

Dokumentation

FC3101 und FC3102

PCI-Karten für PROFIBUS

Version: 3.0
Datum: 17.11.2017

BECKHOFF

Inhaltsverzeichnis

1	Vorwort	5
1.1	Hinweise zur Dokumentation	5
1.2	Sicherheitshinweise	6
1.3	Ausgabestände der Dokumentation	7
2	Produktübersicht	8
2.1	Einleitung	8
2.2	FC310x - Technische Daten	8
2.3	Hardwarebeschreibung	9
3	FC310x als Master	10
3.1	Master	10
3.2	PROFIBUS DP	11
3.3	Synchronisierung	12
3.3.1	Übersicht	12
3.3.2	Slave-Priorisierung/mehrere DP-Zyklen	14
3.3.3	Sync/Freeze-Funktionalität	16
3.4	Diagnose	16
3.4.1	Übersicht	16
3.4.2	Fehlerreaktionen	17
3.4.3	FC310x - Master-Diagnose	20
3.4.4	Slave-Diagnose	24
3.4.5	DP-State der Slaves	26
3.5	PROFIBUS MC	26
3.6	ADS (azyklische Kommunikation)	29
3.6.1	ADS-Interface	29
3.6.2	PROFIBUS DPV1	32
3.6.3	Upload der Konfiguration	35
3.6.4	PKW-Interface von PROFIDRIVE-Slaves	36
3.6.5	S5-FDL-Kommunikation	37
3.6.6	ADS-Error-Codes der FC310x	38
3.7	Master-Redundanz	39
3.8	Karteireiter Device	41
3.8.1	TwinCAT 2.8	41
3.8.2	TwinCAT 2.9	49
3.9	Karteireiter Box	59
3.9.1	Karteireiter Profibus	59
3.9.2	Karteireiter Features	60
3.9.3	Karteireiter Beckhoff	61
3.9.4	Karteireiter ProcessData	62
3.9.5	Karteireiter PrmData (Text)	63
3.9.6	Karteireiter Diag	63
4	FC310x als Slave	65
4.1	Slave	65
5	Anhang	70
5.1	Diagnosedaten - DiagData	70
5.1.1	DPV1-Fehlercodes	73
5.2	Konfigurationsdaten - CfgData	74
5.3	Support und Service	76

1 Vorwort

1.1 Hinweise zur Dokumentation

Zielgruppe

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs- und Automatisierungstechnik, das mit den geltenden nationalen Normen vertraut ist.

Zur Installation und Inbetriebnahme der Komponenten ist die Beachtung der Dokumentation und der nachfolgenden Hinweise und Erklärungen unbedingt notwendig.

Das Fachpersonal ist verpflichtet, für jede Installation und Inbetriebnahme die zu dem betreffenden Zeitpunkt veröffentlichte Dokumentation zu verwenden.

Das Fachpersonal hat sicherzustellen, dass die Anwendung bzw. der Einsatz der beschriebenen Produkte alle Sicherheitsanforderungen, einschließlich sämtlicher anwendbaren Gesetze, Vorschriften, Bestimmungen und Normen erfüllt.

Disclaimer

Diese Dokumentation wurde sorgfältig erstellt. Die beschriebenen Produkte werden jedoch ständig weiter entwickelt.

Wir behalten uns das Recht vor, die Dokumentation jederzeit und ohne Ankündigung zu überarbeiten und zu ändern.

Aus den Angaben, Abbildungen und Beschreibungen in dieser Dokumentation können keine Ansprüche auf Änderung bereits gelieferter Produkte geltend gemacht werden.

Marken

Beckhoff®, TwinCAT®, EtherCAT®, Safety over EtherCAT®, TwinSAFE®, XFC® und XTS® sind eingetragene und lizenzierte Marken der Beckhoff Automation GmbH.

Die Verwendung anderer in dieser Dokumentation enthaltenen Marken oder Kennzeichen durch Dritte kann zu einer Verletzung von Rechten der Inhaber der entsprechenden Bezeichnungen führen.

Patente

Die EtherCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente: EP1590927, EP1789857, DE102004044764, DE102007017835 mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.

Die TwinCAT-Technologie ist patentrechtlich geschützt, insbesondere durch folgende Anmeldungen und Patente: EP0851348, US6167425 mit den entsprechenden Anmeldungen und Eintragungen in verschiedenen anderen Ländern.



EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizenziert durch die Beckhoff Automation GmbH, Deutschland

Copyright

© Beckhoff Automation GmbH & Co. KG, Deutschland.

Weitergabe sowie Vervielfältigung dieses Dokuments, Verwertung und Mitteilung seines Inhalts sind verboten, soweit nicht ausdrücklich gestattet.

Zu widerhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patent-, Gebrauchsmuster- oder Geschmacksmustereintragung vorbehalten.

1.2 Sicherheitshinweise

Sicherheitsbestimmungen

Beachten Sie die folgenden Sicherheitshinweise und Erklärungen!
Produktspezifische Sicherheitshinweise finden Sie auf den folgenden Seiten oder in den Bereichen Montage, Verdrahtung, Inbetriebnahme usw.

Haftungsausschluss






Die gesamten Komponenten werden je nach Anwendungsbestimmungen in bestimmten Hard- und Software-Konfigurationen ausgeliefert. Änderungen der Hard- oder Software-Konfiguration, die über die dokumentierten Möglichkeiten hinausgehen, sind unzulässig und bewirken den Haftungsausschluss der Beckhoff Automation GmbH & Co. KG.

