready at the start of the TwinCAT task.

StartRedundantMasterFlag: Is used for the Redundancy mode [▶ 38]

Master Diagnostics Data

The master diagnostic data can be read by ADS [▶ 28]:

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	200
IndexGroup	0x0000F100
IndexOffset	Offset within the diagnostic data
Length	Length of the diagnostic data that is to be read
Data	Diagnostic data

The master diagnostic data has the following structure:

Offset	Description
0 -125	BusStatus list, one byte per station address 0-125, which contains the station status (see BoxState for PROFIBUS boxes, additional for stations that are not configured: 0x80 - not available, 0x81 - slave, 0x82 - master not ready for token ring, 0x83 - master ready for token ring, 0x84 - master in token ring)
126 - 127	reserved
128 - 135	State of the FC310x (->GlobalState)
136 - 137	Send error counter for all sent telegrams
138 - 139	Receive error counter for all received telegrams
140 - 255	reserved for extensions
256 - 257	Sync Failed counter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
258 - 259	Cycle Start Error Counter, counts up one when the PROFIBUS cycle is restarted before the old cycle is complete (intercepted by the TwinCAT-IO driver, only possible with customized drivers)
260 - 261	Time Control Failed counter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
262 - 263	reserved for extensions
264 - 265	Minimum reload value of real-time timer
266 - 267	Maximum reload value of the real-time timer (max. FCxxxx jitter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9)) = max. reload value - min. reload value)
268 - 269	PLL Overflow counter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
270 - 271	PLL Underflow counter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))

Tclo diagnostic data

The Tclo driver also generates diagnostic data that can be read, activated, deactivated and reset by ADS. It is, however, deactivated by default. They are enabled if the tab EquiDiag [▶ 46] or GeneralDiag [▶ 44] (TwinCAT 2.8) MC-Diag [▶ 46] or DP-Diag [▶ 54] of the device is selected and disabled if the tab if deselected.

Activation, deactivation and resetting the Tclo diagnostic data

ADS-Write parameters	Meaning
Net-ID	PC Net-ID
Port	300
IndexGroup	0x00005000 + Device-Id (device's General tab)
IndexOffset	0xFFFFF100
Length	2
Data	0: Deactivation of the Tclo diagnostic data 1: Activation of the Tclo diagnostic data 2: Resetting the Tclo diagnostic data

Reading the Tclo diagnostic data

ADS-Read parameters	Meaning
Net-ID	PC Net-ID
Port	300
IndexGroup	0x00005000 + Device-Id (device's General tab)
IndexOffset	0xFFFFF100
Length	Length of the Tclo diagnostic data
Data	Tclo diagnostic data

The Tclo diagnostic data has the following structure:

Offset	Description
0 -3	Max. TwinCAT jitter (in 100 ns, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
4 - 7	Min. mapping time (in 100 ns, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
8 - 11	Min. mapping time (in 100 ns, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
12 - 15	Max. FC310x jitter (in FC310x ticks, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
16 - 19	CycleWithNoDxch counter (see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
20 - 23	CycleWithRepeat counter (see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
24 - 27	Max. repeater/cycle (see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
28 - 31	Actual cycle time (in 4/25 µs, see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
32 - 35	Max. cycle time (in 4/25 µs, see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
36 - 39	Max. cycle time (in 4/25 µs, see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
40 - 43	RealFailedCycle counter (see tab GeneralDiag [▶ 44] (TwinCAT 2.8) or DP-Diag [▶ 44] (TwinCAT 2.9))
44 - 47	EquiCycleNoDxch counter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
48 - 51	EquiCycleRepeat counter (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
52 - 55	Max. Repeats/Equi-Cycle (see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
56 - 59	Actual Equi-Cycle-Time (in 4/25 µs, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
60 - 63	Max. Equi-Cycle-Time (in 4/25 µs, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))
64 - 67	Min. Equi-Cycle-Time (in 4/25 µs, see tab EquiDiag [▶ 46] (TwinCAT 2.8) or MC-Diag [▶ 46] (TwinCAT 2.9))

3.4.4 Slave diagnostics

DP-State

Each DP slave has a status variable that indicates the current state of that DP slave. This status is maintained in real time, so that it is always adapted to the current DP slave data, and can be linked to a PLC variable (-> DpState [▶ 25] of the slave):

Fig. 16: Slave diagnostics - DP state

Diagnostic data

Each DP slave can acyclically report DP diagnostic data during data exchange operation. The slave here sets the Diag_Flag in the response to the cyclic Data_Exchange telegram, as a result of which the DP master automatically reads the DP diagnostic data from the slave. This does not affect the Data-Exchange cycle in the Beckhoff DP master, because the DP diagnostic telegram is sent at the end of the cyclic Data-Exchange cycle, and before the beginning of the next cycle. If the DP diagnostic data read from the slave has changed from its previous state, the DP master sets the "ExtDiagFlag" variable, which can be linked to a variable in the control program.

The DP slave's current diagnostic data is displayed in the System Manager on the slave's Diag [▶ 62] tab. It can also be read by the control program via ADS [▶ 28], which will cause the "ExtDiagFlag" flag to be reset once more:

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	200
IndexGroup	0x00yyF181 (yy = station address of the slave)
IndexOffset	Offset within the diagnostic data
Length	Length of the diagnostic data that is to be read
Data	Diagnostic data

The diagnostic data contains the slave statistics (32 bytes) and the DP diagnostic data sent by the slave (up to 244 bytes), and is constructed as follows:

Offset	Meaning
Slave statistics	
0	Receive Error Counter (WORD): The number of faulty telegrams occurring while communicating with this slave
2	Repeat-Counter[8] (WORD): The Repeat Counters indicate how many repeats have had to be made, and how often. Repeat Counter[0] indicates how often a telegram had to be repeated once for this slave, Repeat Counter[1] indicates how often a telegram had to be repeated twice for this slave, etc. The maximum number of retries is set with the parameter **Max Retry Limit** (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Bus Parameters [▶ 49] dialog). The value range is from 0 to 8, therefore there are 8 repeat counters (for 1 to 8 retries)
18	reserved for extensions
20	NoAnswer Counter (DWORD): The number of telegrams in communication with this slave that have not received an answer. The first time that a slave fails to answer, the telegram is repeated up to **MaxRetryLimit** times, but if it does not answer even then, further telegrams are not repeated.
24-27	Last-DPV1-Error[4] (BYTE): The last faulty DPV1 response is entered here (byte 0: DPV1 service (bit 7 is set, indicating an error), byte 1: Error_Decode, byte 2: Error_Code_1 (Error_Class/Error_Code), byte 3: Error_Code_2), see description DPV1 error codes [▶ 71]
27-31	reserved for future use
from 32	DP diagnostic data [▶ 69]

3.4.5 DP State of the Slaves

Value	Description
0	No Error - station is exchanging data
1	Station deactivated - slave has been deactivated, temporary state during StartUp
2	Station not exists - slave does not reply on the bus -> check whether slave is switched on, whether PROFIBUS plug is in, correct station address or bus cables
3	Master lock - slave is exchanging data with another master -> remove other master from bus or release slave again by other master
4	Invalid slave response - incorrect answer from slave, occurs temporarily if slave has ceased data exchange as a result of a local event
5	Parameter fault - check whether Bus Coupler / GSD file is correct, that station address is correct or that UserPrmData settings are correct
6	Not supported - DP function is not supported -> check whether GSD file is correct or whether station address is correct
7	Config fault – configuration fault -> check whether the added terminals / modules are correct
8	Station not ready -> station starting up, temporarily displayed during StartUp
9	Static diagnosis - slave signaling static diagnosis and cannot deliver valid data at present -> check operating state at the slave
10	Diagnosis overflow - slave signaling a diagnosis overflow -> check diagnostic data (using ADS-Read, see below) and operating state at the slave
11	Physical fault - physical fault interfering with slave response -> check cables
13	Severe bus fault -> check cabling
14	Telegram fault - slave responding with an invalid telegram -> must not occur
15	Station has no resources -> slave has insufficient resources for the telegram -> check that GSD file is correct
16	Service not activated -> temporary fault when slave ceases data exchange due to a local event, otherwise check whether DP functions are disabled at the slave
17	Unexpected telegram -> can occur temporarily if two PROFIBUS networks are connected together or check whether bus times for the second master are set identically.
18	Station ready -> can occur temporarily during StartUp and until the task is started
19	DPV1 StartUp -> occurs temporarily after the DP has started up if there is still data to be sent by DPV1 Write
128	FC310x in slave mode, waiting for data transfer -> slave was parameterized and configured but has not yet received a Data_Exchange telegram
129	FC310x in slave mode, waiting for configuration -> slave was parameterized, but has not yet received a Chk_Cfg telegram
130	FC310x in slave mode, waiting for parameters -> slave was not yet parameterized, waiting for Set_Prm (Lock) telegram

3.5 PROFIBUS MC

The difference between PROFIBUS MC and PROFIBUS DP is that the PROFIBUS cycle is constant, with a jitter of a few microseconds (for PROFIBUS DP, the jitter is greater than 100 µs), and at the start of the cycle a broadcast global control telegram is sent, which can be used by the MC slaves for synchronization. This enables precise synchronization of drive control loops with the NC.

However, this precise synchronization means that bus disturbances, switching off of slaves, pulling of bus plugs etc. will usually lead to a loss of synchronicity between master and slave, since the bus timing is changed.

FC310x with Simodrive 611U has Plug&Play functionality

The following steps are required for operating a Simodrive 611U on a FC310x:

1. Set FC310x to operation mode "DP/MC (equidistant)".
2. Append box "Siemens AG, Profidrive MC".
3. Adjust 611U station address ("PROFIBUS" box tab).
4. Append axis (or 2 axes for 611U with 2 axes) to NC task, select axis type "continuous axis".
5. Link axis (or axes) with 611U (select axis type "ProfiDrive MC" in the "Settings" axis tab, then link with 611U; for a 2-axis 611U, both axes have to be linked, otherwise a 611U error will occur).
6. Press the "Calculate Equi-Times" button on the "FC310x" tab at the FC310x.
7. Save project in the registry and start TwinCAT. The 611U should now change to RUN, the axis can be operated via the NC online menu.

Should this not be the case, check the following:

- DpState of the 611U in TwinCAT is 2: Check 611U station address.
- DpState of the 611U in TwinCAT is 5: Check whether the correct PROFIBUS module is inserted at the 611U.
- DpState of the 611U in TwinCAT is 7: Check whether P922 is set to the correct standard telegram (according to the ProcessData tab for 611U in the System Manager).
- DpState of the 611U in TwinCAT is 0, but 611U still does not change to RUN: Check the 611U firmware version; for firmware versions below 3.4.3, 611U synchronization errors (error 597 or 598) can only be rectified via a hardware reset of the 611Us, otherwise look up the error code in the Siemens manual.

If several 611Us are configured, the equidistant times may have to be adjusted (see below).

DP/MC Equidistant Mode

In order to operate the FC310x with PROFIBUS MC, the **Operation Mode** "DP/MC (equidistant)" must be set on the tab **"FC310x"** (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the master. Whichever task uses the equidistant functionality of the FC310x (usually the NC task) should have the highest priority, as otherwise the synchronicity can be disturbed. Additionally, the **Sync Mode** can be selected. This specifies where the synchronization signal is generated.

Disabled (PC is Sync Master)

The synchronization signal is generated by the PC, the FC310x synchronizes itself with the PC (PROFIBUS cycle jitter approx. 2-4 µs).

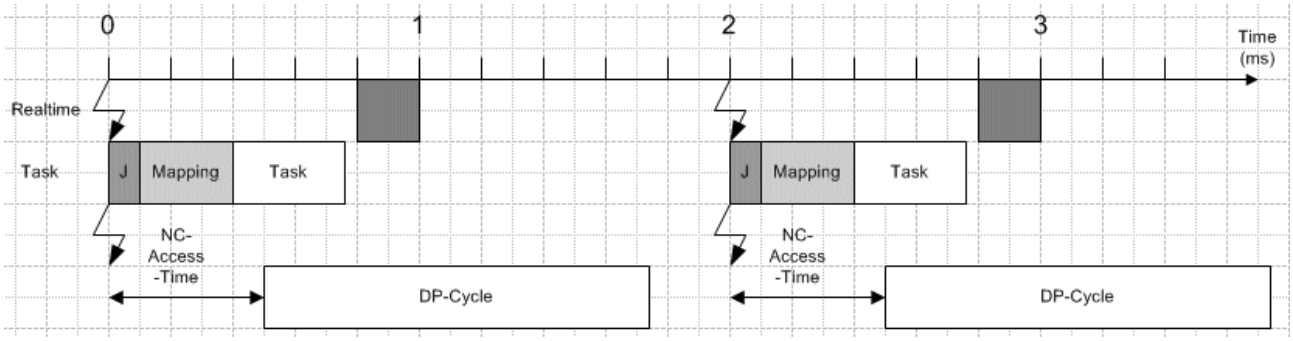

Fig. 17: PC is Sync Master

The **NC Access Time** specifies by how much the PROFIBUS cycle is shifted towards the TwinCAT cycle, the **PLL Sync Time** should be set to approx. 10% of the **NC-Access-Time** (max. 50 µs).

Sync Slave

The synchronization signal comes from another device, whose Sync mode must be set to "Sync Master". The connection between the sync master and the sync slave is made through a hardware link. This is only supported as from FC310x hardware version 4 and firmware version 3.00. No times have to be set.

Sync Master

The synchronization signal is generated by the FC310x, the PC synchronizes itself with the FC310x (jitter of the PROFIBUS cycle approx. 1 µs).

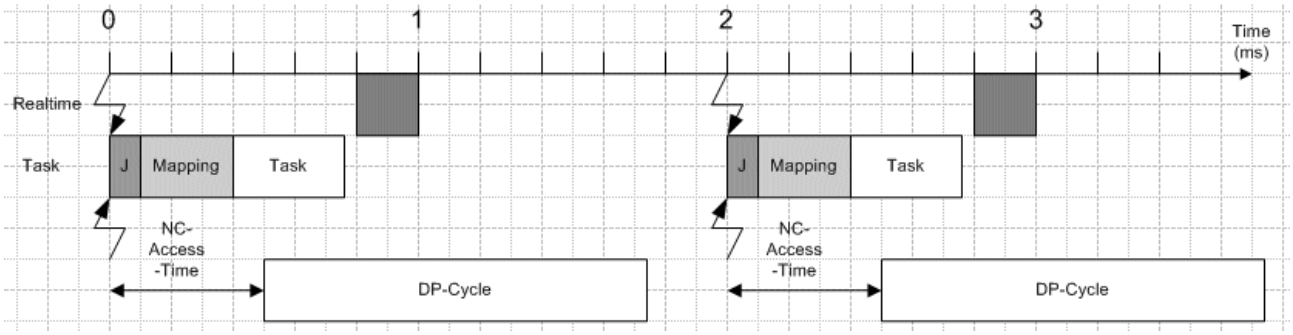

Fig. 18: FC310x is Sync Master

The **NC Access Time** specifies by how much the TwinCAT cycle is shifted towards the PROFIBUS cycle.

Setting of Equidistant Times

The **Calculate Equi-Times** button (TwinCAT 2.8: see FC310x [▶ 40] tab, TwinCAT 2.9: see MC [▶ 52] tab) can be used to automatically set all equidistant parameters. The only parameter that may possibly have to be adjusted later is the **NC Access Time**, because this depends on the maximum TwinCAT jitter and on the maximum mapping time. This in turn depends on all the devices, so that adding and linking boxes to other devices has the effect that it may nevertheless be necessary to change the **NC Access Time** for an unchanged device. If "I/O not at task start" is selected (see the Synchronization [▶ 12] chapter), then the **NC Access Time** also depends on the task runtime. To avoid having to manually adjust the **NC Access Time** every time the **Calculate Equi-Times** button is pressed, the ratio of **NC Access Time** to **Cycle Time** can be specified (set to 15% in delivery state).

Disabled (PC is Sync Master) or Sync Master

The **NC Access Time** must be greater than the maximum TwinCAT jitter plus the maximum mapping time, plus, if "I/O not at task start" is selected for whichever task linked to the FC310x has the highest priority, it's task runtime.

Diagnostics of Equidistant Times

The EquiDiag [▶ 46] tab (TwinCAT 2.8) or MC-Diag [▶ 56] tab (TwinCAT 2.9) in the System Manager can be used for diagnosing the equidistance times, or this can be done via ADS in the control program (see chapter Master diagnostics [▶ 20]).

3.6 ADS (acyclic communication)

3.6.1 ADS Interface

All acyclic data are transmitted to or from the FC310x via ADS-Read, ADS-Write or ADS-Write-Control. The FC310x has its own Net-ID and supports the following ports:

Port	Description
200	This addresses the FC310x itself, i.e. data that reside locally on the FC310x, and for which usually no additional bus access is required
0x1000 - 0x107E	This addresses a connected PROFIBUS device, with the address calculated from port-0x1000; this always involves a bus access

ADS-Read

An overview of the IndexGroups/IndexOffsets supported by the FC310x during ADS-Read is provided below.

IndexGroup for local FC310x addressing (port 200)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Description
0xF100	0x00	BYTE offset within the data	This reads the diagnostic data from the FC310x. If the ADS-Read is answered without error (error code = 0), the data will contain the diagnostic data of the FC310x described in the Master Diagnostics [▶ 20] chapter. The FC310x will reset the FC310x DiagFlag. It will be set again, if the FC310x diagnostic data change again.
0xF181	0x00-0x7E	BYTE offset within the data	This will read the diagnostic data of a configured DP slave. The station address is calculated from the IndexGroup(Hi-Word). If the ADS Read is answered without error (error code = 0), the data will contain the diagnostic data of a configured DP slave described in the Slave diagnostics [▶ 23] chapter.
0xF830	0x8000-0x807E	always 0	This enables detection of the DP slaves present at the PROFIBUS, independent of whether they were configured or not. The station address is calculated from IndexGroup(Hi-Word)-0x8000. If the ADS-Read is answered without error (error code = 0), the corresponding DP slave has answered correctly. The data contain the Ident no. of the slave (BYTE offset 0-1) and the read CfgData (from BYTE offset 2) (see chapter Upload Configuration [▶ 34]).
0xF840	0	BYTE offset within the data	This will read the firmware version and the station address of the FC310x. If the ADS-Read is answered without error (error code = 0), the data will contain the firmware version (BYTE offset 0-1) and the station address of the FC310x (BYTE offset 2).

IndexGroup for addressing of a configured PROFIBUS device (port 0x1000-0x107E)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Description
0x00-0xFF	0x00	0x00-0xFF	This will send a DPV1-Read to the appropriate, configured DPV1 slave via a Class 1 connection; the DPV1 slot number corresponds to the IndexGroup, the DPV1 index corresponds to the IndexOffset. If the ADS-Read is answered without error (error code = 0), the data will contain the read DPV1 data (see chapter DPV1 [▶ 31])
0x100-0x1FF	0x00	0x00-0xFF	This will send a DPV1-Read to the appropriate, configured DPV1 slave via a Class 2 connection; the DPV1 slot number corresponds to the IndexGroup - 0x100, the DPV1 index corresponds to the IndexOffset. If the ADS-Read is answered without error (error code = 0), the data will contain the read DPV1 data (see chapter DPV1 [▶ 31])
0x0000 - 0xFFF	0x10000000 - 0xF0000000	0x00-0xFF	This will transmit a PKW-Read to the appropriate, configured PROFIDRIVE slave; the parameter number (PNU) is contained in the Low WORD of the IndexGroup, the subindex for access to an array is in the IndexOffset, the addressed axis is in bits 28-31 of the IndexGroup (for a 1-axis unit, this must be 1), the PKW compatibility can be adjusted in bits 26,27 (unfortunately, not all PROFIDRIVE slaves are compatible, see chapter PKW Interface [▶ 35]).
0	0x01000000	0	This will send FDL-Read for Siemens AG interfacing to the appropriate configured FDL station (see chapter S5-FDL [▶ 36]).

ADS-Write

An overview of the IndexGroups/IndexOffsets supported by the FC310x during ADS-Write is provided below.

IndexGroup for local FC310x addressing (port 200)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Description
0xF100	0x00	0 -2	This will reset the equidistant diagnostic data (IndexOffset = 0), the Repeat counters (IndexOffset = 1) or the NoAnswer counters (IndexOffset = 2) of the FC310x.

IndexGroup for addressing of a configured PROFIBUS device (port 0x1000-0x107E)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Description
0x00-0xFF	0x00	0x00-0xFF	This will send a DPV1-Write to the appropriate, configured DPV1 slave via a Class 1 connection; the DPV1 slot number corresponds to the IndexGroup, the DPV1 index corresponds to the IndexOffset (see chapter DPV1 [▶ 31]).
0x100-0x1FF	0x00	0x00-0xFF	This will send a DPV1-Write to the appropriate, configured DPV1 slave via a Class 2 connection; the DPV1 slot number corresponds to the 0x100 IndexGroup, the DPV1 index corresponds to the IndexOffset (see chapter DPV1 [▶ 31]).
0x400	0x00	0x00	With this, a DPV1 Abort is sent to the appropriate configured DPV1 slave via a class 2 connection; the abort parameters are included in the data (-> chapter DPV1 [▶ 31]).
0x0000 - 0xFFF	0x10000000 - 0xF0000000	0x00-0xFF	This will transmit a PKW-Write to the appropriate, configured PROFIDRIVE slave; the parameter number (PNU) is contained in the Low WORD of the IndexGroup, the subindex for access to an array is in the IndexOffset, the addressed axis is in bits 28-31 of the IndexGroup (for a 1-axis unit, this must be 1), the PKW compatibility can be adjusted in bits 26,27 (unfortunately, not all PROFIDRIVE slaves are compatible, see chapter PKW Interface [▶ 35]).
0	0x01000000	0	This will send FDL-Write for Siemens AG interfacing to the appropriate configured FDL station (see chapter S5-FDL [▶ 36]).
0	0x02000000	0	This will send a SetSlaveAddress command to a configured DP slave, whereby the DP slave must be configured with the new station address; the old station address must be entered at BYTE offset 0 of the ADS-Write data. Furthermore, the Ident no. of the slave must be contained under BYTE offset 1 and 2, and BYTE offset 3 must contain information as to whether the slave may be modified later (0) or not (not equal 0). Altogether, 4 bytes of ADS-Write data will therefore have to be sent.

ADS-ReadWrite

An overview of the IndexGroups/IndexOffsets supported by the FC310x during ADS-ReadWrite is provided below.

IndexGroup for addressing of a configured PROFIBUS device (port 0x1000-0x107E)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Description
0x100-0x1FF	0x00	0x00-0xFF	This will send a DPV1-Data_Transport to the appropriate, configured DPV1 slave via a Class 2 connection; the DPV1 slot number corresponds to the 0x100 IndexGroup, the DPV1 index corresponds to the IndexOffset (see chapter DPV1 [▶ 31]).
0x200	0x00	0x00	With this, a DPV1 Initiate is sent to the appropriate configured DPV1 slave via a class 2 connection; the initiate parameters are included in the data (-> chapter DPV1 [▶ 31]).

ADS-WriteControl

An overview of the ADS-Write-Control commands supported by the FC310x is provided below.

ADS-WriteControl for local FC310x addressing (port 200)

AdsState	DeviceState	State of the FC310x	Description
STOP (6)	0x00	RUN (5)	This will stop the FC310x, i.e. the process data connections to all DP slaves (Data_Exchange) are removed (with SetPrm,Unlock).
RUN (5)	0x00	STOP (6)	This will restart the FC310x after a stop, i.e. the process data connections to all DP slaves (Data_Exchange) are re-established (normal DP start-up).

ADS-WriteControl for addressing a configured PROFIBUS device (port 0x1000-0x107E)

AdsState	DeviceState	State of the FC310x	Description
STOP (6)	0x00	RUN (5)	This will stop the slave, i.e. the process data connection to the relevant DP slave (Data_Exchange) is removed (with SetPrm,Unlock).
RUN (5)	0x00	STOP (6)	This will restart the slave after a stop, i.e. the process data connection to the relevant DP slave (Data_Exchange) is re-established (normal DP start-up).

ADS Error Codes

The 32 bit ADS error code always consists of a general ADS error code (Low Word, see ADS documentation) and a FC310x-specific, unique error code (High Word, -> chapter FC310x ADS error codes [▶ 37]). The appropriate text message will also be displayed in the TwinCAT System Manager Logger.

3.6.2 PROFIBUS DPV1

On a C1 connection, the master supports the Read and Write services, and on the C2 connection it supports the Read, Write, Data_Transport, Initiate and Abort services.

C1 Connection (MSAC-C1)

The C1 connection is reserved for the master that cyclically exchanges data with the slave (C1 master). In order for a slave to be able to use the C1 connection, the slave must support DPV1 (this means that the line "DPV1_Slave = 1" and the keyword "C1_Max_Data_Len" with an appropriate length must be in the GSD file). If it is also generally necessary to activate the C1 functionality by setting bit 7 in the **PrmData** byte 0 (see the slave's PROFIBUS [▶ 58] tab) for the corresponding slave (this is done automatically for those Beckhoff devices that support DPV1).

MSAC-C1-Read is shown in ADS-Read, and MSAC-C1-Write is mapped in ADS-Write:

MSAC-C1 Read

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	Slot number (DPV1 parameter)
IndexOffset	Index (DPV1 parameter)
Length	Length of the data that is to be read
Data	In response: data that has been read

MSAC-C1 Write

ADS-Write parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	Slot number (DPV1 parameter)
IndexOffset	Index (DPV1 parameter)
Length	Length of the data that is to be written
Data	In request: data that is to be written

C2 connection (MSAC-C2)

The C2 connection is as a rule intended for a second master (C2 master) that does not communicate with the slave cyclically, but it is also possible for the C1 master to make use of the C2 connection. In order for a slave to be able to use the C2 connection, the slave must support DPV1 (this means that the line "DPV1_Slave = 1" and the keyword "C2_Max_Data_Len" with an appropriate length must be in the GSD file).

The connection is automatically established by the master as soon as a Read, Write or Data_Transport access is requested; it can, however, also be explicitly established through an Initiate. When the connection is being established automatically, the master sends the Initiate parameters that have most recently been passed (see the description of Initiate), and initializes the Initiate parameters with 0 after a TwinCAT start (or restart); monitoring of the connection is an exception - this is initialized in accordance with the value set in the System Manager (**Watchdog** under **DPV1 Class 2** on the slave's PROFIBUS [▶ 58] tab).

It is also necessary for the C2 functionality to be activated for each slave that is to be addressed using C2 services by selecting the **Enable** check box under **DPV1 Class 2** (see the slave's PROFIBUS [▶ 58] tab).

If a different master performs cyclic data exchange with the slave, then the "No cyclic connection" setting must be chosen under **DP Class 2** (see the slave's PROFIBUS [▶ 58] tab). This could, for instance, be useful in order to be able to debug a BC3100/IL23xx-C310 over PROFIBUS, even though it is being operated by an external controller.

MSAC-C2-Read is mapped in ADS-Read, MSAC-C2-Write in ADS-Write, MSAC-C2-Data_Transport in ADS-ReadWrite, MSAC-C2-Initiate in ADS-ReadWrite, and MSAC-C2-Abort in ADS-Write:

MSAC-C2 Read

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	0x100 + slot number (DPV1 parameter)
IndexOffset	Index (DPV1 parameter)
Length	Length of the data that is to be read
Data	In response: data that has been read

MSAC-C2 write

ADS-Write parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	0x100 + slot number (DPV1 parameter)
IndexOffset	Index (DPV1 parameter)
Length	Length of the data that is to be written
Data	In request: data that is to be written

MSAC-C2 Data_Transport

ADS-ReadWrite parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	0x100 + slot number (DPV1 parameter)
IndexOffset	Index (DPV1 parameter)
Write-Length	Length of the data that is to be written
Read-Length	Length of the data that is to be read
Data	In request: data that is to be written; in Response: data that has been read

MSAC-C2 Initiate

The MSAC-C2-Initiate service allows the C2 connection to the slave to be established or, if it already exists, for new Initiate parameters to be passed.

ADS-ReadWrite parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	0x200 + slot number (DPV1 parameter)
IndexOffset	0
Read-Length	Length of the Initiate Response parameter (6)
Write-Length	Length of the Initiate Request parameter (10 - 42)
Data	Initiate Request parameter or Initiate Response parameter

Initiate Request parameter

0x00 - 0x01	Feature_Supported
0x02 - 0x03	Profile_Feature_Supported
0x04 - 0x05	Profile_Ident_number
0x06	sType
0x07	sLen: Length of sAddr (0 - 16)
0x08	dType
0x09	dLen: Length of dAddr (0 - 16)
0x0A - 0x19	sAddr
0x1A - 0x29	dAddr

Initiate Response parameter

0x00 - 0x01	Feature_Supported (value received from slave)
0x02 - 0x03	Profile_Feature_Supported (value received from slave)
0x04 - 0x05	Profile_Ident_number (value received from slave)

MSAC-C2 Abort

The MSAC-C2 Abort service allows the C2 connection to the slave to be removed again.

ADS-Write parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	0x400 + slot number (DPV1 parameter)
IndexOffset	0
Length	Length of the Abort parameter (3)
Data	In request: Abort parameter

Abort parameter

0x00	Reason_Code
0x01 - 0x02	Additional_Detail

3.6.3 Uploading the Configuration

The PROFIBUS can be scanned by ADS [▶ 28] Read for new devices during operation:

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	200
IndexGroup	0xzzyyF830 (yy = station address, zz = 0: for Beckhoff devices, tables 0,1 and 9 are read, zz = 0x80: Beckhoff devices provide the same information as third-party devices)
IndexOffset	0
Length	1538
Data	Configuration data of the slave

If the IndexGroup indicates that, for Beckhoff devices, tables 0, 1 and 9 are to be read, then the following data is supplied, provided the device is a Beckhoff device:

Offset	Description
0 -1	0
2 -513	Table 0. Amongst the information contained here is the precise coupler type and the firmware version
514 - 1025	Table 9 (includes the coupler number and the terminal numbers)
1026 - 1537	Table 1 (only relevant for bus controllers. The assignment of the terminals is part of the information contained here)

If the device is from another manufacturer, or if the IndexGroup indicates that Beckhoff devices are to behave in exactly the same way as devices from other manufacturers, then the following information is returned in the ADS read response:

Offset	Description
0 -1	1
2 - 7	DP diagnostic data bytes 0-5 (see Slave Diagnostics [▶ 23])
8 - 251	DP configuration data (CfgData [▶ 73])

3.6.4 PKW Interface of PROFIDRIVE Slaves

The PKW interface is integrated into the FC310x; it can then be accessed via ADS [▶ 28] from the control program. PKW Read is then mapped in ADS Read, PKW Write in ADS Write and PKW-Read No Of Array Elements in ADS Read:

PKW Read

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	Bits 0-11: parameter number (PNU)
	Bits 12-25: 0
	Bit 26: 1 = Subindex in octet 3 (standard), 0 = subindex in octet 4 (Simodrive 611U)
	Bit 27: 1 = ARRAY codes are not supported by PROFIDRIVE slave
	Bits 28-31: axis number (for single axis modules always 1)
IndexOffset	Subindex (for ARRAY access)
Length	Parameter length: 2 or 4
Data	In response: Parameter value

PKW Write

ADS-Write parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	Bits 0-11: parameter number (PNU)
	Bits 12-25: 0
	Bit 26: 1 = Subindex in octet 3 (standard), 0 = subindex in octet 4 (Simodrive 611U)
	Bit 27: 1 = ARRAY codes are not supported by PROFIDRIVE slave
	Bits 28-31: axis number (for single axis modules always 1)
IndexOffset	Subindex (for ARRAY access)
Length	Parameter length: 2 or 4
Data	In request: Parameter value

PKW ReadNoOfArrayElements

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 44] tab)
Port	0x1000 + station address of the slave
IndexGroup	Bits 0-11: parameter number (PNU)
	Bits 12-15: 0
	Bit 16: 1
	Bits 17-25: 0
	Bit 26: 1 = Subindex in octet 3 (standard), 0 = subindex in octet 4 (Simodrive 611U)
	Bit 27: 1 = ARRAY codes are not supported by PROFIDRIVE slave
	Bits 28-31: axis number (for single axis modules always 1)
IndexOffset	0
Length	Parameter length: 1
Data	In response: Number of the parameter 's array elements

3.6.5 S5-FDL Communication

FDL-AGAG communication is possible with S5 controllers or with other PROFIBUS FDL devices. The following specifications apply here to the PROFIBUS SAPs:

FC310x sends

SDA request with DSAP = station address + 1 of the FC310x and SSAP = station address + 1 of the other FDL device. Data corresponds to the data length passed with the ADS write. The SDA telegram is only sent when there is a call to ADS write:

ADS-Write parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 28] tab)
Port	0x1000 + station address of the other FDL device
IndexGroup	0x01000000
IndexOffset	0
Length	Length of the data that is to be written
Data	In request: data that is to be written

Other FDL device sends:

SDA request with DSAP = station address + 1 of the other FDL device and SSAP = station address + 1 of the FC310X. Data corresponds to the data length set at the other FDL device. The FC310x temporarily stores the received data which can be read by ADS read:

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 28] tab)
Port	0x1000 + station address of the other FDL device
IndexGroup	0x01000000
IndexOffset	0
Length	Length of the received data
Data	In response: received data. There is a receive counter at the end of the received data that is incremented with every reception

ADS-Read parameters	Meaning
Net-ID	Net-ID of the master (see the device's ADS [▶ 28] tab)
Port	0x1000 + station address of the other FDL device
IndexGroup	0x04000000
IndexOffset	Station address of the other FDL device * 2
Length	2
Data	In response: receive counter, incremented with each reception

3.6.6 ADS Error Codes of the FC310x

Error code	Meaning
0x1129	IndexOffset too large during reading of the FC310x diagnostic data
0x112B	IndexOffset too large during reading of the slave diagnostic data
0x112D	Invalid station address during reading of the slave diagnostic data
0x2023	Invalid IndexOffset during resetting of the FC310x diagnostic data
0x2024	Invalid data during resetting of the FC310x diagnostic data
0x2025	Invalid data length during resetting of the FC310x diagnostic data
0x2101	DPV1-C1-Read: cyclic connection to slave not yet established
0x2102	PKW-Read: only data lengths 2 and 4 are permitted
0x2103	PKW-Read: slave not in data exchange
0x2105	PKW-Read: slave does not support PKW
0x2106	PKW-Read: Incorrect IndexOffset
0x2107	PKW-Read: Incorrect IndexGroup
0x2109	DPV1-C1-Read: FDL fault (no response etc.)
0x210A	DPV1-C1-Read: syntax error (DPV1 syntax not correct)
0x210B	DPV1-C1-Read: DPV1 fault (4 bytes error code in the diagnostic data of the slave)
0x210C	PKW-Read: syntax error
0x210D	PKW-Read: PKW error
0x210E	PKW-Read: incorrect data type
0x210F	DPV1-C1-Write: cyclic connection to slave not yet established
0x2110	PKW-Write: only data lengths 2 and 4 are permitted
0x2111	PKW-Write: Incorrect IndexOffset
0x2112	PKW-Write: slave does not support PKW
0x2113	PKW-Write: Incorrect IndexGroup
0x2114	Read general: Incorrect IndexGroup
0x2115	DPV1-C1-Write: FDL fault (no response etc.)
0x2116	DPV1-C1-Write: syntax error (DPV1 syntax not correct)
0x2117	DPV1-C1-Write: DPV1 fault (4 bytes error code in the diagnostic data of the slave)
0x211C	Read general: Incorrect IndexGroup
0x211D	SetSlaveAdress: Incorrect IndexOffset
0x211E	FDL-AGAG-Write: Incorrect IndexOffset
0x211F	FDL-AGAG-Read: Incorrect IndexOffset
0x2120	FDL-AGAG-Write: Incorrect length
0x2121	SetSlaveAddress: Incorrect length
0x2122	FDL-AGAG-Read: Incorrect length

Error code	Meaning
0x2131	Write general: wrong IndexGroup
0x2132	Write general: wrong IndexGroup
0x2137	PKW-Read: WORD received, but read data length does not equal 2
0x2138	PKW-Read: DWORD received, but read data length does not equal 4
0x2139	PKW-Read: unknown AK received (1,2 or 7 expected)
0x213A	PKW-Read-Array: WORD received, but read data length does not equal 2
0x213B	PKW-Read-Array: DWORD received, but read data length does not equal 4
0x213C	PKW-Read-Array: unknown AK received (4,5 or 7 expected)
0x213D	PKW-Write-Array: unknown AK received (2 or 7 expected)
0x213E	PKW-Write: unknown AK received (1 or 7 expected)
0x213F	PKW-Write: unknown AK received (2 or 7 expected)
0x2140	PKW-Write-Array: unknown AK received (1 or 7 expected)
0x2142	SetSlaveAddress: wrong parameter during setting of address in slave mode
0x2144	Incorrect IndexGroup in ReadWrite
0x2147	DPV1-C2-Initiate: MSAC_C2 is not activated
0x2148	Incorrect IndexGroup in Read
0x2149	Incorrect IndexGroup in Write
0x214E	DPV1-C2-Read: MSAC_C2 is not activated
0x214F	DPV1-C2-Write: MSAC_C2 is not activated
0x2150	DPV1-C2-DataTransport: MSAC_C2 is not activated
0x2151	DPV1-C2-Read: FDL fault (no response etc.)
0x2152	DPV1-C2-Read: connection aborted
0x2153	DPV1-C2-Read: DPV1 fault (4 bytes error code in the diagnostic data of the slave)
0x2154	PKW-ReadNoOfElements: length must equal 1
0x2155	PKW-ReadNoOfElements: PKW is not activated
0x2156	PKW-ReadNoOfElements: axis number is too great
0x2157	PKW-ReadNoOfElements: slave not in data exchange
0x2158	PKW-ReadNoOfElements: unknown AK received (6 or 7 expected)
0x215A	DPV1-C2-Write: FDL fault (no response etc.)
0x215B	DPV1-C2-Write: connection aborted
0x215C	DPV1-C2-Write: DPV1 fault (4 bytes error code in the diagnostic data of the slave)
0x215D	DPV1-C2-DataTransport: FDL fault (no response etc.)
0x215E	DPV1-C2-DataTransport: connection aborted
0x215F	DPV1-C2-DataTransport: DPV1 fault (4 bytes error code in the diagnostic data of the slave)
0x2163	DPV1-C2-DataTransport: Incorrect IndexOffset
0x2600-0x26FF	AK 7 (error) during PKW processing, error code in low byte
0x2700-0x27FF	Fault during DPV1 processing, 4 bytes error code in the slave diagnostic data, byte 3 of the error code (error class, error code) is in Low byte

3.7 Master redundancy

It is possible to start the DP master in redundancy mode in order to assemble a redundant control system. In this case, the DP master only listens to the bus, but is not active on it.