Qualifikation des Personals

Diese Beschreibung wendet sich ausschließlich an ausgebildetes Fachpersonal der Steuerungs-, Automatisierungs- und Antriebstechnik, das mit den geltenden Normen vertraut ist.

Erklärung der Symbole

In der vorliegenden Dokumentation werden die folgenden Symbole mit einem nebenstehenden Sicherheitshinweis oder Hinweistext verwendet. Die Sicherheitshinweise sind aufmerksam zu lesen und unbedingt zu befolgen!

 GEFAHR	Akute Verletzungsgefahr! Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht unmittelbare Gefahr für Leben und Gesundheit von Personen!
 WARNUNG	Verletzungsgefahr! Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, besteht Gefahr für Leben und Gesundheit von Personen!
 VORSICHT	Schädigung von Personen! Wenn der Sicherheitshinweis neben diesem Symbol nicht beachtet wird, können Personen geschädigt werden!
 Achtung	Schädigung von Umwelt oder Geräten Wenn der Hinweis neben diesem Symbol nicht beachtet wird, können Umwelt oder Geräte geschädigt werden.
 Hinweis	Tipp oder Fingerzeig Dieses Symbol kennzeichnet Informationen, die zum besseren Verständnis beitragen.

1.3 Ausgabestände der Dokumentation

Version	Kommentar
3.0	<ul style="list-style-type: none">• Migration
2.2	<ul style="list-style-type: none">• Layout und Vorwort aktualisiert
2.1	<ul style="list-style-type: none">• Technische Daten und Sicherheitshinweise ergänzt
2.0	<ul style="list-style-type: none">• Technische Beschreibungen ergänzt
1.0	<ul style="list-style-type: none">• Erste Ausgabe

2 Produktübersicht

2.1 Einleitung

Im Folgenden werden die Funktionalitäten der Beckhoff PCI-Karte FC310x (als [Master](#) [[▶ 10](#)] und [Slave](#) [[▶ 11](#)]) für den Einsatz unter TwinCAT (NCI, PTP, PLC und IO) beschrieben.

TwinCAT 2.7 unterstützt nicht alle im Folgenden beschriebenen Funktionen. TwinCAT 2.8 hingegen unterstützt alle nachfolgend beschriebenen Funktionen mit Ausnahme der Redundanz und des ADS-Server-Interfaces der FC310x als Slave (beides wird erst ab TwinCAT 2.9 unterstützt).

Die folgenden Kapitel gelten auch für die PROFIBUS-Anschaltung des CX1000 (CX1500-M310 (Master) bzw. CX1500-B310 (Slave)), die Bezeichnung FC310x bezieht sich dann auch auf die CX1500-M310-Master bzw. CX1500-B310-Slave-Anschaltung.

2.2 FC310x - Technische Daten

Technische Daten	FC3101	FC3102
Bussystem	PROFIBUS DP (Standard), PROFIBUS DP-V1 (Cl. 1+2: azyklische Dienste, Alarmer), DP-V2, PROFIBUS MC (Äquidistanz)	
Anzahl der Feldbuskanäle	1	2
Übertragungsrate	9,6 kbit/s – 12 Mbit/s ¹⁾	
Diagnose	2 LEDs je Kanal	
Interface zum PC	Plug-and-play-PCI-Interface 32 Bit mit 4 kByte DPRAM je Kanal	
Businterface	1 x Buchse D-Sub, 9-polig, galvanisch entkoppelt	2 x Buchse D-Sub, 9-polig, galvanisch entkoppelt
Kommunikation	Master- und Slavefunktionalität (auch gemischt)	
Busteilnehmer	je Kanal: max. 125 Slaves mit bis zu 244 Bytes Input-, Output-, Parameter-, Konfigurations-, Diagnosedaten je Slave	
Prozessabbild	Summe max.: 3 kBytes Input- und Outputdaten	
Zykluszeit	durch CDL-Konzept sind unterschiedliche DP-Zykluszeiten je Slave möglich	
Standard-Treiber	Standard Betriebssystem-Treiber für Intel-kompatible NIC	
Echtzeit-Treiber	TwinCAT-Treiber für Echtzeit Ethernet	
Versorgungsspannung (PCI-Bus)	5 V	
Stromaufnahme (PCI-Bus, 5 V)	typ. 600 mA	typ. 1000 mA
Leistungsaufnahme aus PCI-Bus	< 5 W	
Gewicht	ca. 45 g	ca. 75 g
Abmessungen (B x H x T, ohne Slot-Blende)	ca. 14 mm x 106 mm x 175 mm	
zulässiger Umgebungstemperaturbereich im Betrieb	0°C ... + 55°C	
zulässiger Umgebungstemperaturbereich bei Lagerung	-25°C ... + 85°C	
zulässige relative Luftfeuchtigkeit	95%, keine Betauung	
Vibrations- / Schockfestigkeit	gemäß EN 60068-2-6 / EN 60068-2-27	
EMV-Festigkeit / Aussendung	gemäß EN 61000-6-2 / EN 61000-6-4	
Einbaulage	beliebig	
Zulassung	CE	

¹⁾ ab Hardware Stand 10 werden 6 MBit/s der FC3101 oder FC3102 und deren Derivate nicht mehr unterstützt.