To assemble a redundant control system, two masters are on the PROFIBUS (both have identical configurations): the primary master, which performs communication under normal circumstances, and the redundancy master, which only listens to the bus without transmitting. The only difference in the PROFIBUS configuration between primary and redundant master should be in the settings **Redundancy Mode** and

SetPrm Unlock before DP-Start-Up or **SetPrm-Unlock at Shutdown** (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog) and perhaps the device **watchdog** (TwinCAT 2.8: see FC310x [▶ 40] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog).

Primary-Master: the **Redundancy Mode** is not active. The settings of **SetPrm-Unlock before DP Start-Up** and of **SetPrm-Unlock at Shutdown** should be deactivated, if there is to be no interaction on the DP slaves when the primary master starts or stops (outputs remain unchanged). In addition, the device **watchdog** must be set (TwinCAT 2.8: see FC310x [▶ 40] tab of the master, TwinCAT 2.9: see Fault Settings [▶ 51] dialog), to ensure that the primary master logs off the bus in the event of a PC crash.

Redundancy Master: the **Redundancy Mode** is active. The settings of **SetPrm-Unlock before DP Start-Up** and of **SetPrm-Unlock at Shutdown** should be deactivated, if there is to be no interaction on the DP slaves when the primary master starts or stops (outputs remain unchanged).

There are also three **counters** and a **StartRedundancyMasterFlag** as interfaces to the PC:

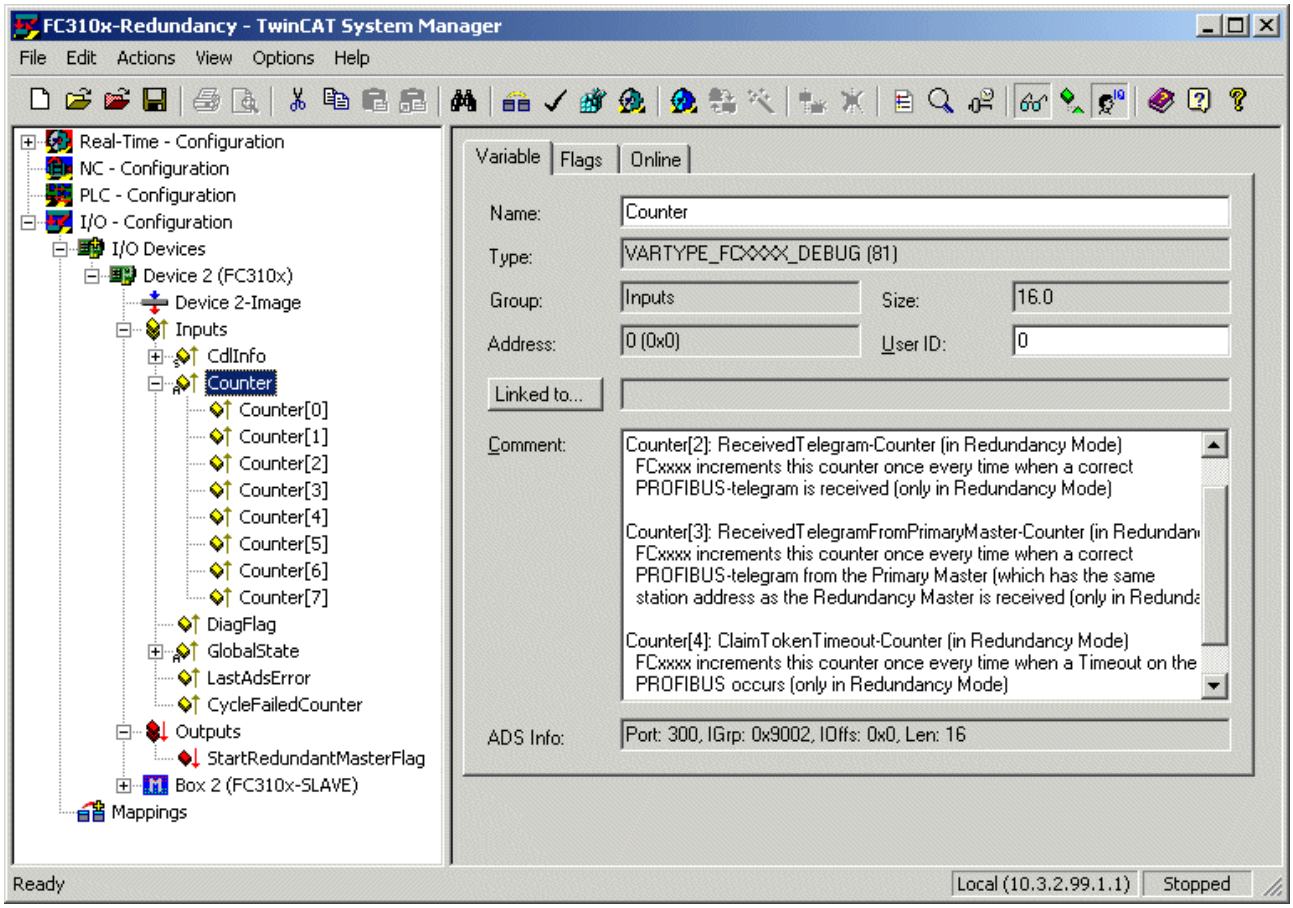

Fig. 19: Counter and StartRedundancyMasterFlag

Counter[2] (ReceivedTelegram-Counter): This counter is incremented every time a valid PROFIBUS telegram is received

Counter[3] (ReceivedTelegramFromPrimary-Counter): This counter is incremented every time a valid PROFIBUS telegram is received from the primary master (which has the same station address as the redundancy master)

Counter[4] (ClaimTokenTimeout-Counter): This counter is incremented every time the redundancy master detects a timeout on the bus after it has taken over bus activity under normal circumstances, i.e. with **Redundancy mode** deactivated. (ClaimTokenTimeout time = (6 + 2 * station address of the DP master) * slot time).

StartRedundancyMasterFlag: This can be used to start or stop the redundancy master.

The application (PLC task or other program) is therefore responsible for diagnosing a failure of the primary master (by detecting that the **ReciveTelegram-Counter** and the **ReceivedTelegramFromPrimaryMaster-Counter** no longer increment, that the **ClaimTokenTimeout-Counter** increments or the user-specific

monitoring of the two PCs is triggered). The redundancy master only becomes active at the bus when the **StartRedundancyMaster flag** is set (the startup takes approx. 10 times the min. slave interval (TwinCAT 2.8: see PROFIBUS [▶ 42] tab of the master, TwinCAT 2.9: see Bus Parameters [▶ 49] dialog). If the **StartRedundancyMaster** flag is reset, the redundancy master stops its bus activity when the next token is sent (at the end of the DP cycle, but no later than the **Estimated Cycle Time** (see **"FC310x"** tab (for TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the device)), without interrupting the connection to the slaves (irrespective of the setting **SetPrm-Unlock at Shutdown**).

When setting the DP slave's DP watchdog (see the box's PROFIBUS [▶ 58] tab) it is important to ensure that the DP watchdog time is longer than the application's monitoring time for the primary master plus the start-up time of the redundancy master, so that the redundancy master can take over the DP slave without interactions.

The redundancy master, furthermore, does not update any process data as long as it is only listening to the bus. The DpState [▶ 25] of the boxes should be evaluated when it starts; if this is 0, the process data is also up-to-date.

3.8 Device tab

3.8.1 TwinCAT 2.8

3.8.1.1 FC310x tab

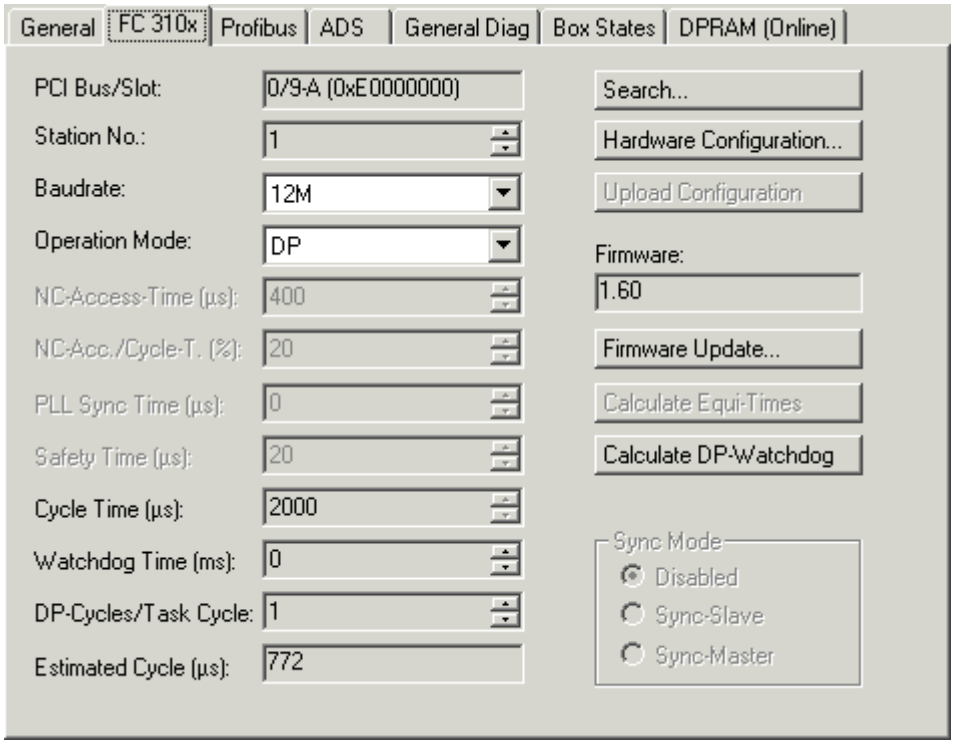

Fig. 20: TwinCAT 2.8 - FC310 tab

PCI Slot/Irq: Indicates in which logical PCI slot the card was found.

Search...: Searches for all connected FC310x channels. Select those required. In the case of an FC3102 both channels A and B appear. These behave in logical terms like two FC3101 cards.

Hardware Configuration...: The hardware version number of the FC310x can be displayed here

Upload Configuration...: The PROFIBUS is scanned with this command, and all the devices found are added to the FC310x device. (A box may not be added in TwinCAT 2.8; as from TwinCAT 2.9 scanning can also take place even when boxes are inserted. The FC310x then accepts the new configuration, but does show changes). In the case of Beckhoff boxes, the configuration is read precisely. In the case of external devices, the corresponding GSD file will be searched.

Verify Configuration...: This causes the PROFIBUS to be scanned and compared with the currently inserted boxes. Changes are displayed (from TwinCAT 2.9).

Firmware: Displays the current FC310x firmware version.

Firmware Update...: This command can be used to update the FC310x card firmware.

Stations No.: Each Profibus device requires a unique station number - including the master.

Baud rate: Set the Profibus baud rate.

Operation Mode: In all three operation modes, the highest-priority task linked to the appropriate device will take control of the PROFIBUS cycle and is therefore synchronized with the DP cycle (see the Synchronization [▶ 12] chapter). If this task is stopped or reaches a breakpoint, the FC310x switches to CLEAR mode (slave outputs will assume 0 or safe values) (see the Error Reactions [▶ 16] chapter). All other tasks are served asynchronously via corresponding buffers. If one of these tasks is stopped or reaches a breakpoint, the System Manager will generally display a message saying that the watchdog of the appropriate asynchronous mapping has been activated, and the appropriate outputs are set to 0. For all operation modes, one poll rate per slave can be set (in the Features [▶ 59] tab for the Box). The sequence of the slaves in the PROFIBUS cycle corresponds to the sequence in which they are located in the FC310x device tree. The operation mode "DP" is designed for standard DP operation, the operation modes "DP/MC (equidistant)" and "Equidistant (no GC)" are described in chapter PROFIBUS MC [▶ 25], as well as the parameters **NC Access Time, Relation NC Access Time/Cycle Time, PLL Sync Time** and **Safety Time**, the **Sync Mode** settings and the **Calculate Equi-Times** button, which are only relevant for PROFIBUS MC.

Cycle Time: Displays the cycle time of the corresponding highest priority task.

Estimated Cycle: Displays the expected PROFIBUS cycle time.

Watchdog Time: Here a watchdog can be activated, which, in case of a PC crash, will cause the FC310x to enter the STOP state and terminate the data exchange with all configured slaves (see Error Reactions [▶ 16]). The time is important in redundancy mode [▶ 38] of the primary master.

Calculate DP-Slave Watchdog Time: This will set the DP watchdog time for all the DP slaves to a reasonable value, in accordance with the formula **Estimated cycle time * 10**

3.8.1.2 Profibus tab

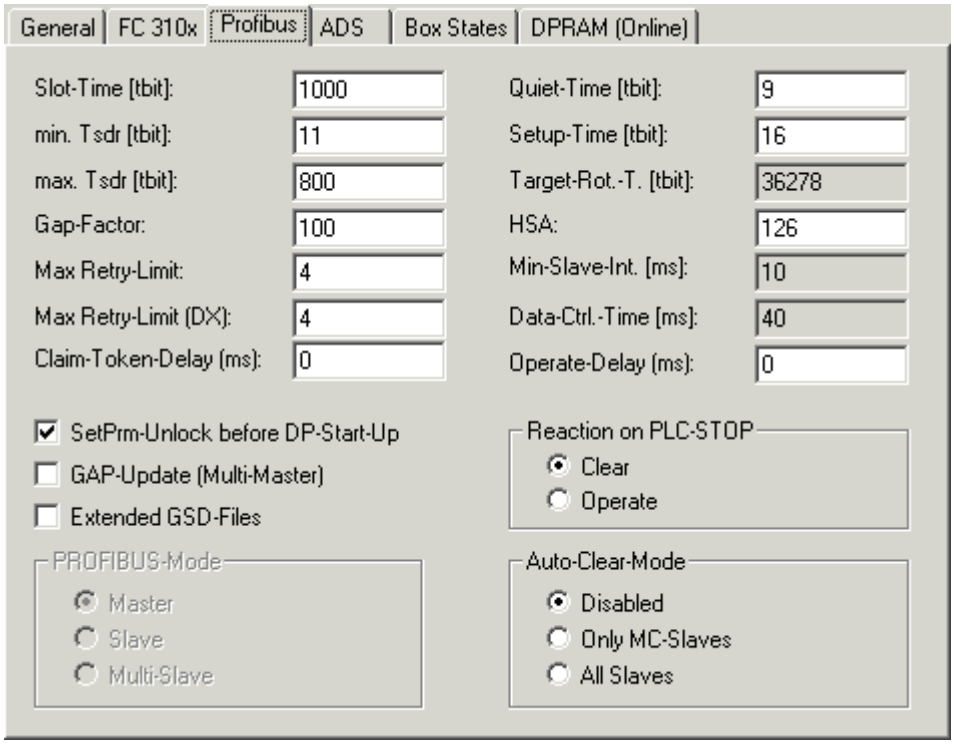

Fig. 21: TwinCAT 2.8 - Profibus tab

Slot-Time: The Slot-Time indicates how long the DP master will wait for a response from the DP slave before it sends either a repetition or the next telegram.

min. Tsdr: The min. Tsdr indicates the minimum length of time for which the DP slave will wait with a response. This time is set for all the DP slaves during the DP start-up (the value range is 11-255 bit periods). The min. Tsdr must be smaller than the max. Tsdr.

max. Tsdr: The max. Tsdr indicates the maximum length of time for which the DP slave may wait with a response. This time is set according to the DP slave's GSD file entries. The max. Tsdr must be smaller than the slot time.

Max-Retry-Limit: The Max-Retry-Limit specifies how often a telegram should be repeated, if the device addressed does not answer. The minimum value should be 1, so that, in case of an error, there will be at least one repeat for acyclic telegrams (see the Error Reactions [▶ 16] chapter).

Max-Retry-Limit (DX): Since the Data_Exchange telegram is repeated cyclically, a value of 0 could be used for the repetition of the Data_Exchange telegram here, in order to keep the cycle relatively constant in equidistant mode, even if there is no response from a device. However, in this case it would make sense to set the Features [▶ 59] tab for the box such that lack of response of the slave would not lead to DATA EXCH being exited. The fact that a device has not responded is apparent from DpState [▶ 25], which would not be equal 0 for one cycle (see the Error Reactions [▶ 16] chapter).

GAP Update: The GAP update asks all stations up to HSA at intervals to confirm their presence. It can be en/disabled. The GAP update is relevant only for multi-master operation. In single master operation it increases PROFIBUS cycle jitter and is therefore switched off by default.

GAP-Factor: The GAP factor determines how often the GAP update will be carried out (assuming it is activated). The time between two GAP updates cycles is **Gap-Factor * Target-Rot.-T**.

HSA: The HSA specifies the highest active address up to which the GAP update is carried out (assuming it is active).

Min. Slave-Int.: The MinSlaveInterval indicates the minimum cycle time with which the DP StartUp telegrams are sent the DP slaves (it is determined from the settings found in the GSD file).

PROFIBUS Mode: This is where the selection is made between master [▶ 10] functionality (the default setting) and slave [▶ 64] functionality.

Auto-Clear-Mode: It is possible to specify here whether the master enters (or stays in) the "Clear" state as long as either at least one MC slave (the "Only MC-Slaves" setting) or any slave (the "All Slaves" setting) does not respond correctly (has a DpState [▶ 25] other than 0) (see the Error Reactions [▶ 16] chapter).

Clear-Delay: The DP master changes automatically into the clear mode (the outputs of the slaves are set either to 0 or to the fail-safe values) when it ceases to receive an interrupt from the associated task (e.g. a PLC breakpoint has been reached, or the system has crashed). It is possible to specify here how many missing tasks cycles can be tolerated before the master switches into the clear mode. This setting is independent of the setting in the **Auto-Clear-Mode**.

Operate-Delay: The DP master changes automatically, observing the Auto-Clear-Mode, into the operate state when the task is started. The transition from Clear to Operate can be delayed with the Operate delay time. In the Clear state, all the outputs are set to 0 (if the DP slave does not support Fail_Safe values) or to the Fail_Safe value (if the DP slave supports Fail_Safe), whereas in the Operate state the outputs have the values specified by the task.

Reaction on PLC-STOP: It is possible to specify here whether the DP master should set the outputs to 0 when reaching a PLC stop or breakpoint, or should leave them unchanged (see the Fault Reactions [▶ 16] chapter).

Redundancy-Mode: Redundancy mode can be set here for the DP master. In that case all that it does is to listen to the bus (see the Master Redundancy [▶ 38] chapter).

SetPrm-Unlock before DP-Start-Up: Normally, during DP start-up, the DP master removes the cyclic connections, so that the DP slave can always recognize that the DP master has restarted. In redundancy mode, however, it may be specifically desirable for the DP slave to remain unaware of this, because the switch-over from the primary master to the redundant master should not have any interactions for the DP slave (see the Master Redundancy [▶ 38] chapter).

SetPrm-Unlock at DP-Shutdown: Normally, during DP shut-down, the DP master removes the cyclic connections, so that the DP slave can always recognize that the DP master has stopped. In redundancy mode, however, it may be specifically desirable for the DP slave to remain unaware of this, because the switch-over from the primary master to the redundant master should not have any interactions for the DP slave (see the Master Redundancy [▶ 38] chapter).

3.8.1.3 ADS tab

The FC310x is an ADS device with its own Net-ID, which can be changed here. All ADS services (diagnostics, acyclic communication) going to the FC310x must address this Net-ID.

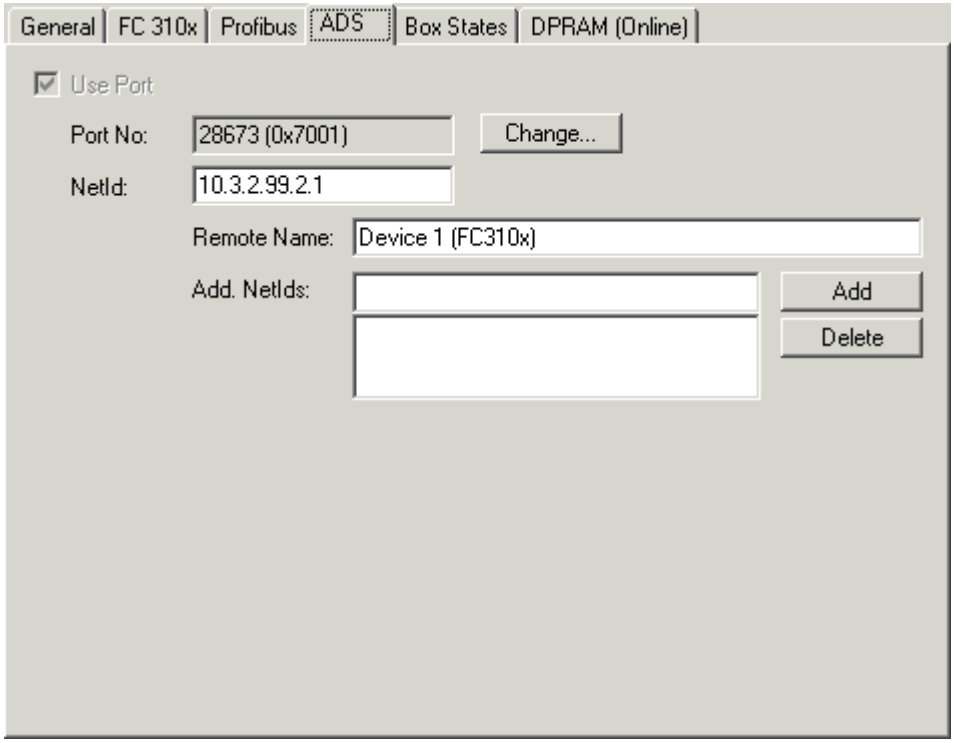

Fig. 22: TwinCAT 2.8 - ADS tab

3.8.1.4 General Diag tab

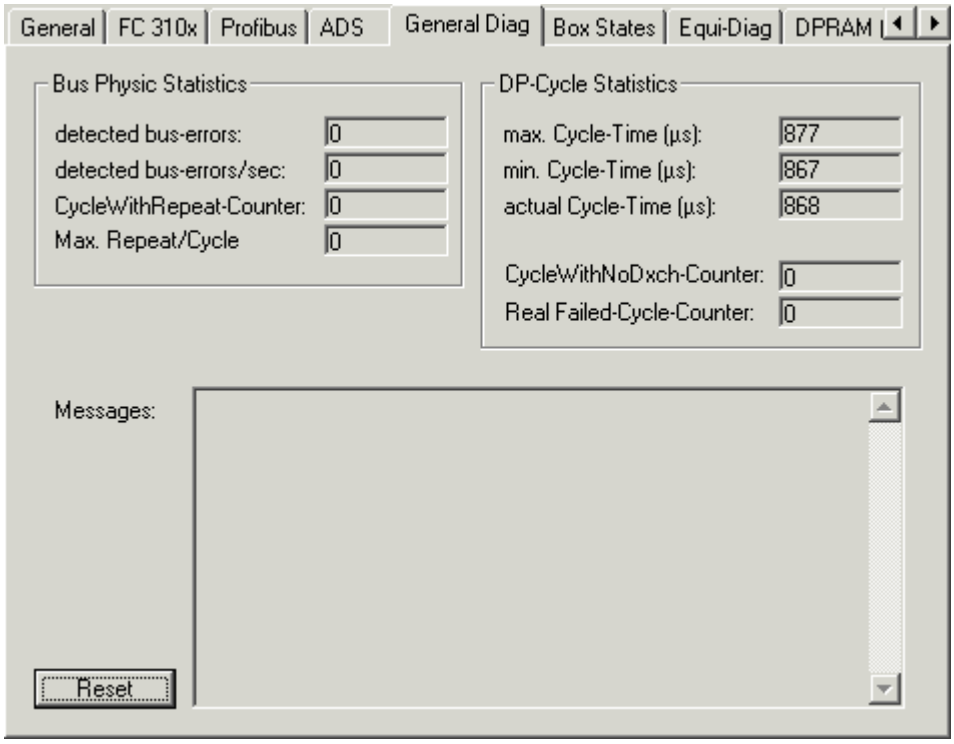

Fig. 23: TwinCAT 2.8 - General Diag tab

Here, bus cabling problems and DP cycle times are displayed:

detected bus-errors: Here, the number of detected bus errors is displayed. If this counter is not equal 0, the cabling should be checked (provided that no PROFIBUS connectors were pulled or inserted - usually there are short bus disturbances during pulling or inserting of PROFIBUS connectors).

CycleWithRepeatCounter: Here, the number of PROFIBUS cycles is displayed, in which a telegram was repeated at least once. Repetitions are also an indication that the physical bus characteristics are not 100% OK.

max. Repeat/Cycle: Here, the maximum number of repetitions within a cycle is displayed.

min./max./actual Cycle-Time: Here, the minimum, maximum and current DP cycle time is displayed. Only those cycles are considered, during which all slaves participated in the data exchange and no repetitions occurred.

CycleWithNoDxch-Counter: Increments if not all slaves participate in the data exchange (i.e. have a DpState not equal 0).

Real Failed-Cycle-Counter: Increments if the DP cycle was not completed before the next task cycle began and all the slaves are involved in the data exchange (i.e. have a DpState of 0).

3.8.1.5 Box States tab

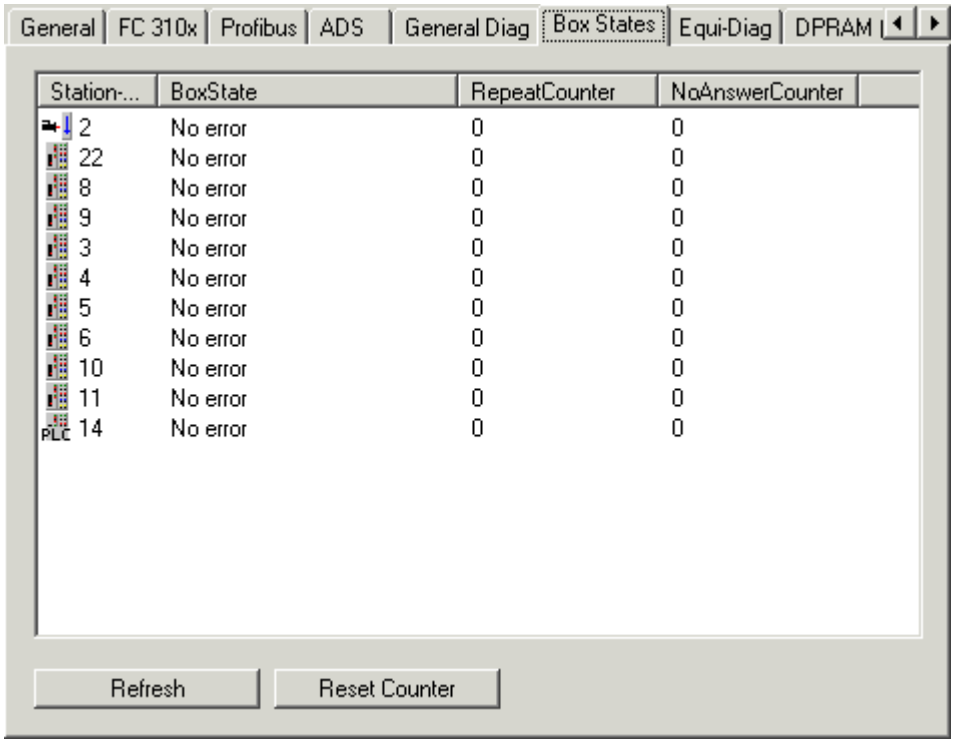

Fig. 24: TwinCAT 2.8 - Box States tab

Here, an overview of all current box states, the Repeat counter (increments for each telegram repeat to the slave) and the NoAnswer counter (increments every time the slave fails to answer) is displayed.

3.8.1.6 EquiDiag tab

The "EquiDiag" tab is available for displaying various equidistant monitoring parameters online:

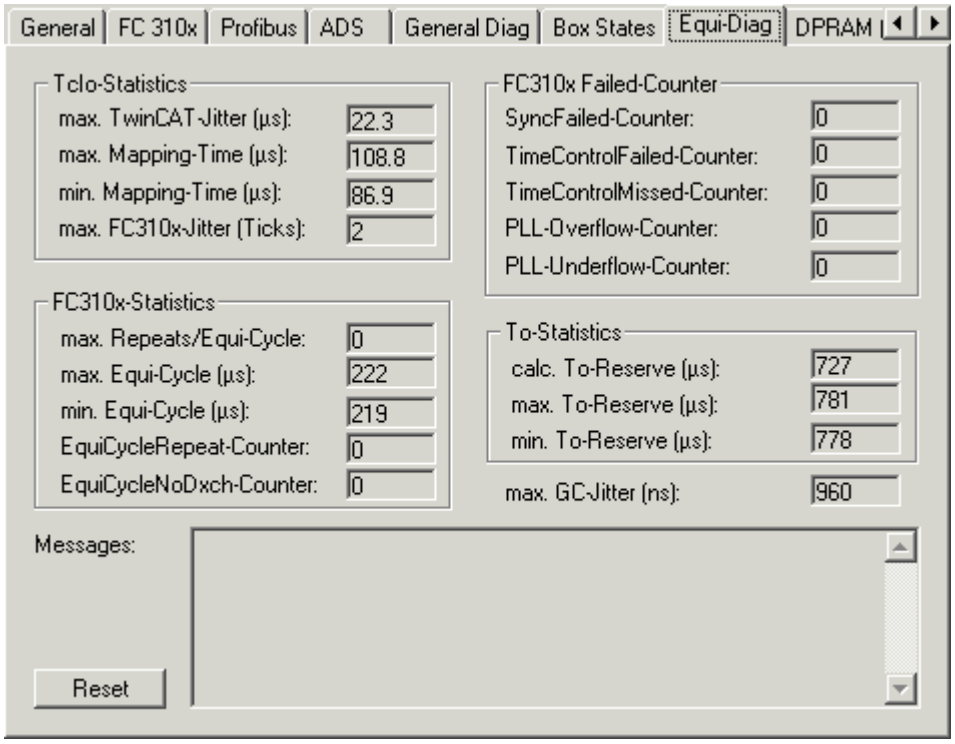

Fig. 25: TwinCAT 2.8 - EquiDiag tab

Tclo-Statistics

max. TwinCAT-Jitter: Here, the maximum TwinCAT jitter is displayed; the measurement is reset as soon as the tab is no longer active.

min./max. Mapping Time or NC Access Time: Here, the minimum or maximum NC Access Time is displayed. Apart from the Mapping Time, this also contains the task runtime (provided that "IO at Task Start" for the highest-priority task linked to the FC310x (usually the NC task) is set).

The NC Access Time should be greater than the sum of the two measured times (max. TwinCAT Jitter plus max. Mapping Time or (NC) Task Time), with a safety margin of approx. 10%.

max. FCxxxx Jitter: This is only relevant for Sync Mode = "disabled". If the value is greater than 5, real-time jitter will become excessive, and a more powerful PC should be used. If the associated (NC) task does not have the highest priority, the maximum FCxxxx jitter can also be larger. This should be avoided.

FC310x-Statistics

Since the PROFIBUS MC slaves should always be added to the FC310x device first, they are always addressed before the DP slaves in the DP cycle. This part of the DP cycle will be called Equi-Cycle below. If the Equi-Cycle becomes greater than the To-time of the MC slaves, the last MC slaves at the FC310x device will usually get a synchronization error (error 597 or 598 for Simodrive 611U).

max. Repeats/Equi-Cycle: This will show the maximum number of repeated Data_Exchange telegrams during an Equi-Cycle. This will extend the Equi-Cycle, usually no repetitions should occur (unless a bus plug was pulled or an MC slave was switched off).

min./max. Equi-Cycle: Here, the minimum or maximum Equi-Cycle Time is displayed.

Equi-Cycle Repeat Counter: Here, the number of telegram repetitions within the Equi-Cycle is displayed.

Equi-Cycle NoDxch Counter: Here, the number of occurrences is displayed, for which not all MC slaves were in data exchange during an Equi-Cycle.

FC310x-Failed Counter

Sync-Failed Counter: This counter will increment if TwinCAT task and DP cycle are not synchronized with each other. This may happen during start-up of the TwinCAT system, after which this counter should no longer increment. If the associated (NC) task does not have the highest priority, this counter can also increment. This should be avoided.

Time-Control-Failed Counter: This counter will increment if the PROFIBUS was not free at the time of the DP cycle start. Possible causes are bus faults, non-existent device, a second master or a safety time that is too small.

PLL-Overflow-/-Underflow-Counter: This counter is only relevant during Sync mode "disabled" and will increment in case of excessive jitter of the TwinCAT task which the DP cycle uses for synchronization (this may happen, for example, if the DP cycle is not synchronized with the highest-priority task). If the associated (NC) task does not have the highest priority, this counter can also increment. This should be avoided.

To-Statistics

For each MC slave the To-time specifies when, relative to the DP cycle start, the slave should accept the outputs received from the master. The MC slaves can be synchronized with each other, if the same To value is set for all MC slaves. However, this value must be equal or greater than the Equi-Cycle Time plus a safety margin of approx. 200 µs. The To-time for all MC slaves is calculated with the button "Calc. Equi-Times" (see above).

calc. To-Reserve: This contains the calculated To-Reserve (To-Time - Equi-Cycle-Time)

min./max. To-Reserve: Here, the min. or max. To-Reserve is measured.

Max. GC jitter (from TwinCAT 2.8)

Here, the maximum jitter of the DP cycle is measured (GC for global control telegram, which is always sent at the start of a cycle). During start-up, the jitter may be somewhat greater, in the steady state it should not exceed 1 µs (for Sync mode "Sync Master") or 2 µs (for Sync mode "Disabled").

3.8.2 TwinCAT 2.9

3.8.2.1 FC310x tab

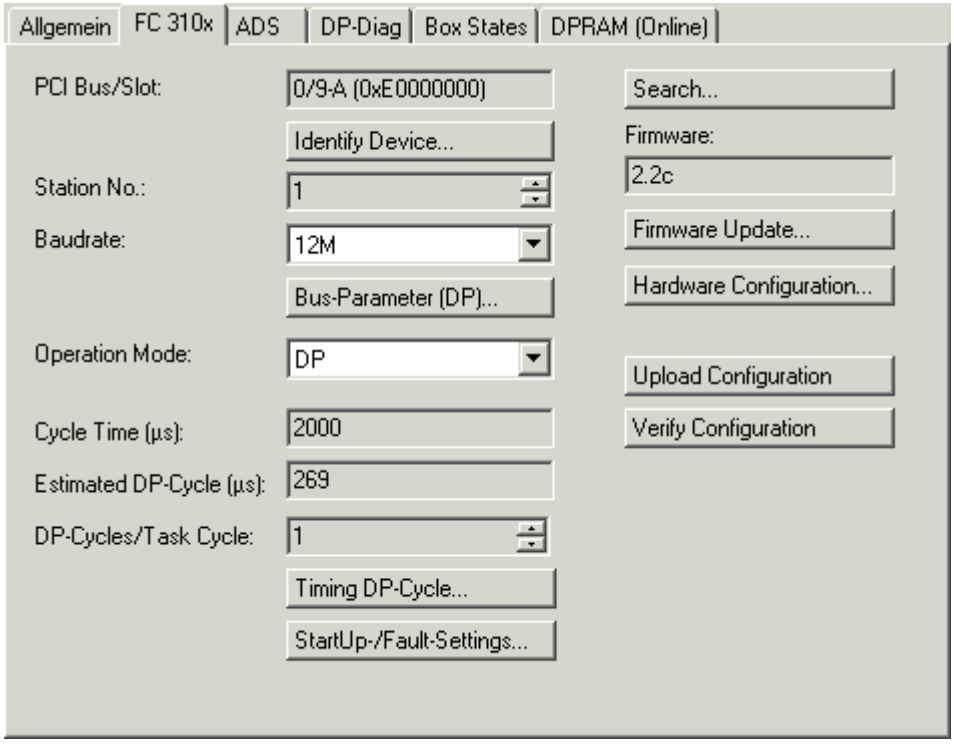

Fig. 26: TwinCAT 2.9 - FC310x tab

PCI Slot/Irq: Indicates in which logical PCI slot the card was found.

Search...: Searches for all connected FC310x channels. Select those required. In the case of an FC3102 both channels A and B appear. These behave in logical terms like two FC3101 cards.

Identify Device...: Here an LED code can be output on the corresponding FC310x channel.

Stations No.: Each Profibus device requires a unique station number - including the master.

Baud rate: Set the Profibus baud rate.

Bus parameters (DP)...: The Bus parameters dialog [▶ 49] is selected here.

Operation Mode: In all three operation modes, the highest-priority task linked to the appropriate device will take control of the PROFIBUS cycle and is therefore synchronized with the DP cycle (see the Synchronization [▶ 12] chapter). If this task is stopped or reaches a breakpoint, the FC310x switches to CLEAR mode (slave outputs will assume 0 or safe values) (see the Error Reactions [▶ 16] chapter). All other tasks are served asynchronously via corresponding buffers. If one of these tasks is stopped or reaches a breakpoint, the System Manager will generally display a message saying that the watchdog of the appropriate asynchronous mapping has been activated, and the appropriate outputs are set to 0. For all operation modes, one poll rate per slave can be set (in the Features [▶ 59] tab for the Box). The sequence of the slaves in the PROFIBUS cycle corresponds to the sequence in which they are located in the FC310x device tree. "DP" mode is used for standard DP operation; the operation modes "DP/MC (equidistant)" and "Equidistant (no GC)" are described in the PROFIBUS-MC [▶ 25] chapter.

Cycle Time: Displays the cycle time of the corresponding highest priority task.

Estimated Cycle: Displays the expected PROFIBUS cycle time.

DP-Cycles/Task-Cycle: This can be used to set several DP cycles in a task cycle, in order to obtain as many new inputs as possible (see Slave prioritization/several DP cycles [▶ 14] chapter).