2.3 Hardwarebeschreibung

Eine FC3102 PROFIBUS-Feldbuskarte verhält sich logisch wie zwei FC3101-Karten, d.h. alle Komponenten (mit Ausnahme des PCI-Interfaces und des optionalen NOV-RAMs) sind auf der FC3102-Karte doppelt vorhanden. Jeder Kanal der FC310x-Karte besteht aus den folgenden Komponenten:

- 80165-Mikrocontroller mit 25 MHz
- 512 k RAM
- 256 k Flash
- 4 k DP-RAM
- Feldbus-Interface (Altera 6016 mit 48 MHz, RS485-Busanschaltung, 9-poliger SUB-D-Stecker)
- 2 LEDs (1 x grün, 1 x rot)
- 1 vierpoliger und ein zweipoliger Stecker für den Bootstrap-Lademodus

Die folgenden Komponenten sind nur einfach vorhanden:

- PCI-Interface (PLX9050: je Kanal ein Interrupt-Eingang, ein Interrupt-Ausgang sowie ein Chip-Select-Signal zum DP-RAM und eine Reset-Leitung für beide Kanäle)
- optional sind 32 k NOV-RAM bestückbar, die über eine weitere Chip-Select-Leitung adressiert werden

Bedeutung der LEDs

Zustand der FC310x	Anzeige der LEDs
RESET, OFFLINE	rote LED an, grüne LEDs aus
STOP	grüne LED aus, rote LED blinkt (mit 10 Hz) während Upload der Bus-Konfiguration läuft: grüne und rote LED blinken (mit 10 Hz)
RUN	wenn zugehörige TwinCAT-Task gestartet wurde ist die grüne LED an, sonst blinkt die grüne LED (mit 1 Hz) wenn alle Boxen fehlerfrei sind ist die rote LED aus, sonst blinkt die rote LED (mit 1 Hz)

Zustände der FC310x

RESET, OFFLINE

Nach dem Power-On ist die FC310x im Zustand RESET, sie verlässt den Zustand RESET durch den Start von TwinCAT, nach dem Stopp von TwinCAT (oder einem schwerem Busfehler) geht sie in den Zustand OFFLINE. Im Zustand OFFLINE ist die FC310x nicht am Bus aktiv.

STOP

Nach dem Start von TwinCAT oder beim Auslesen der Bus-Konfiguration aus dem System Manager geht die FC310x in den Zustand STOP, in dem sie am Bus aktiv ist, aber noch keinen Datenaustausch (Data_Exchange) mit der DP-Slaves durchführt. Während des TwinCAT Starts oder auch während eines IO-Resets ist die FC310x im Zustand STOP, der Zustand STOP wird sowohl beim TwinCAT Start als auch beim IO-Reset automatisch verlassen und die FC310x geht in den Zustand RUN über.

RUN

Im Zustand RUN baut die FC310x automatisch die DP-Verbindungen mit allen projektierten DP-Slaves auf. Sobald die zugehörige TwinCAT-Task gestartet wurde, kommuniziert sie mit den DP-Slaves per Data_Exchange, solange die zugehörige TwinCAT-Task noch nicht gestartet wurde, fragt sie nur die Diagnose ab. Wenn die zugehörige Task gestoppt wird (z. B. PLC-STOP, Breakpoint in der PLC), geht die FC310x automatisch in den CLEAR-Mode (Ausgänge auf 0 bzw. es wird slavespezifisch reagiert, wenn dieser den FailSafe-Mode unterstützt). Wenn die zugehörige Task wieder läuft, geht die FC310x wieder automatisch in den OPERATE-Mode (alle Ausgänge auf den von TwinCAT gesetzten Werten).

3 FC310x als Master

3.1 Master

Als Master werden die Protokolle PROFIBUS DP, PROFIBUS DPV1, PROFIBUS DPV2, S5-FDL-AGAG-Kommunikation und das PROFIDRIVE-PKW-Interface unterstützt.

PROFIBUS DP

Es folgt eine Übersicht der PROFIBUS-DP-Master-Funktionen:

Funktion	Beschreibung
Standard DP	In dem Kapitel PROFIBUS DP [► 11] werden die notwendigen Schritte beschrieben, die zum Aufbau einer DP-Verbindung (Set_Prm - Parameter, Chk_Cfg - Konfiguration) und zum Austausch von Nutzdaten (Data_Exchange) notwendig sind.
Task-Synchronisierung	Im dem Kapitel Synchronisierung [► 12] wird beschrieben, wie die TwinCAT-Task mit dem PROFIBUS-Zyklus synchronisiert ist.
Slave-Prioritäten	Die Slaves können Telegramme mit unterschiedlichen Zykluszeiten bekommen, in dem Kapitel Slave-Priorisierung/mehrere DP-Zyklen [► 14] sind die notwendigen Einstellungen beschrieben.
mehrere DP-Zyklen	Um bei großen Task-Zyklen möglichst neue Inputs zu erhalten, können mehrere DP-Zyklen je Task-Zyklus durchgeführt werden, wie es in dem Kapitel Slave-Priorisierung/mehrere DP-Zyklen [► 14] beschrieben ist.
Diagnose	In diesem Kapitel sind die Diagnose [► 16] -Möglichkeiten beschrieben.
Fehlerreaktionen	Im Falle eines Fehlers (Slave fällt aus oder Task wird gestoppt) können unterschiedliche Fehlerreaktionen [► 17] eingestellt werden.
Sync/Freeze	In dem Kapitel Sync/Freeze [► 16] wird Aktivierung des Sync- bzw. Freeze-Kommandos beschrieben.
Upload Configuration	Die am PROFIBUS angeschlossenen Slaves können per Upload Configuration [► 35] ausgelesen werden.
Master-Redundanz	In dem Kapitel Master-Redundanz [► 39] sind die Einstellungen beschrieben, um einen zweiten Master mit der gleichen Konfiguration als StandBy-Master zu haben (ab TwinCAT 2.9).