Timing DP-Cycle...: Shows the timing of the DP cycle; this is particularly advantageous when slave prioritization [▶ 14] is used.

StartUp-/Fault-Settings...: This is used to select the Fault settings dialog [▶ 51].

Firmware: Displays the current FC310x firmware version.

Firmware Update...: This command can be used to update the FC310x card firmware.

Hardware Configuration...: The hardware version number of the FC310x can be displayed here

Upload Configuration...: The PROFIBUS is scanned with this command, and all the devices found are added to the FC310x device. (A box may not be added in TwinCAT 2.8; as from TwinCAT 2.9 scanning can also take place even when boxes are inserted. The FC310x then accepts the new configuration, but does show changes). In the case of Beckhoff boxes, the configuration is read precisely. In the case of external devices, the corresponding GSD file will be searched.

Verify Configuration...: This causes the PROFIBUS to be scanned and compared with the currently inserted boxes. Changes are displayed (from TwinCAT 2.9).

3.8.2.2 Bus parameters dialog

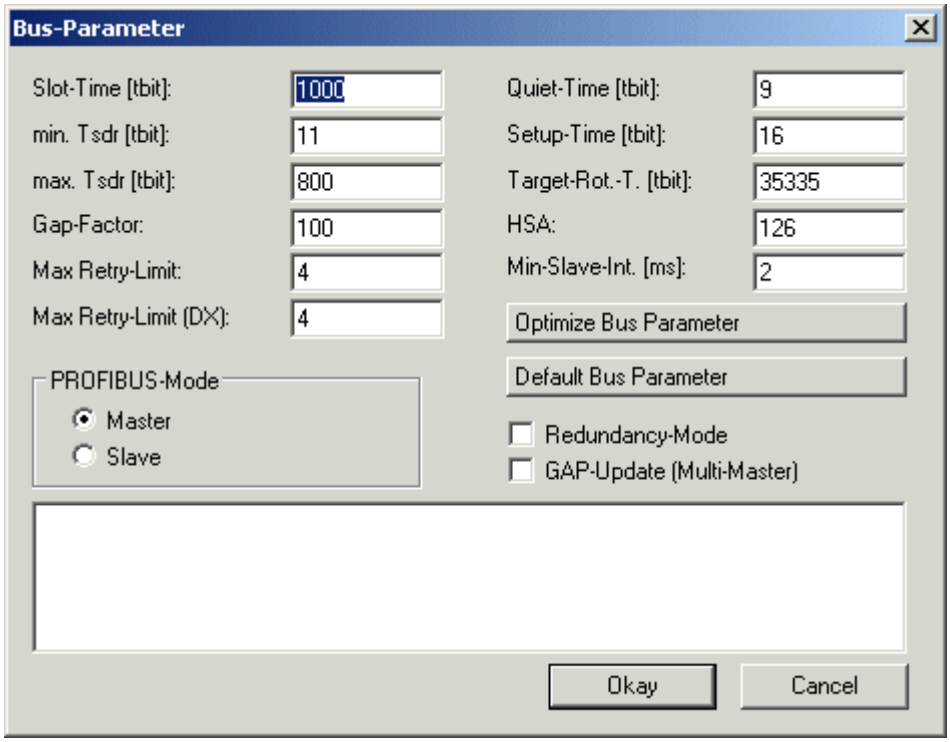

Fig. 27: TwinCAT 2.9 - Bus Parameters dialog

Slot-Time: The Slot-Time indicates how long the DP master will wait for a response from the DP slave before it sends either a repetition or the next telegram.

min. Tsdr: The min. Tsdr indicates the minimum length of time for which the DP slave will wait with a response. This time is set for all the DP slaves during the DP start-up (the value range is 11-255 bit periods). The min. Tsdr must be smaller than the max. Tsdr.

max. Tsdr: The max. Tsdr indicates the maximum length of time for which the DP slave may wait with a response. This time is set according to the DP slave's GSD file entries. The max. Tsdr must be smaller than the slot time.

GAP-Factor: The GAP factor determines how often the GAP update will be carried out (assuming it is activated). The time between two GAP updates cycles is **Gap-Factor * Target-Rot.-T**.

Max-Retry-Limit: The Max-Retry-Limit specifies how often a telegram should be repeated, if the device addressed does not answer. The minimum value should be 1, so that, in case of an error, there will be at least one repeat for acyclic telegrams (see the Error Reactions [▶ 16] chapter).

Max-Retry-Limit (DX): Since the Data_Exchange telegram is repeated cyclically, a value of 0 could be used for the repetition of the Data_Exchange telegram here, in order to keep the cycle relatively constant in equidistant mode, even if there is no response from a device. However, in this case it would make sense to set the Features [▶ 59] tab for the box such that lack of response of the slave would not lead to DATA EXCH being exited. The fact that a device has not responded is apparent from DpState [▶ 25], which would not be equal 0 for one cycle (see the Error Reactions [▶ 16] chapter).

HSA: The HSA specifies the highest active address up to which the GAP update is carried out (assuming it is active).

Min. Slave-Int.: The MinSlaveInterval indicates the minimum cycle time with which the DP StartUp telegrams are sent the DP slaves (it is determined from the settings found in the GSD file).

PROFIBUS Mode: This is where the selection is made between master [▶ 10] functionality (the default setting) and slave [▶ 64] functionality.

Redundancy-Mode: Redundancy mode can be set here for the DP master. In that case all that it does is to listen to the bus (see the Master Redundancy [▶ 38] chapter).

GAP Update: The GAP update asks all stations up to HSA at intervals to confirm their presence. It can be en/disabled. The GAP update is relevant only for multi-master operation. In single master operation it increases PROFIBUS cycle jitter and is therefore switched off by default.

Optimize bus parameters: This is used to set the optimized bus parameters.

Default bus parameters: This is used to set the default bus parameters.

3.8.2.3 Fault Settings dialog

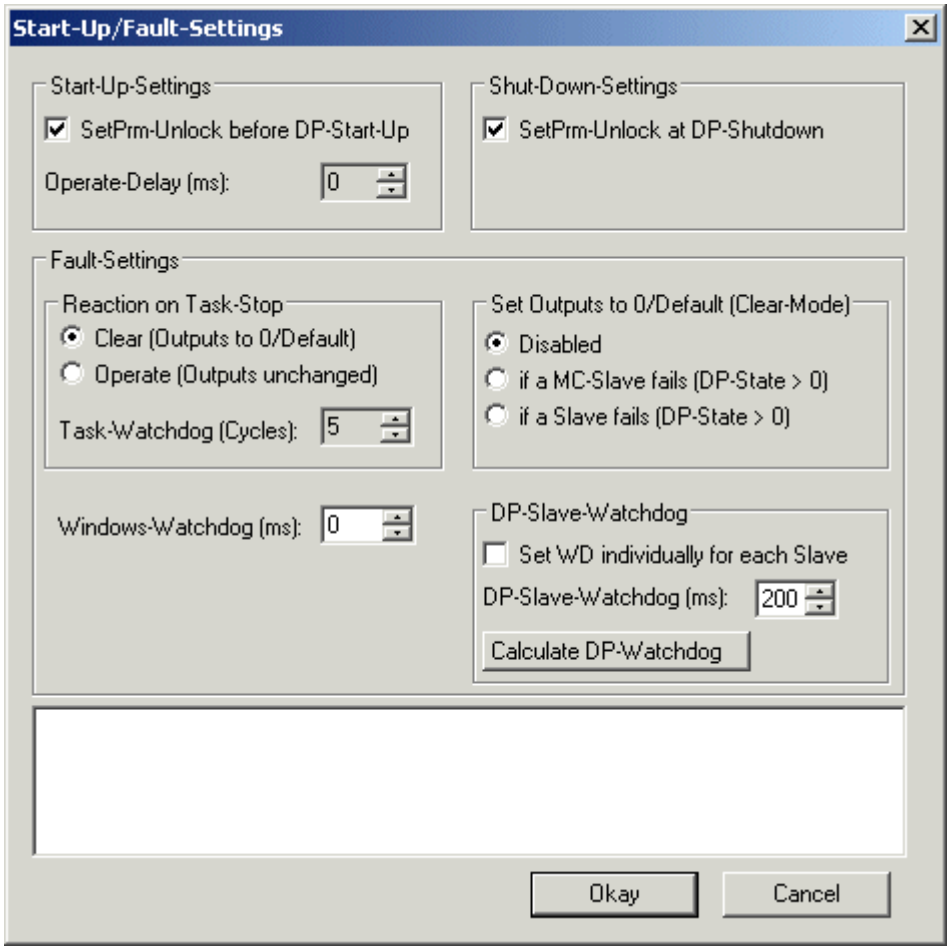

Fig. 28: TwinCAT 2.9 - Fault Settings dialog

SetPrm-Unlock before DP-Start-Up: Normally, during DP start-up, the DP master removes the cyclic connections, so that the DP slave can always recognize that the DP master has restarted. In redundancy mode, however, it may be specifically desirable for the DP slave to remain unaware of this, because the switch-over from the primary master to the redundant master should not have any interactions for the DP slave (see the Master Redundancy [▶ 38] chapter).

SetPrm-Unlock at DP-Shutdown: Normally, during DP shut-down, the DP master removes the cyclic connections, so that the DP slave can always recognize that the DP master has stopped. In redundancy mode, however, it may be specifically desirable for the DP slave to remain unaware of this, because the switch-over from the primary master to the redundant master should not have any interactions for the DP slave (see the Master Redundancy [▶ 38] chapter).

Operate-Delay: The DP master changes automatically, observing the Auto-Clear-Mode, into the operate state when the task is started. The transition from Clear to Operate can be delayed with the Operate delay time. In the Clear state, all the outputs are set to 0 (if the DP slave does not support Fail_Safe values) or to the Fail_Safe value (if the DP slave supports Fail_Safe), whereas in the Operate state the outputs have the values specified by the task.

Reaction on task STOP: It is possible to specify here whether the DP master should set the outputs to 0 when reaching a PLC stop or breakpoint, or should leave them unchanged (see the Fault Reactions [▶ 16] chapter).

Task-Watchdog: The DP master changes automatically into the clear mode (the outputs of the slaves are set either to 0 or to the fail-safe values) when it ceases to receive an interrupt from the associated task (e.g. a PLC breakpoint has been reached, or the system has crashed). It is possible to specify here how many missing tasks cycles can be tolerated before the master switches into the clear mode. This setting is independent of the setting in the **Clear mode**.

Clear-Mode: It is possible to specify here whether the master enters (or stays in) the "Clear" state as long as either at least one MC slave (the "Only MC-Slaves" setting) or any slave (the "All Slaves" setting) does not respond correctly (has a DpState [▶ 25] other than 0) (see the Error Reactions [▶ 16] chapter).

Windows watchdog: Here a watchdog can be activated, which, in case of a PC crash, will cause the FC310x to enter the STOP state and terminate the data exchange with all configured slaves (see Error Reactions [▶ 16]). The time is important in redundancy mode [▶ 38] of the primary master.

Set WD individually for each slave: Here you can select whether the WD should be set individually for each slave (on the Profibus [▶ 58] tab of the box)

DP-Watchdog-Time: If the checkbox "Set WD individually for each slave" is not ticked, the DP watchdog can be set here to a uniform value for all slaves.

Calculate DP-Slave Watchdog Time: This is used to set the DP watchdog time for all DP slaves to a sensible value.

3.8.2.4 MC tab

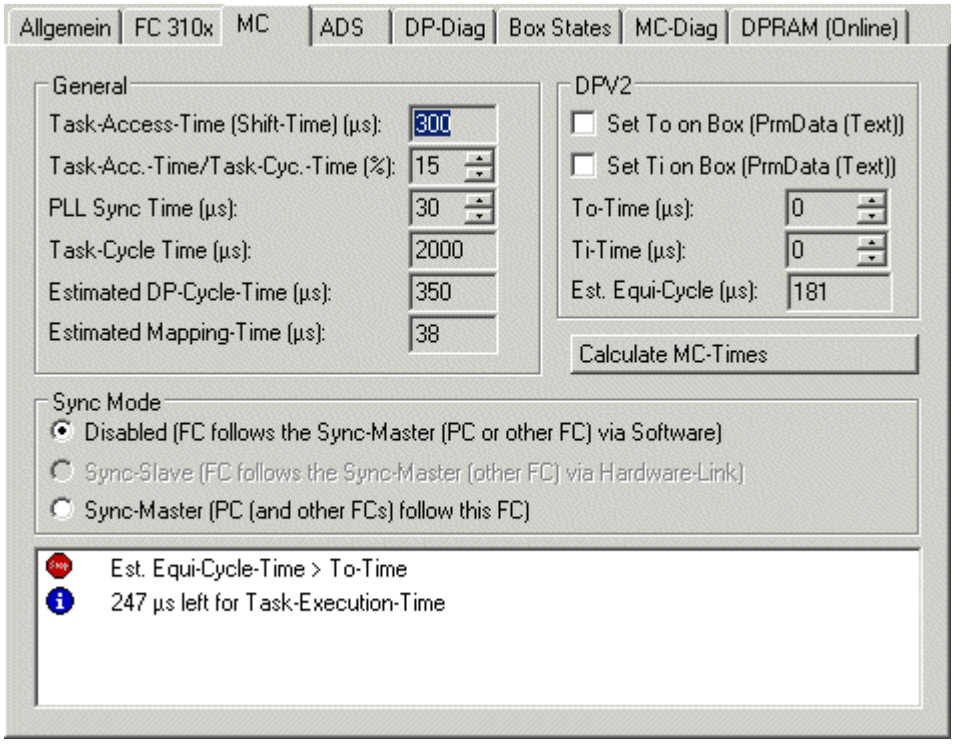

Fig. 29: TwinCAT 2.9 - MC tab

Task-Access-Time (Shift-Time): This time must be greater than the maximum TwinCAT jitter plus the maximum mapping time (see chapter PROFIBUS MC [▶ 25])

Task access time/task cycle time: This ratio makes sense if you want to avoid manually adjusting the task access time when using the **Calculate MC-Times** button.

PLL-Sync-Time: Only relevant for **Sync-Mode Disabled**, sets the PLL window on the FC310x

Task Cycle Time: Displays the cycle time of the corresponding highest priority task

Estimated DP-Cycle Time: Displays the expected PROFIBUS cycle time.

Estimated Mapping Time: Indicates the expected mapping time.

Set To on Box: The To-time can be set individually for each slave on the Prm data (text) [▶ 62] tab of the box

Set Ti on Box: The Ti-time can be set individually for each slave on the Prm data (text) [▶ 62] tab of the box

To-Time: If the checkbox **Set To on Box** is not ticked, the To-time can be set to the same value for all slaves.

Ti-Time: If the checkbox **Set Ti on Box** is not ticked, the Ti-time can be set to the same value for all slaves

Estimated Equi-Cycle Time: Shows the DPV2 part of the expected PROFIBUS cycle time.

Calculate MC-Times: This button can be used to calculate all DPV2 times automatically

Sync-Mode: The Sync-Mode decides whether the FC follows the PC (disabled) or whether the PC follows the FC (master), in order to synchronize the TwinCAT cycle with the PROFIBUS cycle

3.8.2.5 ADS tab

The FC310x is an ADS device with its own Net-ID, which can be changed here. All ADS services (diagnostics, acyclic communication) going to the FC310x must address this Net-ID.

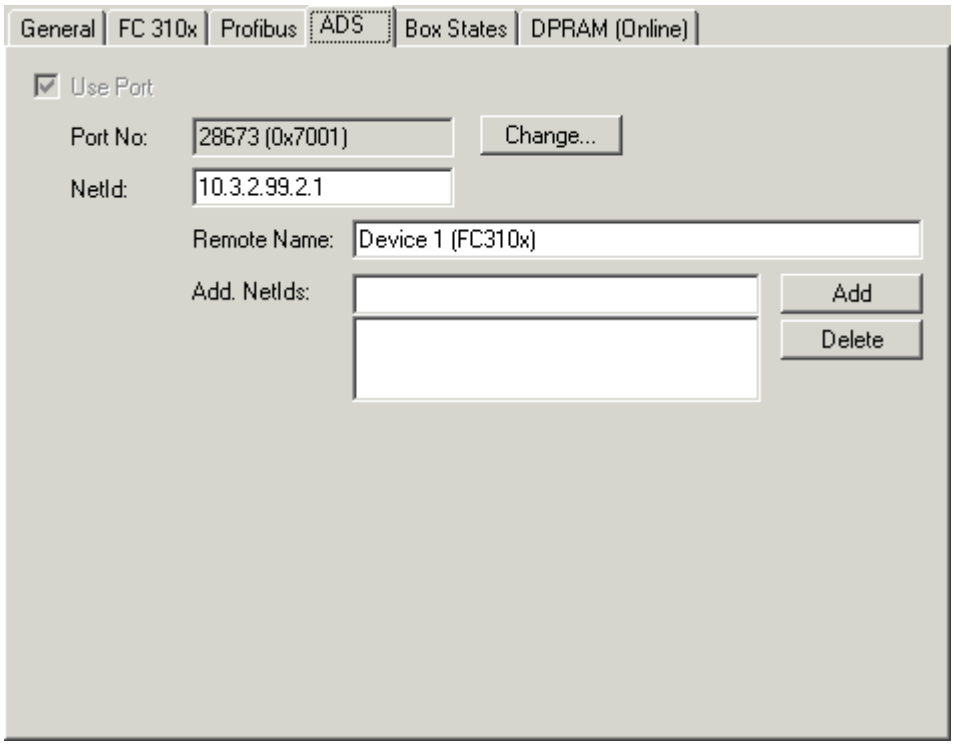

Fig. 30: TwinCAT 2.9 - ADS tab

3.8.2.6 DP Diag tab

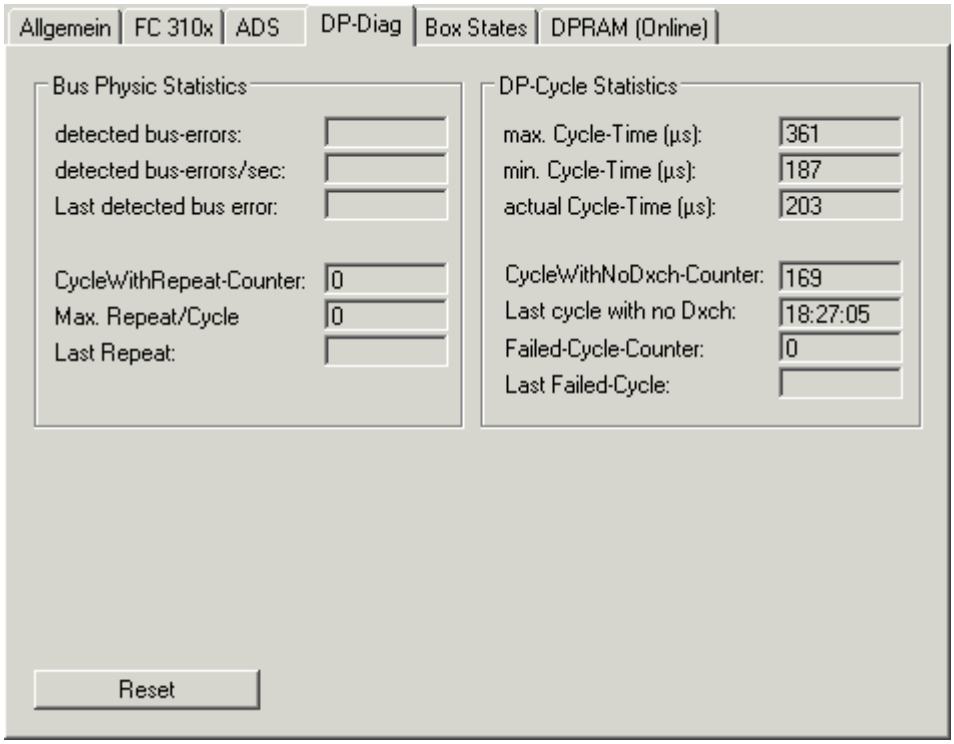

Fig. 31: TwinCAT 2.9 - DP-Diag tab

Here, bus cabling problems and DP cycle times are displayed:

detected bus-errors: Here, the number of detected bus errors is displayed. If this counter is not equal 0, the cabling should be checked (provided that no PROFIBUS connectors were pulled or inserted - usually there are short bus disturbances during pulling or inserting of PROFIBUS connectors).

CycleWithRepeatCounter: Here, the number of PROFIBUS cycles is displayed, in which a telegram was repeated at least once. Repetitions are also an indication that the physical bus characteristics are not 100% OK.

max. Repeat/Cycle: Here, the maximum number of repetitions within a cycle is displayed.

min./max./actual Cycle-Time: Here, the minimum, maximum and current DP cycle time is displayed. Only those cycles are considered, during which all slaves participated in the data exchange and no repetitions occurred.

CycleWithNoDxch-Counter: Increments if not all slaves participate in the data exchange (i.e. have a DpState not equal 0)

Real Failed-Cycle-Counter: Increments if the DP cycle was not completed before the next task cycle began and all the slaves are involved in the data exchange (i.e. have a DpState of 0).

3.8.2.7 Box States tab

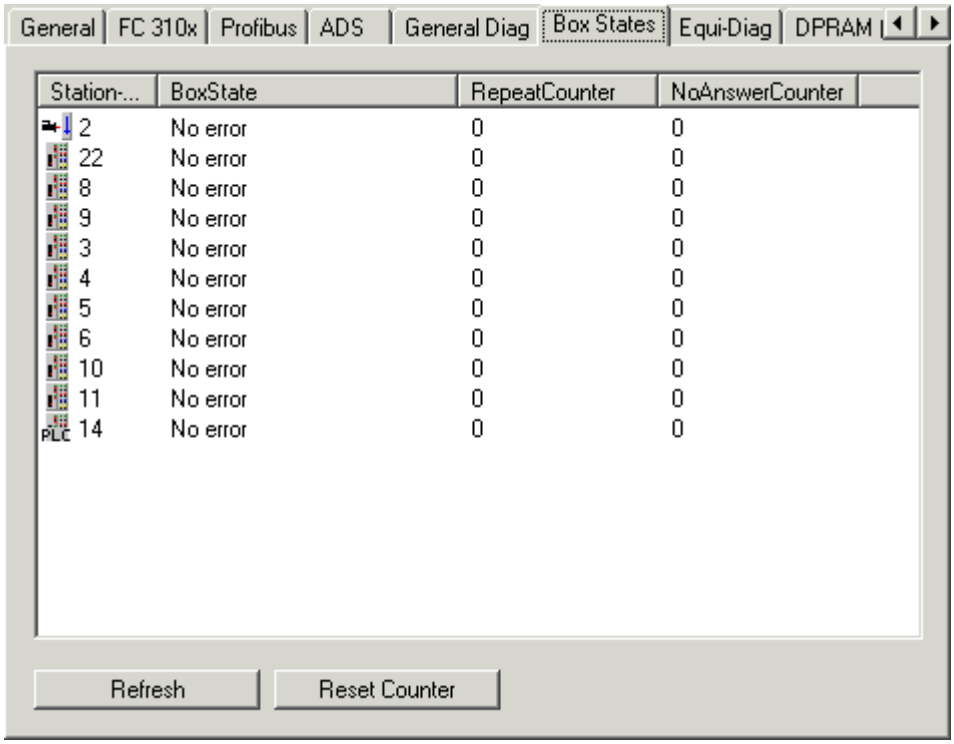

Fig. 32: TwinCAT 2.9 - Box States tab

Here, an overview of all current box states, the Repeat counter (increments for each telegram repeat to the slave) and the NoAnswer counter (increments every time the slave fails to answer) is displayed.

3.8.2.8 MC Diag tab

The "EquiDiag" tab is available for displaying various equidistant monitoring parameters online:

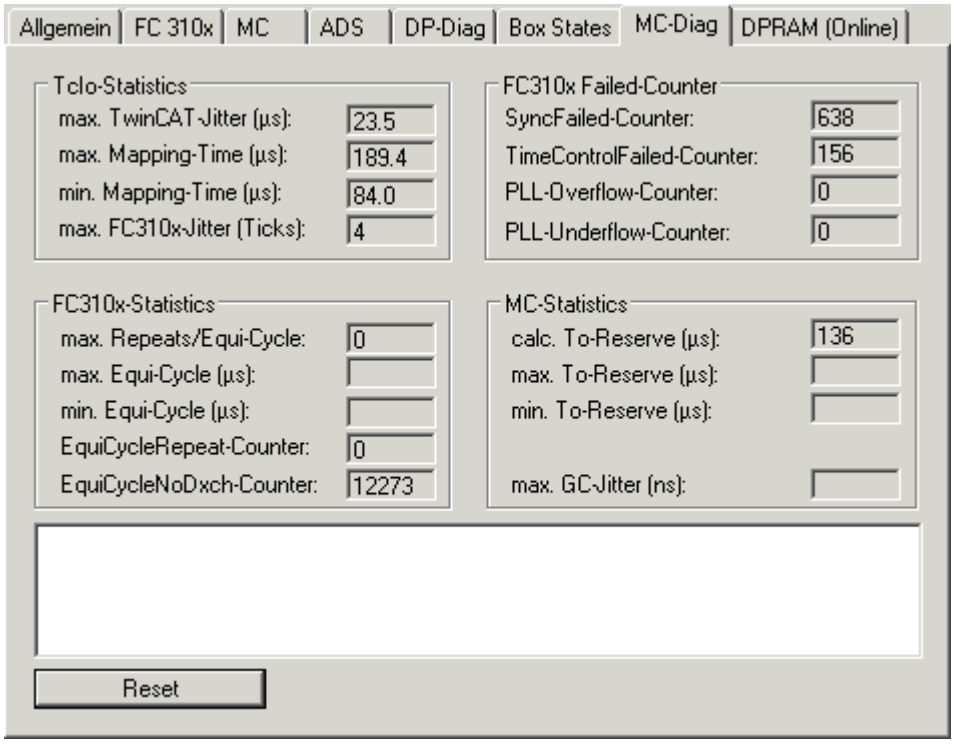

Fig. 33: TwinCAT 2.9 - MC-Diag tab

Tclo-Statistics

max. TwinCAT-Jitter: Here, the maximum TwinCAT jitter is displayed; the measurement is reset as soon as the tab is no longer active.

min./max. Mapping Time or NC Access Time: Here, the minimum or maximum NC Access Time is displayed. Apart from the Mapping Time, this also contains the task runtime (provided that "IO at Task Start" for the highest-priority task linked to the FC310x (usually the NC task) is set).

The NC Access Time should be greater than the sum of the two measured times (max. TwinCAT Jitter plus max. Mapping Time or (NC) Task Time), with a safety margin of approx. 10%.

max. FCxxxx Jitter: This is only relevant for Sync Mode = "disabled". If the value is greater than 5, real-time jitter will become excessive, and a more powerful PC should be used. If the associated (NC) task does not have the highest priority, the maximum FCxxxx jitter can also be larger. This should be avoided.

FC310x-Statistics

Since the PROFIBUS MC slaves should always be added to the FC310x device first, they are always addressed before the DP slaves in the DP cycle. This part of the DP cycle will be called Equi-Cycle below. If the Equi-Cycle becomes greater than the To-time of the MC slaves, the last MC slaves at the FC310x device will usually get a synchronization error (error 597 or 598 for Simodrive 611U).

max. Repeats/Equi-Cycle: This will show the maximum number of repeated Data_Exchange telegrams during an Equi-Cycle. This will extend the Equi-Cycle, usually no repetitions should occur (unless a bus plug was pulled or an MC slave was switched off).

min./max. Equi-Cycle: Here, the minimum or maximum Equi-Cycle Time is displayed.

Equi-Cycle Repeat Counter: Here, the number of telegram repetitions within the Equi-Cycle is displayed.

Equi-Cycle NoDxch Counter: Here, the number of occurrences is displayed, for which not all MC slaves were in data exchange during an Equi-Cycle.

FC310x-Failed Counter

Sync-Failed Counter: This counter will increment if TwinCAT task and DP cycle are not synchronized with each other. This may happen during start-up of the TwinCAT system, after which this counter should no longer increment. If the associated (NC) task does not have the highest priority, this counter can also increment. This should be avoided.

Time-Control-Failed Counter: This counter will increment if the PROFIBUS was not free at the time of the DP cycle start. Possible causes are bus faults, non-existent device, a second master or a safety time that is too small.

PLL-Overflow-/-Underflow-Counter: This counter is only relevant during Sync mode "disabled" and will increment in case of excessive jitter of the TwinCAT task which the DP cycle uses for synchronization (this may happen, for example, if the DP cycle is not synchronized with the highest-priority task). If the associated (NC) task does not have the highest priority, this counter can also increment. This should be avoided.

To-Statistics

For each MC slave the To-time specifies when, relative to the DP cycle start, the slave should accept the outputs received from the master. The MC slaves can be synchronized with each other, if the same To value is set for all MC slaves. However, this value must be equal or greater than the Equi-Cycle Time plus a safety margin of approx. 200 µs. The To-time for all MC slaves is calculated with the button "Calc. Equi-Times" (see above).

calc. To-Reserve: This contains the calculated To-Reserve (To-Time - Equi-Cycle-Time).

min./max. To-Reserve: Here, the min. or max. To-Reserve is measured.

Max. GC jitter (from TwinCAT 2.8)

Here, the maximum jitter of the DP cycle is measured (GC for global control telegram, which is always sent at the start of a cycle). During start-up, the jitter may be somewhat greater, in the steady state it should not exceed 1 µs (for Sync mode "Sync Master") or 2 µs (for Sync mode "Disabled").

3.9 Box tab

3.9.1 Profibus tab

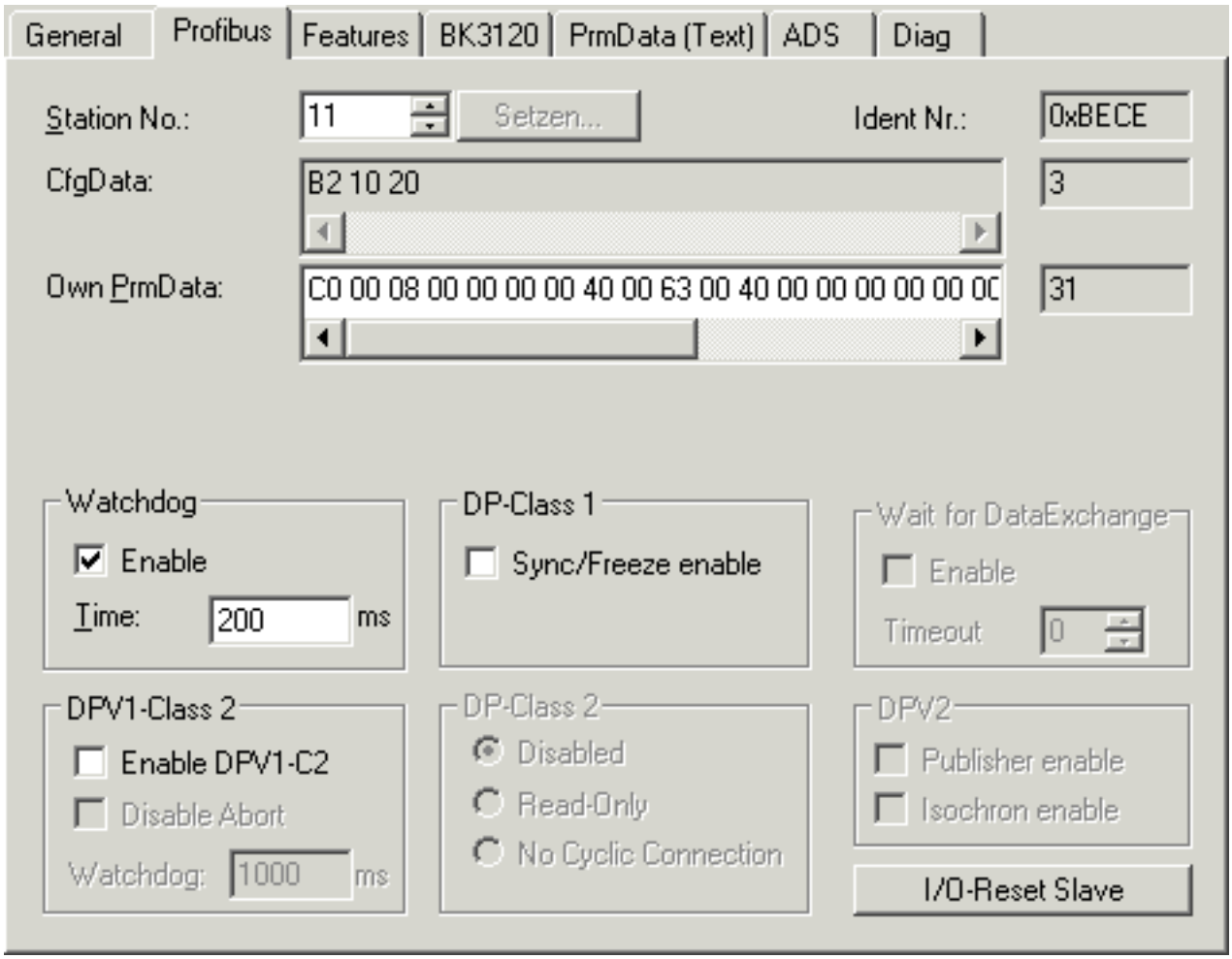

Fig. 34: Profibus tab

Stations No.: Here, the PROFIBUS station address must be set for each slave. For some slaves, the station address cannot be set in the hardware, but only via the SetSlaveAddress service. In this case, the button "Set.." should be pressed. This will open a dialog, through which transmission of a SetSlaveAddress telegram can be triggered.

Watchdog: Activates the DP watchdog. If the slave does not receive a DP telegram for the duration of the watchdog time with the watchdog switched on, it will automatically exit the data exchange. The minimum watchdog time to be set depends on the DP cycle time, and should be larger than the value calculated by the following formula: **Estimated-Cycle-Time * 10**

For particularly critical outputs it is possible to set a DP watchdog time down to as little as 2 ms for DP slaves that support a watchdog base time of 1 ms (namely all Beckhoff slaves with the exception of the BK3000 and BK3100, and any third-party devices whose GSD file contains the entry "WD_Base_1ms_supp = 1"). The DP watchdog time should, however, be at least twice as long as the greater of the **Cycle time** and the **Estimated cycle time** (see the master's FC310x [▶ 40] tab).

Ident No.: Here, the Ident number from the GSD file is displayed.

PrmData: Allows editing of the Profibus-specific parameter data. The values of the current parameter data are also displayed. The PrmData can usually be set as text (-> PrmData (text)) or for Beckhoff DP slaves partly via the "Beckhoff" tab

CfgData: The current configuration data (resulting from the attached modules or terminals) as well as their length is displayed.

Sync/Freeze: In operation mode DP/MC (equidistant) of the master, slaves can be operated with Sync and Freeze [▶ 15].

DPV1 Class 2: With FC310x, a DPV1 class 2 connection to a DPV1 slave can be activated. This is a good idea, for example, if the DP slave is in data exchange with another master, but should nevertheless be addressed acyclically by TwinCAT. The class 2 connection monitoring time is set via the timeout parameter (see chapter DPV1 [▶ 31]).

DP Class 2: "No Cyclic Connection" or "ReadOnly" should be selected under DP class 2, if the DP slave is in data exchange with another master, but should nevertheless be addressed acyclically by TwinCAT, or the DP inputs and outputs should be read cyclically. If "ReadOnly" is selected, then the modules are to be selected as in the case of the normal cyclic connection. They all, however, appear in the TwinCAT system with input variables, regardless of whether they are in fact input or output modules (ReadOnly only as from firmware version 3.00).

ResetSlave: With this button, provided TwinCAT has been started, cyclic data exchange with the DP slave can be disabled and re-established immediately (corresponds to a IO reset but only for the one slave).

3.9.2 Features tab

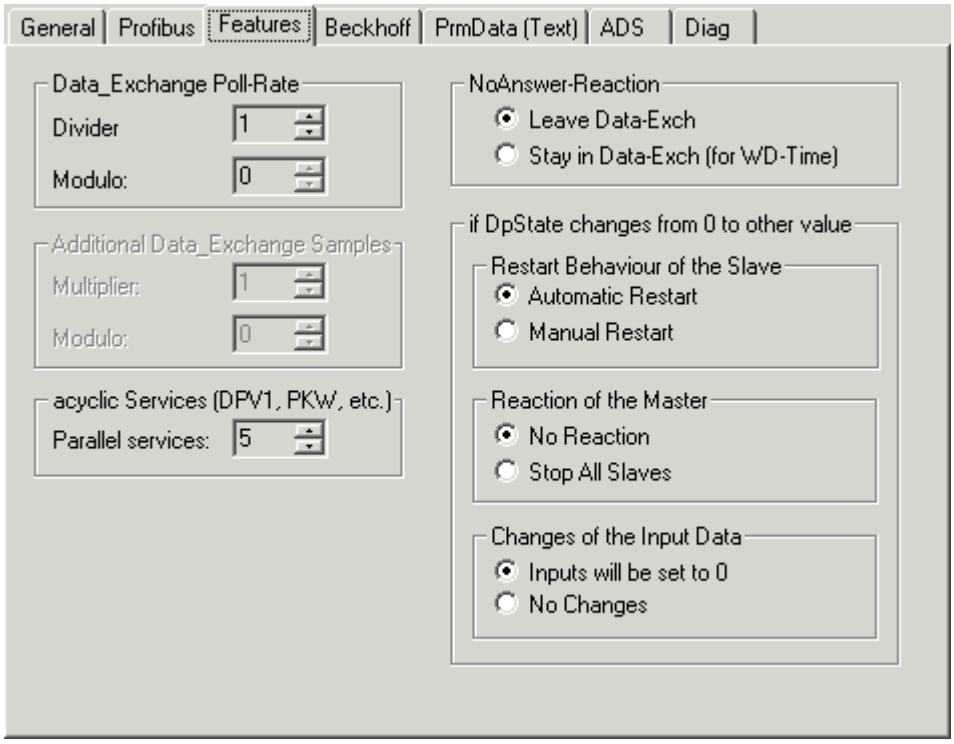

Fig. 35: Features tab

Data_Exchange Poll Rate: For each slave a different polling rate (divider) can be set. Divider 1 means that the slave is polled in each cycle, Divider 2 means every second cycle, etc. The Modulo parameter can be used to allocate slaves with divider greater than 1 to different cycles, in order to reduce the maximum cycle time (Divider 2 and Modulo 0 means that the slave is polled in each even cycle, Divider 2 and Modulo 1 means that the slave is polled in each odd cycle) (see chapter Slave prioritization/several DP cycles [▶ 14]).