PROFIBUS DPV1

Es folgt eine Übersicht der PROFIBUS-DPV1-Master-Funktionen:

Funktion	Beschreibung
MSAC_C1	Die MSAC_C1 [► 32] -Verbindung wird zusammen mit der zyklischen Verbindung aufgebaut, es werden die Dienste Read, Write und Data_Transport unterstützt.
MSAC_C2	Die MSAC_C2 [► 32] -Verbindung wird unabhängig von der zyklischen Verbindung aufgebaut und kann auch von einem zweiten Master benutzt werden (während der erste über die zyklische bzw. die MSAC_C1-Verbindung mit dem Slave kommuniziert). Es werden die Dienste Initiate, Abort, Read, Write und Data_Transport unterstützt.

PROFIBUS DPV2

Es folgt eine Übersicht der PROFIBUS-DPV2-Master-Funktionen:

Funktion	Beschreibung
Äquidistanz	Die DPV2-Äquidistanz-Funktionalität ist in dem Kapitel PROFIBUS MC [► 26] beschrieben.

S5-FDL-AGAG-Kommunikation

Die S5-FDL-AGAG-Kommunikation ist in dem Kapitel [S5-FDL \[► 37\]](#) beschrieben.

PROFIDRIVE-PKW-Interface

Das PROFIDRIVE-PKW-Protokoll [\[► 36\]](#) ist im PROFIBUS-Master implementiert und kann über azyklische ADS-Aufrufe genutzt werden.

3.2 PROFIBUS DP

Standard DP-Betrieb

Um den Standard-DP-Betrieb zu konfigurieren, ist im TwinCAT System Manager wie folgt vorzugehen:

DP-Master projektieren

Zuerst ist ein E/A-Gerät "PROFIBUS Master FC310x, PCI" zu projektieren (mit rechter Maustaste auf "E/A-Geräte", dann "Gerät anfügen" auswählen). Der entsprechende Kanal ist auf dem Karteireiter "FC310x" (s. Karteireiter FC310x unter [TwinCAT 2.8 \[► 41\]](#) bzw. [TwinCAT 2.9 \[► 49\]](#)) und zu suchen ("Search"-Button) und ggf. die Baudrate anzupassen (ist standardmäßig auf 12 Mbit/s eingestellt).

DP-Slaves anfügen

Die Beckhoff-Slaves oder Fremd-Geräte sind zu projektieren. (Es werden alle Slaves automatisch angezeigt (nach Herstellern sortiert), deren GSD-Datei im Unterverzeichnis PROFIBUS des System Managers gespeichert sind, um andere GSD-Dateien einzubinden, ist unter Verschiedenes die "Allgemeine PROFIBUS Box (GSD)" anzuwählen.

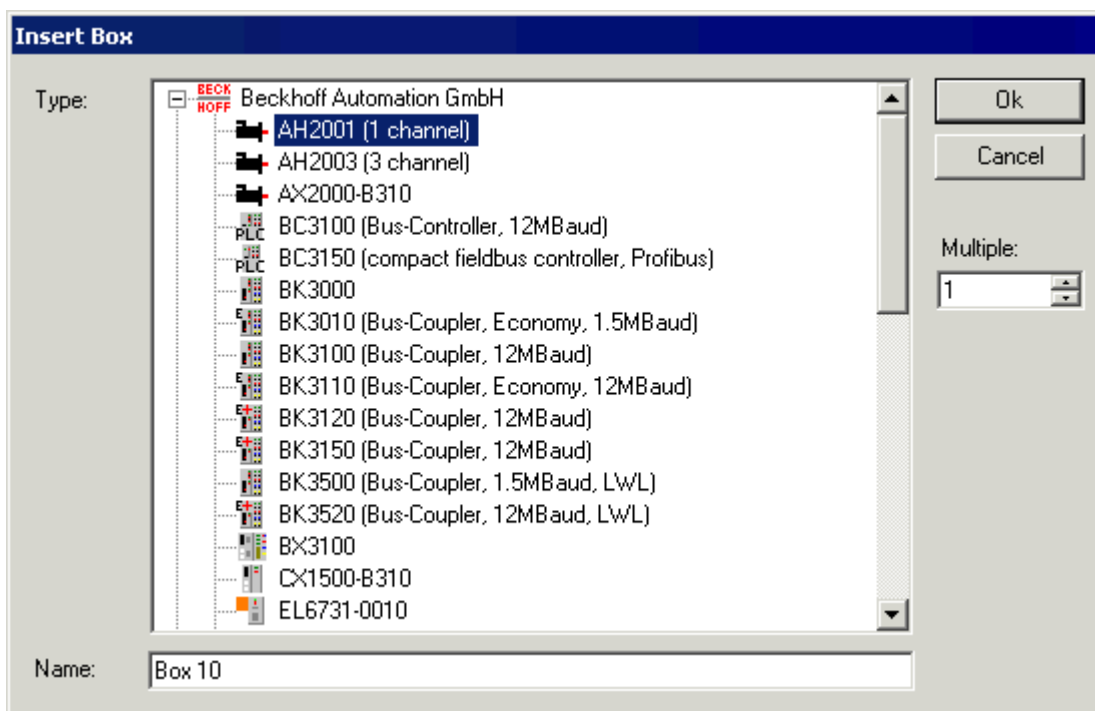


Abb. 1: Anfügen eines DP-Slaves

Bei modularen Slaves sind anschließend noch die Klemmen/IL-Module (Beckhoff-Slaves) bzw. die DP-Module (Fremd-Geräte) anzufügen.