Additional Data_Exchange Samples: It is possible to run a number of DP cycles within one task cycle. It is then optionally possible for each slave to be supplied with different output data in each DP cycle, and for the input data from each of those DP cycles to be transferred to the controller. In this case, there is an individual set of variables for each DP cycle (see the Slave Prioritization/Multiple DP Cycles [▶ 14] chapter).

NoAnswer-Reaction: You can specify, for each slave, whether it should remain in the Data Exch, despite responding incorrectly or not at all. In this case (Stay in Data-Exch), data exchange is only exited if the slave has never responded correctly within the address monitoring time (provided the **watchdog** (see the box's Profibus [▶ 58] tab) is activated, otherwise data exchange is only terminated once the slave has not responded correctly 65535 times) (see the Error Reactions [▶ 16] chapter).

Restart-Behaviour: It is possible to specify for each slave whether, after leaving Data-Exch, it should automatically start up again, or should remain in the Wait-Prm state (see chapter Error Reactions [▶ 16]).

Reaction of the Master: You can specify for each slave, whether its exit from Data-Exch should cause the PROFIBUS cycle to stop (all slaves abandon data exchange and go into Wait Prm mode, restart after IO reset or TwinCAT system restart) (see chapter Error Reactions [▶ 16]).

Changes of the Input Data: For each slave it can be specified whether, on exiting of Data Exch (DpState not equal 0), its input data should be set to 0 or remain unchanged (see chapter Error Reactions [▶ 16]).

acyclic Services: The number of parallel ADS services to one box can be set here.

3.9.3 Beckhoff tab

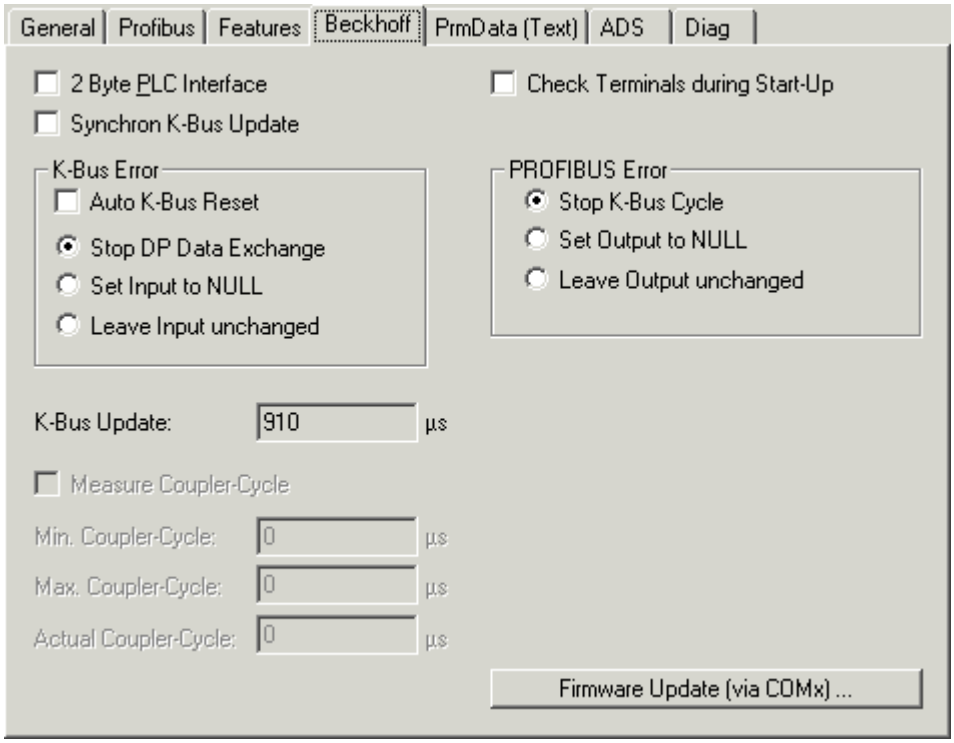

Fig. 36: Beckhoff tab

FirmwareUpdate: This button enables updating of the firmware of a Beckhoff DP slave over a KS2000 cable via the serial interface

2 byte PLC interface: Switches on the 2 byte PLC interface of the Beckhoff DP slave.

Synchron K-Bus-Update: For Bus Couplers, the expected internal cycle time (K-bus cycle + DP buffer transfer) is specified here.

Check Terminals during Start-Up: If this checkbox is activated, table 9 is transferred to the coupler by means of DPV1 write, and the coupler only enters data exchange (DpState = 0) if the entries agree. This makes it possible to check the terminals more precisely when starting up than is done with PROFIBUS CfgData.

K-Bus Error: It is possible to specify the reaction to a K-bus error here (automatic or manual K-bus reset, reaction in the input data in the coupler)

PROFIBUS Error: It is possible to specify the reaction to a PROFIBUS error here (reaction in the output data in the coupler)

Measure Coupler-Cycle: It is possible here to measure the cycle time for the coupler (DP + K-Bus); the K-bus update time is sometimes rather imprecise for synchronized processes.

3.9.4 ProcessData tab

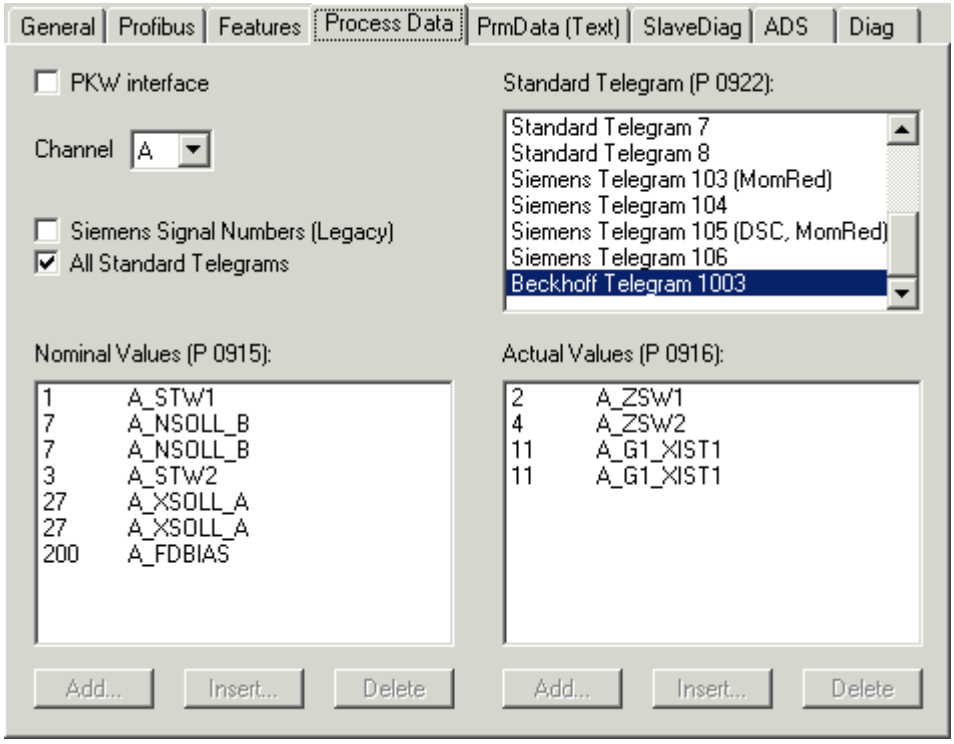

Fig. 37: ProcessData tab

Under Siemens AG or Profidrive MC, the boxes Profidrive MC and Profidrive MC (twice) may be selected. By default the box is allocated the settings for a Simodrive 611U with standard telegram 3. For other PROFIBUS MC slaves, the ID number on the Profibus [▶ 58] tab of the slave has to be changed accordingly. The required telegram type can be set on the **ProcessData** tab and must usually correspond with parameter 922 of the PROFIBUS MC slave, which can be set via a manufacturer-specific configuration tool (SimoCon U for Simodrive 611U).

This tab also contains a "PKW interface" checkbox, which activates the PKW interface. In online mode, this can be used to display the box parameters on the Online tab (at present this only works with the Simodrive 611U, since a parameter file is required, which is generated by the SimoCon U). In any case it is possible to read and write the parameters via the PKW interface per ADS (see chapter PKW Protocol [▶ 35]).

3.9.5 PrmData (text) tab

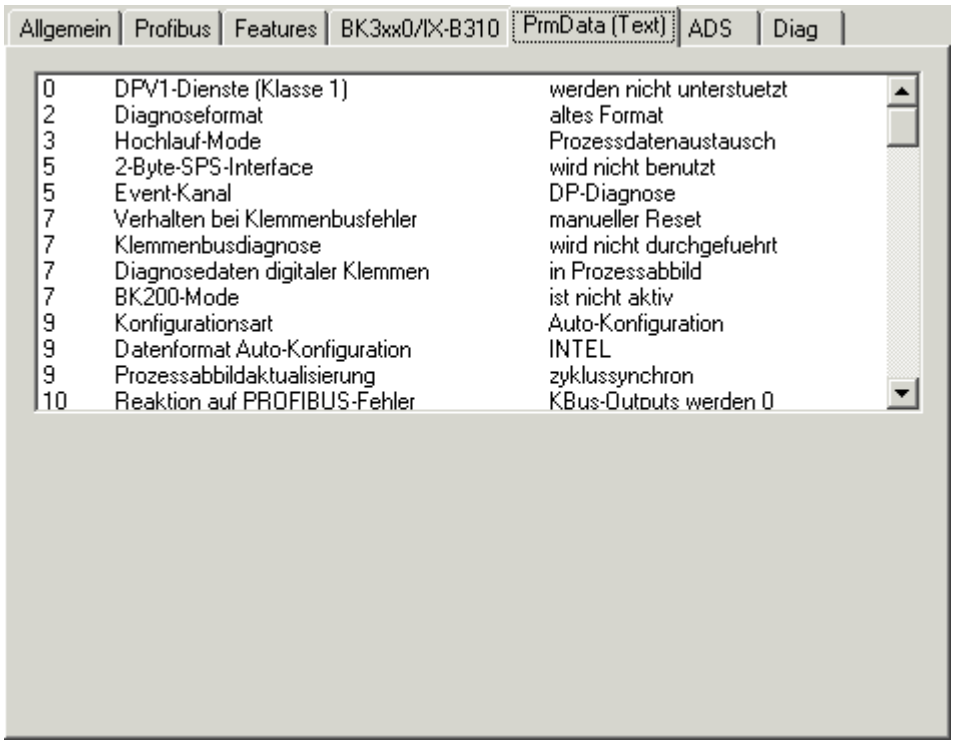

Fig. 38: PrmData (text) tab

Click on a line to change the current value. The description of the respective settings can be found in the documentation of the relevant manufacturer.

3.9.6 Diag tab

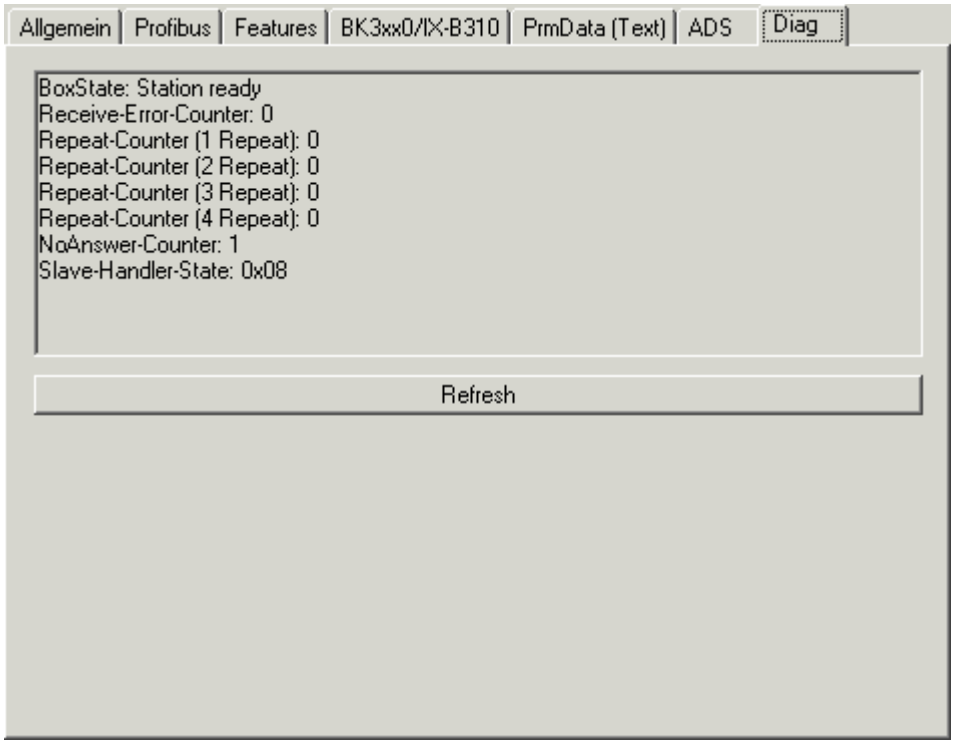

Fig. 39: Diag tab

The following information can be displayed here:

BoxState: The current DpState [▶ 25] is displayed here.

Receive-Error-Counter: Number of disturbed telegrams from the slave.

Repeat Counter: Number of required repetitions due to missing or disturbed response from the slave.

NoAnswer-Counter: Number of telegrams that remained unanswered by the slave.

Last DPV1 error: Error-Decode, Error-Class, Error-Code and Error-Code 2 (see description of the DPV1 Error Codes [▶ 71]).

For Beckhoff DP slaves, further diagnostic information will be displayed.

4 FC310x as slave

4.1 Slave

As a slave, the PROFIBUS DP and PROFIBUS DPV1 protocols are supported.

PROFIBUS DP

In order to configure the slave for cyclic DP operation, proceed as follows in the TwinCAT System Manager:

Configure DP slave

It is first necessary to configure a "PROFIBUS Slave FC310x, PCI" I/O device (right-click on "I/O devices", and then selecting "Append Device"). The device and a box are appended (to this end the GSD file "TCDPSLAV.GSD" must be in directory "TwinCAT\Io\PROFIBUS"):

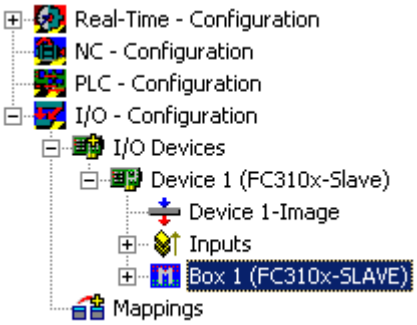

Fig. 40: FC310x appended as slave

Find the corresponding channel on the tab **"FC310x"** (TwinCAT 2.8 [▶ 40] or TwinCAT 2.9 [▶ 48]) of the device ("Search" button), adjust the station address and the baud rate, if necessary (the default setting is 12 Mbit/s).

Append modules

Modules are to be appended to the box corresponding to the data that is to be transferred cyclically. This is done by clicking with the right mouse button on the box, and then selecting "Append modules":

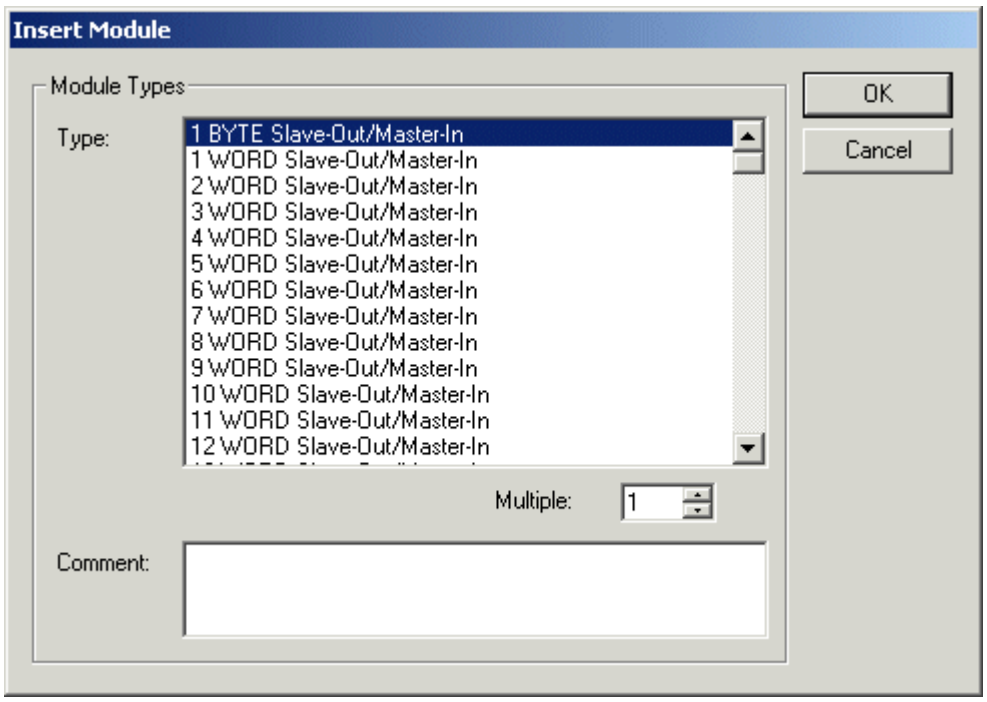

Fig. 41: Append modules

Configuring the Master

The GSD file "TCDPSLAV.GSD" is to be taken out of the "TwinCAT\Io\PROFIBUS" directory to configure the master. The modules must be inserted in the master configuration **in the same sequence** as they are in the configuration of the slave in the System Manager.

Error Reactions

In the default setting the inputs of the DP slave are set to 0, if the DP slave is not involved in data exchange. It is possible to change this input error reaction to "No changes" under **Changes of the Input Data** on the "Features" tab for the box:

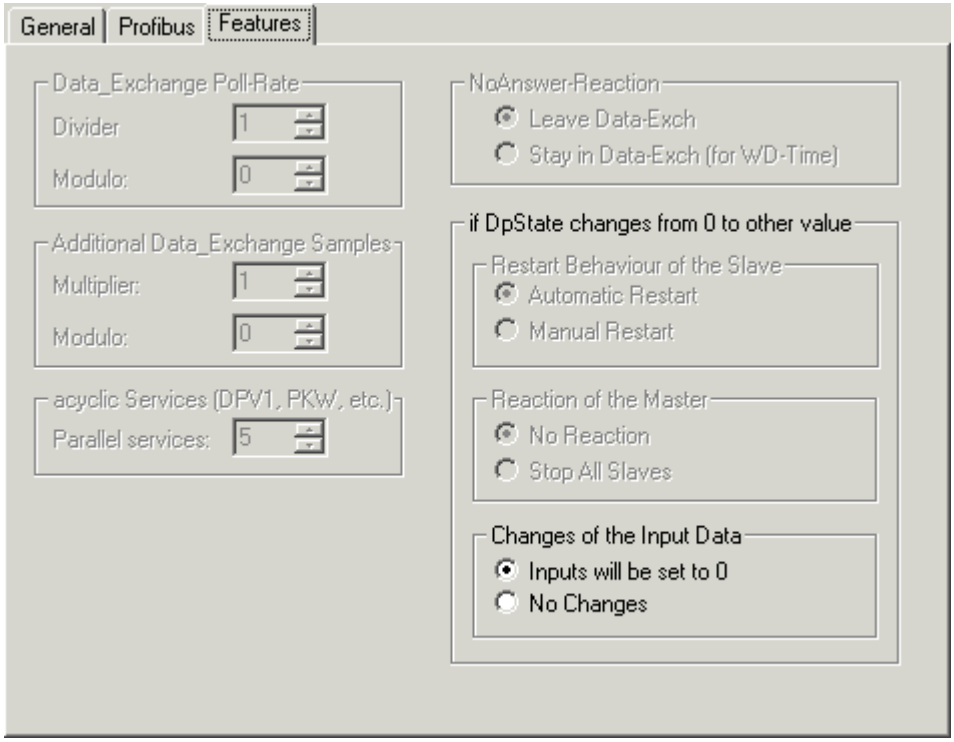

Fig. 42: Features

PROFIBUS DPV1

The DP slave supports a DPV1-MSAC_C1 server connection that is established along with the cyclic connection. This can be used so that larger quantities of acyclic data can be transferred alongside the cyclic data. A DPV1 read telegram received by the master is reported to the PLC as an ADS read indication, while a DPV1 write telegram is reported to the PLC as an ADS write indication. The PLC program is then responsible for the read or write response. To do this, the ADS read response or ADS write response functions are to be called.

MSAC-C1 Read

A DPV1-MSAC_C1 read indication is mapped in an ADS read indication as follows:

ADS read indication parameter	Meaning
Source-Net-ID (NETID)	Net-ID of the slave (see the device's ADS [▶ 44]tab)
Source-Port (PORT)	0x200
Invoke-ID (INVOKEID)	A unique number that must reappear in the response
IndexGroup (IDXGRP)	Slot number (DPV1 parameter)
IndexOffset (IDXOFFS)	Index (DPV1 parameter)
Length (LENGTH)	Length of the data that is to be read

An ADS read response is mapped in a DPV1-MSAC_C1 read response as follows:

ADS read response parameter	Meaning
Destination-Net-ID (NETID)	Net-ID of the slave (see the device's ADS [▶ 44] tab)
Destination-Port (PORT)	0x200
Invoke-ID (INVOKEID)	A unique number, as under indication
Result (RESULT)	Result of the read: 0 = no error, otherwise: bits 0-15 = standard ADS error codes, bits 16-23 = Error_Code_1, bits 24-31 = Error_Code_2, see description of the DPV1 error codes [▶ 71]
Length (LENGTH)	Length of the data that has been read
Data (DATAADDR)	read data

MSAC-C1 Write

A DPV1-MSAC_C1 write indication is mapped in an ADS write indication as follows:

ADS write indication parameter	Meaning
Source-Net-ID (NETID)	Net-ID of the slave (see the device's ADS [▶ 44] tab)
Source-Port (PORT)	0x200
Invoke-ID (INVOKEID)	A unique number that must reappear in the response
IndexGroup (IDXGRP)	Slot number (DPV1 parameter)
IndexOffset (IDXOFFS)	Index (DPV1 parameter)
Length (LENGTH)	Length of the data that is to be written
Data (DATAADDR)	data that is to be written

An ADS read response is mapped in a DPV1-MSAC_C1 read response as follows:

ADS read response parameter	Meaning
Destination-Net-ID (NETID)	Net-ID of the slave (see the device's ADS [▶ 44] tab)
Destination-Port (PORT)	0x200
Invoke-ID (INVOKEID)	A unique number, as under indication
Result (RESULT)	Result of the read: 0 = no error, otherwise: bits 0-15 = standard ADS error code, bits 16-23 = Error_Code_1, bits 24-31 = Error_Code_2, see description of the DPV1 error codes [▶ 71]
Length (LENGTH)	Length of the data that has been read

ADS Interface from TwinCAT 2.9

Communication can also take place via ADS in TwinCAT systems. The functionality is extremely similar to that of an ADS connection between two PCs over Ethernet, although transmission takes place over PROFIBUS, with the exception that the requester that initiates the ADS job is always the DP master PC. An FC310x DP master is then linked to an FC310x-DP slave.

In the DP master, the **ADS Interface** should be activated on the ADS tab of the box, and the **Net-ID** of the DP slave PC is to be entered:

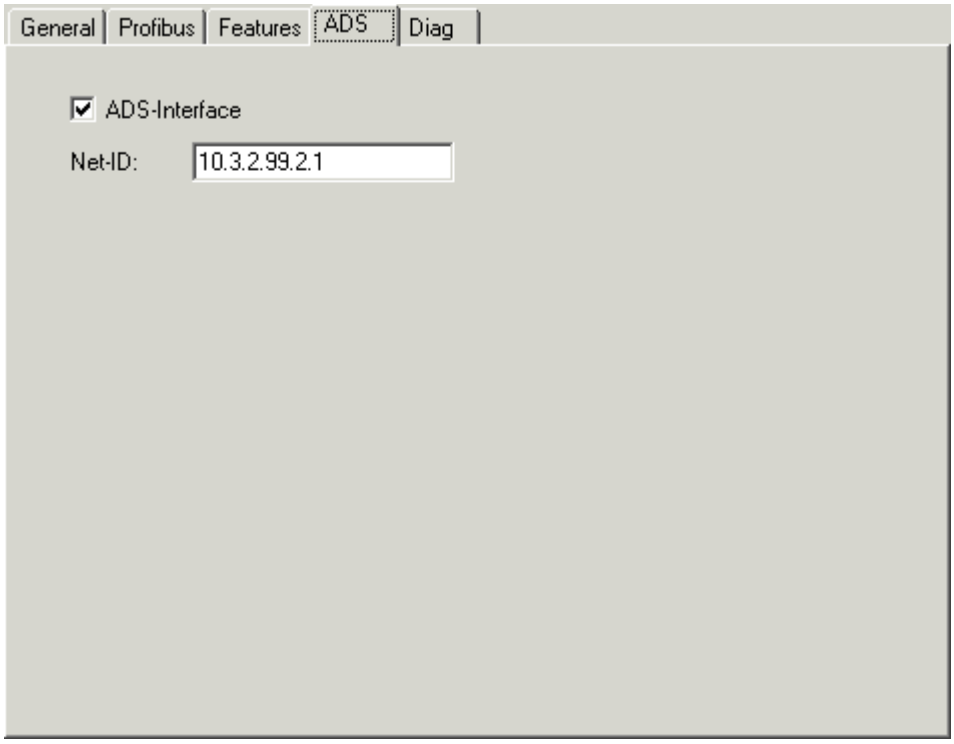

Fig. 43: Enabling the ADS interface on the DP master

For the DP slave, the Net-ID of the DP master PC should be entered under **Add. NetIDs** in the ADS tab of the device:

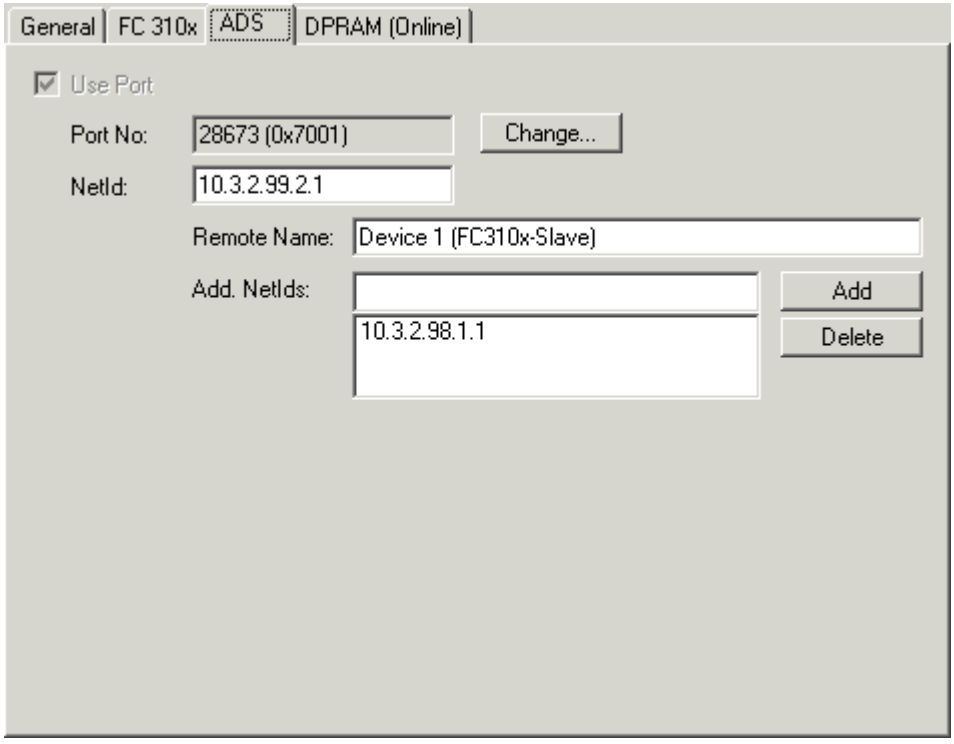

Fig. 44: ADS tab on the DP slave

5 Appendix

5.1 Diagnostic Data - DiagData

There follows a description of the DP diagnostic data

Offset	Meaning
0x00.0	StationNonExistent: slave did not reply to the last telegram
0x00.1	StationNotReady: slave still processing the Set_Prm / Chk_Cfg telegram
0x00.2	CfgFault: slave signaling a configuration error
0x00.3	ExtDiag: extended DiagData available and valid
0x00.4	NotSupported: slave does not support a feature requested via Set_Prm or Global_Control
0x00.5	InvalidSlaveResponse: slave response not DP-compatible
0x00.6	PrmFault: slave reports a parameterization error
0x00.7	MasterLock: slave currently exchanging data with another master
0x01.0	PrmReq: re-parameterize and reconfigure slave
0x01.1	StatDiag: slave signaling static diagnostics / DPV1 slave application not yet ready for data exchange
0x01.2	PROFIBUS-DP-Slave
0x01.3	WdOn: DP watchdog switched on
0x01.4	FreezeMode: DP slave in freeze mode
0x01.5	SyncMode: DP slave in sync mode
0x01.6	reserved
0x01.7	Deactivated: DP slave has been deactivated
0x02.0	reserved
0x02.1	reserved
0x02.2	reserved
0x02.3	reserved
0x02.4	reserved
0x02.5	reserved
0x02.6	reserved
0x02.7	ExtDiagOverflow: too much extended data present
0x03	MasterAdd: station address of master with which slave is exchanging data
0x04,0x05	IdentNumber
from 0x06	Extended DiagData

Extended DiagData

A distinction is made in the Extended DiagData between identification diagnosis, channel diagnosis and manufacturer-specific diagnosis. The first byte indicates the type of the diagnosis and the length of the associated data. In the Extended DiagData several diagnostic types can be entered consecutively.

Header Byte

Bit	Meaning
0-5	Length of the associated diagnostic data, including header byte
6-7	0 = manufacturer-specific diagnostics (DPV1 is not supported) or DPV1 diagnostics (DPV1 is supported (DPV1_Enable = 1) in associated GSD file)
	Module diagnostics
	Channel diagnostics
	Revision number

Manufacturer-specific diagnostics

The structure of the manufacturer-specific diagnostics may be found in the documentation for the DP slave.

DPV1 diagnostics

In the case of DP slaves that also support DPV1, the DPV1 diagnostics, in which a distinction is made between status messages and alarms, is sent instead of the manufacturer-specific diagnostics.

Byte	Meaning
0	Header-Byte (bits 6,7 = 0, bits 0-5 = 4..63)
1	Bits 0-6: Alarm type
	Bit 7: always 0
2	Slot number (0-254)
3	Bits 0-1: Alarm specifier
	Bit 2: Additional acknowledge
	Bits 3-7: Sequence number
4-63	Manufacturer specific (see the documentation for the DP slave)

Alarm type

Value	Meaning
0	reserved
1	Diagnostic alarm
2	Process alarm
3	Withdrawal alarm
4	Insertion alarm
5	Status alarm
6	Update alarm
7-31	reserved
20-126	Manufacturer specific (see the documentation for the DP slave)
127	reserved

Module diagnostics

The module diagnostics contain one bit for each DP module. The bit indicates whether a diagnosis for the corresponding DP module is present.

Byte	Meaning
0	Header byte (bits 6,7 = 1, bits 0-5 = 2..32)
1	Bit 0: 1st DP module has diagnostics
	Bit 1: 2nd DP module has diagnostics
	...
	Bit 7: 8th DP module has diagnostics
...	...
31	Bit 0: 241th DP module has diagnostics
	Bit 1: 242th DP module has diagnostics
	Bit 2: 243th DP module has diagnostics
	Bit 3: 244th DP module has diagnostics (a maximum of 244 DP modules is possible)

Channel diagnostics

The channel diagnostics provide a closer description of the cause of the diagnosis of a DP module.

Byte	Meaning
0	Header byte = 0x83 (3 bytes including header, bits 6,7 = 2)
1	Bits 0-5: Channel number
	Bits 6-7: 0 = reserved, 1 = input, 2 = output, 3 = input/output
2	Bits 0-4: Error type
	Bits 5-7: Channel type

Error type

Value	Meaning
0	reserved
1	Short circuit
2	Undervoltage
3	Overvoltage
4	Overload
5	Overtemperature
6	Wire breakage
7	Upper limit value exceeded
8	Value below lower limit
9	Error
10-15	reserved
16-31	Manufacturer specific (see the documentation for the DP slave)

Channel type

Value	Meaning
0	Any type
1	Bit
2	2 bit
3	4 bit
4	Byte
5	Word
6	2 words
7	reserved

Revision number

The structure of the revision number may be found in the documentation for the DP slave.

5.1.1 DPV1 Error Codes

In the event of an incorrect DPV1 access, the slave replies with 4 bytes of data (any values that are not described here are not defined in the DPV1 standard, and are therefore to be found in the slave's manual):

Byte 0	DPV1 service
0xD1	Data_Transport
0xD7	Initiate
0xDE	Read
0xDF	Write

Byte 1	Error_Decode
0x80	DPV1
0xFE	FMS
0xFF	HART

Byte 2		Error_Code_1
Error-Class (bits 4-7)	Error-Code (bits 0-3)	
0x0A	0x00	Application, Read Error
	0x01	Application, Write Error
	0x02	Application, Module Failure
	0x08	Application, Version Conflict
	0x09	Application, Feature Not Supported
0x0B	0x00	Access, Invalid Index
	0x01	Access, Write Length Error
	0x02	Access, Invalid Slot
	0x03	Access, Type Conflict
	0x04	Access, Invalid Area
	0x05	Access, State Conflict
	0x06	Access, Access Denied
	0x07	Access, Invalid Range
	0x08	Access, Invalid Parameter
	0x09	Access, Invalid Type
0x0C	0x00	Resource, Read Constrain Conflict
	0x01	Resource, Write Constrain Conflict
	0x02	Resource, Busy
	0x03	Resource, Unavailable

Byte 3	Error_Code_2

5.2 Configuration Data - CfgData

The CfgData describes the structure and length of the input and output data that is to be cyclically exchanged via Data_Exchange. There follows a description of the DP configuration data bytes

Bits 4-7	Meaning
0000B	Module without data. Bits 0-3 indicate how many bytes of manufacturer-specific data are still to follow
0001B	Inputs of type byte, with no consistency. Bits 0-3 contain the length of the input data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 byte, while bits 0-3 = 1111B corresponds to a length of 16 bytes)
0010B	Outputs of type byte, with no consistency. Bits 0-3 contain the length of the output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 byte, while bits 0-3 = 1111B corresponds to a length of 16 bytes)
0011B	Inputs and outputs of type byte, with no consistency. Bits 0-3 contain the length of the input or output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 byte, while bits 0-3 = 1111B corresponds to a length of 16 bytes)
0100B	A special identification format for inputs. A byte follows that describes the associated input data (see below). Bits 0-3 indicate how many bytes of manufacturer-specific data are still to follow
0101B	Inputs of type word, with no consistency. Bits 0-3 contain the length of the input data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of one word, while bits 0-3 = 1111B corresponds to a length of 16 words)
0110B	Outputs of type word, with no consistency. Bits 0-3 contain the length of the output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 word, while bits 0-3 = 1111B corresponds to a length of 16 words)
0111B	Inputs and outputs of type word, with no consistency. Bits 0-3 contain the length of the input or output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 word, while bits 0-3 = 1111B corresponds to a length of 16 words)
1000B	A special identification format for outputs. A byte follows that describes the associated output data (see below). Bits 0-3 indicate how many bytes of manufacturer-specific data are still to follow
1001B	Inputs of type byte, with consistency. Bits 0-3 contain the length of the input data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 byte, while bits 0-3 = 1111B corresponds to a length of 16 bytes)
1010B	Outputs of type byte, with consistency. Bits 0-3 contain the length of the output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 byte, while bits 0-3 = 1111B corresponds to a length of 16 bytes)
1011B	Inputs and outputs of type byte, with consistency. Bits 0-3 contain the length of the input or output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 byte, while bits 0-3 = 1111B corresponds to a length of 16 bytes)
1100B	A special identification format for inputs and outputs. A byte first follows that describes the associated output data, and then one describing the associated input data (see below). Bits 0-3 indicate how many bytes of manufacturer-specific data are still to follow
1101B	Inputs of type word, with consistency. Bits 0-3 contain the length of the input data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of one word, while bits 0-3 = 1111B corresponds to a length of 16 words)
1110B	Outputs of type word, with consistency. Bits 0-3 contain the length of the output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 word, while bits 0-3 = 1111B corresponds to a length of 16 words)
1111B	Inputs and outputs of type word, with consistency. Bits 0-3 contain the length of the input or output data minus 1 (i.e. bits 0-3 = 0000B corresponds to a length of 1 word, while bits 0-3 = 1111B corresponds to a length of 16 words)

If the first byte has the type "special identification format", then the second or third bytes have the following meaning:

Bits 6-7	Meaning
00B	Type byte, with no consistency. Bits 0-5 contain the length of the data minus 1 (i.e. bits 0-5 = 000000B corresponds to a length of 1 byte, while bits 0-5 = 111111B corresponds to a length of 64 bytes)
01B	Type word, with no consistency. Bits 0-5 contain the length of the data minus 1 (i.e. bits 0-5 = 000000B corresponds to a length of 1 word, while bits 0-5 = 111111B corresponds to a length of 64 words)
10B	Type byte, with consistency. Bits 0-5 contain the length of the data minus 1 (i.e. bits 0-5 = 000000B corresponds to a length of 1 byte, while bits 0-5 = 111111B corresponds to a length of 64 bytes)
11B	Type word, with consistency. Bits 0-5 contain the length of the data minus 1 (i.e. bits 0-5 = 000000B corresponds to a length of 1 word, while bits 0-5 = 111111B corresponds to a length of 64 words)

5.3 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages:
http://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49(0)5246/963-0
Fax:	+49(0)5246/963-198
e-mail:	info@beckhoff.com

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49(0)5246/963-157
Fax:	+49(0)5246/963-9157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49(0)5246/963-460
Fax:	+49(0)5246/963-479
e-mail:	service@beckhoff.com

List of illustrations