System-Start

TwinCAT Config-Mode (ab TwinCAT 2.9)

Für den TwinCAT Config-Mode reicht das schon aus, um Daten mit den projektierten Slaves auszutauschen. Der TwinCAT Config-Mode ist dazu zu starten und mit dem "Reload Devices"-Button in der Toolbar die Konfiguration des DP-Masters zu aktivieren. Danach können die Daten der projektierten Slaves aus dem System Manager auf dem jeweils zugehörigen Variablen-Karteireiter gelesen und beschrieben werden.

TwinCAT Run-Mode

Für den TwinCAT Run-Mode muss noch mindestens eine Variable des PROFIBUS-Masters oder der projektierten Slaves mit einer Task verknüpft werden. Dann ist das Projekt in der Registry zu speichern und das TwinCAT System im Run-Mode zu starten. Der Datenaustausch wird mit den Slaves erst durchgeführt, wenn die zugehörige Task gestartet wurde (wenn mehrere Tasks mit dem PROFIBUS-Master (oder den projektierten Slaves) verknüpft sind, muss die höchstpriorie der verknüpften Tasks gestartet sein, damit Datenaustausch mit den Slaves durchgeführt wird).

Busparameter

TwinCAT 2.8: Die PROFIBUS DP-Busparameter befinden sich auf dem Karteireiter PROFIBUS [▶ 43] des Devices und sollten nur von sachkundigen Benutzern geändert werden.

TwinCAT 2.9: Die PROFIBUS DP-Busparameter befinden sich im Dialog Bus-Parameter [▶ 50] der über den Karteireiter FC310x [▶ 49] (Button Bus-Parameter (DP)) angewählt wird und sollten nur von sachkundigen Benutzern geändert werden.

3.3 Synchronisierung

3.3.1 Übersicht

Im TwinCAT RunMode ist der DP-Master immer mit der höchstpriorien Task synchronisiert, mit der Variablen verknüpft sind. Die Zykluszeit der entsprechenden Task wird in der **Cycle-Time** auf dem Karteireiter "**FC310x**" (für TwinCAT 2.8 [▶ 41] bzw. TwinCAT 2.9 [▶ 49]) des Masters angezeigt, sobald einmal das Mapping erzeugt wurde. Bei der Task kann eingestellt werden, ob das "I/O am Task-Anfang" aktualisiert werden soll oder nicht.

I/O am Task-Anfang

Wenn bei der Task die Einstellung "I/O am Task-Anfang" (Check-Box) angewählt ist (Default-Einstellung bei NC-Task), wird vor dem Start der Task geprüft, ob der vorangegangene DP-Zyklus beendet wurde, dann die Inputs und Outputs (die Outputs sind dann vom letzten Task-Zyklus) kopiert und der DP-Zyklus gestartet (im Beispiel Task-Zykluszeit 2 ms, Echtzeit-Ressourcen 80 %):

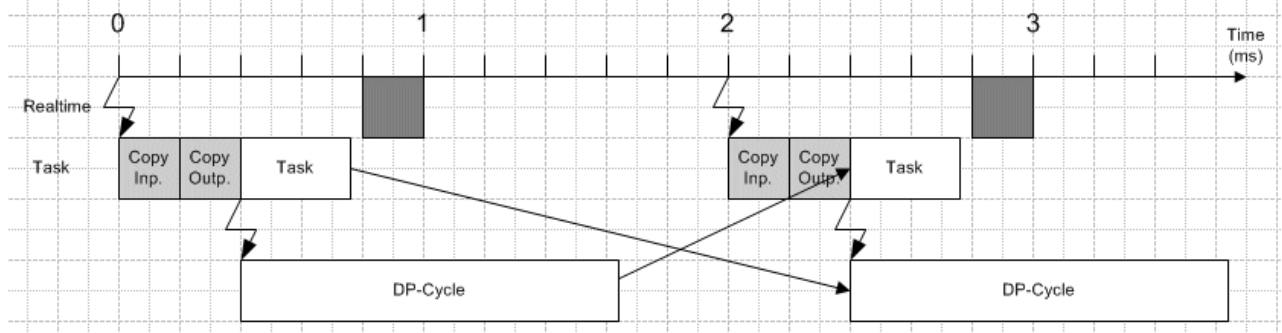


Abb. 2: I/O am Task-Anfang und keine Überschreitung der Echtzeit-Ressourcen

Wenn in dem vorliegenden Fall das Kopieren der Inputs und Outputs sowie die Task-Rechenzeit 0,8 ms überschreiten, würde die Ausführung der Task durch NT unterbrochen werden, da die 80 % Echtzeit-Ressourcen erreicht sind:

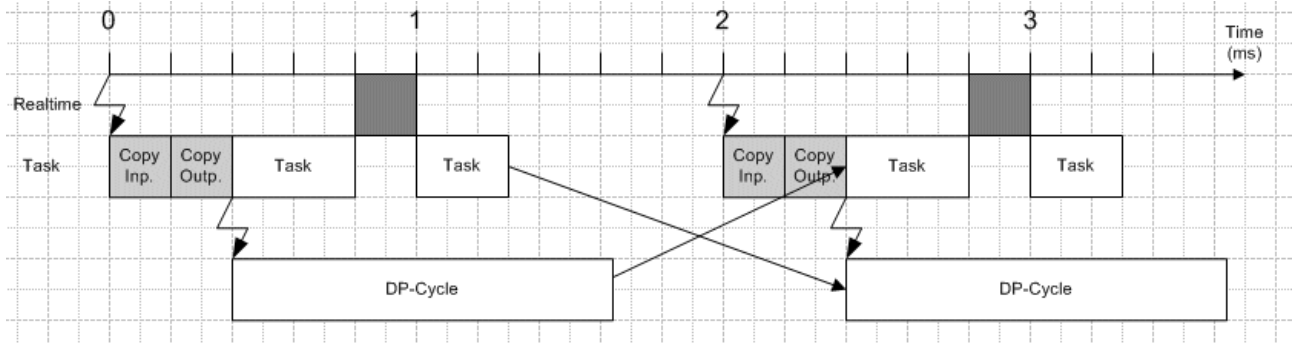


Abb. 3: I/O am Task-Anfang und Überschreitung der Echtzeit-Ressourcen

Dieser Fall wäre auch noch unproblematisch, da der DP-Zyklus rechtzeitig fertig ist. Wenn die Einstellung "IO am Task-Anfang" nicht angewählt ist, ist der Ablauf etwas kritischer, wie im Folgenden beschrieben ist.

I/O nicht am Task-Anfang

Wenn bei der Task die Einstellung "I/O am Task-Anfang" (Check-Box) nicht angewählt ist (Default-Einstellung bei SPS-Task), wird vor dem Start der Task geprüft, ob der vorangegangene DP-Zyklus beendet wurde, und die Inputs werden kopiert. Danach erfolgt die Taskbearbeitung und am Ende der Task werden die Outputs kopiert und der DP-Zyklus gestartet (im Beispiel Task-Zykluszeit 2 ms, Echtzeit-Ressourcen 80 %):

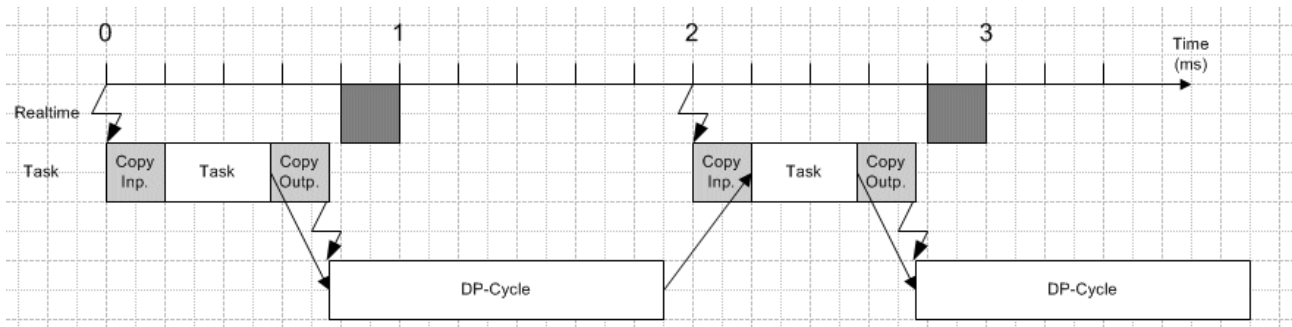


Abb. 4: I/O nicht am Task-Anfang und keine Überschreitung der Echtzeit-Ressourcen

Da sich im Fall "I/O nicht am Task-Anfang" die Task und der PROFIBUS die Bandbreite teilen müssen, wirkt sich eine Überschreitung der Echtzeit-Ressourcen wesentlich stärker aus als beim Fall "I/O am Task-Anfang":

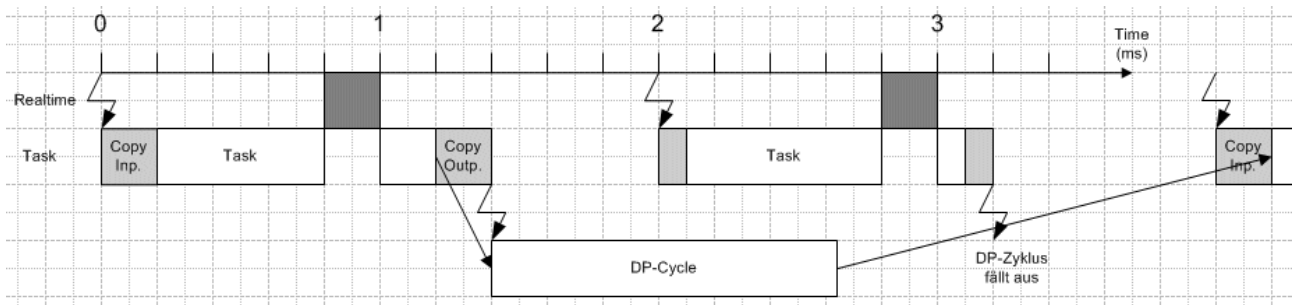


Abb. 5: I/O nicht am Task-Anfang und Überschreitung der Echtzeit-Ressourcen

In dem beschriebenen Fall fängt der DP-Zyklus später an und ist nicht mehr rechtzeitig (vor dem Start des folgenden Task-Zyklus) fertig. Das hat zur Folge, dass vor dem Ausführen der Task festgestellt wird, dass der letzte DP-Zyklus noch nicht beendet war, woraufhin vor dem Starten der Task keine Inputs kopiert werden (die Task also mit den alten Inputs rechnet) und nach der Taskbearbeitung auch keine Outputs

kopiert sowie der DP-Zyklus nicht neu gestartet wird, es wird also ein DP-Zyklus ausgelassen. Das Ausfallen eines DP-Zyklus kann mit dem **CycleCounter** erkannt werden, der im Kapitel [Master-Diagnose](#) [► 20] beschrieben ist.

Gegenüberstellung I/O am Task-Anfang und I/O nicht am Task-Anfang

Bei der Einstellung "I/O am Task-Anfang" hat man den Vorteil, dass Task und DP-Zyklus sich die zur Verfügung stehende Bandbreite nicht teilen müssen und dass das Starten des DP-Zyklus sehr konstant erfolgt (Jitter entspricht dem TwinCAT-Jitter), während bei der Einstellung "I/O nicht am Task-Anfang" es leichter passieren kann, dass ein DP-Zyklus ausgelassen wird und dass die zeitliche Konstanz des DP-Zyklus-Starts zusätzlich von dem Jitter der Taskarbeitung abhängt. Der Nachteil der Einstellung "I/O am Task-Anfang" ist, dass die Totzeit (System-Reaktionszeit) größer wird.

3.3.2 Slave-Priorisierung/mehrere DP-Zyklen

Aufteilung der DP-Slaves auf mehrere DP-Zyklen (Priorisierung der Slaves)

Um in Systemen, bei denen einige Slaves sehr schnell gepollt, bei anderen aber eine größere Zykluszeit ausreichend wäre, eine möglichst geringere DP-Zykluszeit zu bekommen, können die Slaves priorisiert werden. Dabei kann je Slave angegeben werden, in jedem wievielten Zyklus (**Divider** unter **Data-Exch Poll-Rate** auf dem Karteireiter [Features](#) [► 60] des Slaves) er gepollt werden soll. Weiterhin ist es dann sinnvoll, dass z. B. in dem nachfolgend dargestellten Fall bei 4 Slaves, die jeweils nur in jedem zweiten Zyklus angesprochen werden sollen, auch einstellbar ist, dass jeweils 2 Slaves in dem einen Zyklus und 2 Slaves in dem anderen Zyklus gepollt werden, um die Gesamt-DP-Zyklus möglichst konstant zu halten. Dazu gibt es auf dem Karteireiter [Features](#) [► 60] des Slaves die Einstellung des **Modulo** unter **Data-Exch Poll-Rate**, wobei in dem dargestellten Fall Slave 3 und 5 den **Modulo** 0 und Slave 4 und 6 den **Modulo** 1 erhalten würden. Der aktuelle Modulo kann in der Variablen **ActualModulo** gelesen werden, die im Kapitel [Master-Diagnose](#) [► 20] beschrieben ist.

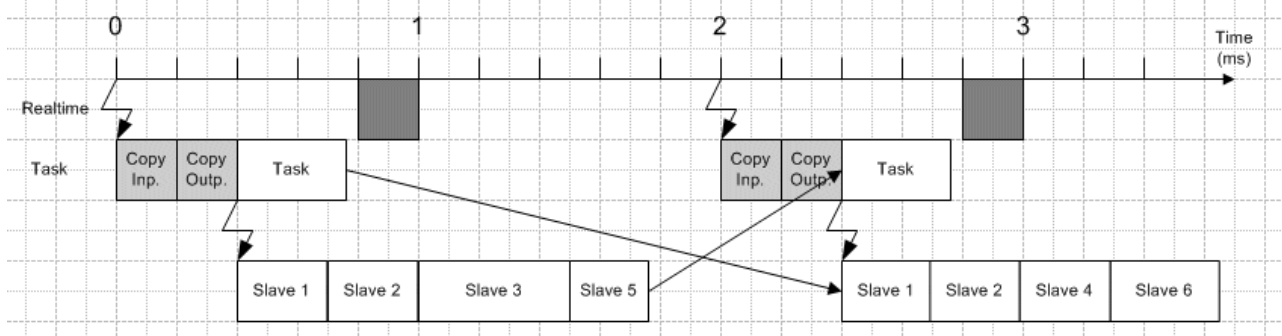


Abb. 6: Aufteilung der DP-Slaves auf mehrere DP-Zyklen

Mehrere DP-Zyklen in einem Task-Zyklus

Bei Task-Zykluszeiten, die mehr als doppelt so groß wie die DP-Zykluszeit sind, ist es möglich, mehrere DP-Zyklen innerhalb eines Task-Zyklus durchzuführen, um möglichst aktuelle Inputdaten zu bekommen. Dabei wird entsprechend des auf dem Karteireiter **"FC310x"** (für [TwinCAT 2.8](#) [► 41] bzw. [TwinCAT 2.9](#) [► 49]) des Masters einstellbaren Faktors **AnzahlDP-Zyklen je Task-Zyklus** mit dem Starten des ersten DP-Zyklus ein Timer mit der Zykluszeit (Task-Zykluszeit/(Anzahl DP-Zyklen je Task-Zyklus)) gestartet, über den dann weitere DP-Zyklen gestartet werden. Dabei ist allerdings aufzupassen, dass der letzte DP-Zyklus rechtzeitig (vor dem nächsten Start der Task) beendet ist, da es sonst zu einem (bzw. mehreren, in Abhängigkeit vom Verhältnis Anzahl DP-Zyklen je Task-Zyklus) DP-Zyklusausfall kommt, wie im Kapitel [Synchronisierung](#) [► 12] beschrieben wurde.

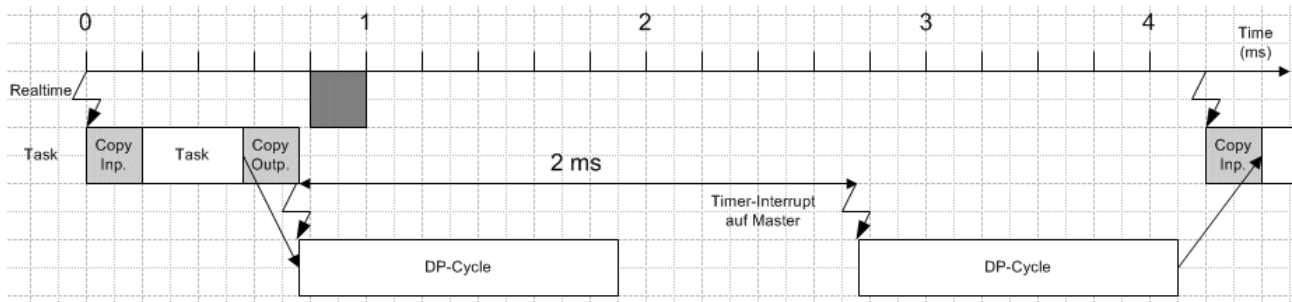


Abb. 7: Mehrere DP-Zyklen in einem Task-Zyklus

Mehrere Data-Samples innerhalb eines Task-Zyklus

Die beiden beschriebenen Funktionalitäten lassen sich jetzt kombinieren, um z. B. einem oder mehreren Slaves in einem 2 ms-Zyklus jede ms neue Daten zu geben bzw. von dem Slave neue Daten zu bekommen, um eine bessere Regelqualität zu erhalten. Für diesen Fall werden die Einstellungen unter **Additional Data_Exchange Samples** auf dem Karteireiter Features [▶ 60] des Slaves statt unter **Data-Exch Poll-Rate** (wie oben beschrieben) vorgenommen.

In dem unten dargestellten Beispiel ist zunächst der Faktor **AnzahlDP-Zyklen je Task-Zyklus** auf dem Karteireiter **"FC310x"** (für TwinCAT 2.8 [▶ 41] bzw. TwinCAT 2.9 [▶ 49]) des Masters auf 2 einzustellen. Damit die Task 2 verschiedene Werte an den Slave senden bzw. 2 verschiedene Werte von dem Slave empfangen kann, ist der entsprechende Slave zweimal im System Manager einzutragen, wobei alle Einstellungen, mit Ausnahme des **Modulo** unter **Additional Data_Exchange Samples** auf dem Karteireiter Features [▶ 60] des Slaves, gleich sein müssen. Dort wäre für den entsprechenden Slave eine 0 bei der einen und eine 1 bei der anderen Box im System Manager anzugeben. Der **Multiplifier** unter **Additional Data_Exchange Samples** ist für diesen Slave bei beiden Boxen auf 2 zu stellen, d.h., jede der beiden eingetragenen Boxen wird nur in jedem 2. DP-Zyklus gepollt (der Slave wird dann in jedem DP-Zyklus gepollt, aber er ist ja zweimal eingetragen). Für alle anderen Slaves, die nur einmal innerhalb des Task-Zyklus gepollt werden sollen (die sind natürlich auch nur einmal im System Manager eingetragen), ist der **Multiplifier** unter **Additional Data_Exchange Samples** ebenfalls auf 2, mit dem **Modulo** unter **Additional Data_Exchange Samples** können diese Slaves jetzt noch auf die beiden Zyklen aufgeteilt werden. Ein Slave, der zweimal gepollt, aber nur ein Variablenabbild in der Task haben soll, ist nur einmal einzufügen, der **Multiplifier** wäre auf 1, der **Modulo** auf 0 eingestellt.

Im vorliegenden Beispiel wären die Slaves 1 und 2 jeweils zweimal im System Manager eingetragen mit den Einstellungen:

- Additional Data_Exchange Samples/Multiplier = 2
- Additional Data_Exchange Samples/Modulo = 0 bzw. 1

Die Slaves 3 und 5 wären nur einmal im System Manager eingetragen und hätten die Einstellungen:

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

Die Slaves 4 und 6 wären ebenfalls nur einmal im System Manager eingetragen mit den Einstellungen:

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

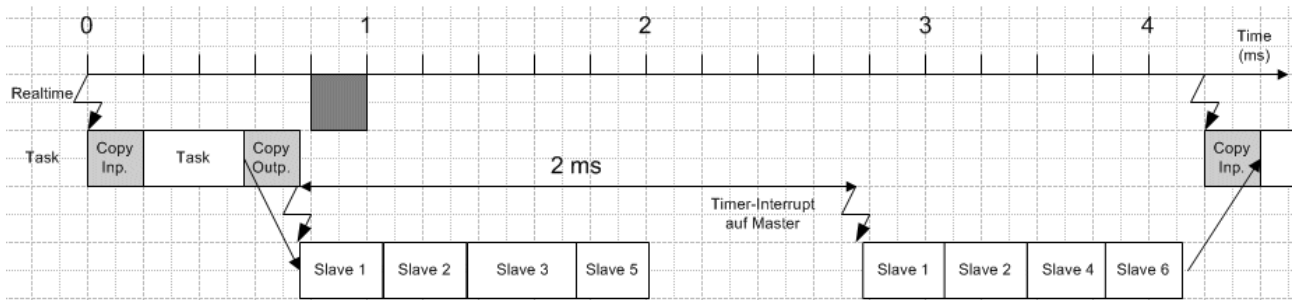


Abb. 8: Mehrere Data-Samples innerhalb eines Task-Zyklus

Für die Slaves 1 und 2 liegen die Variablen zweimal vor (jeweils 2 Boxen im System Manager), die Variablen der Box mit dem Additional Data_Exchange Samples/Modulo 0 werden zuerst gesendet bzw. empfangen.

3.3.3 Sync/Freeze-Funktionalität

Mit Sync will man bei mehreren Slaves gleichzeitig Ausgänge ausgeben, mit Freeze will man bei mehreren Slaves gleichzeitig Eingänge einlesen.

Der Ablauf in TwinCAT mit FC310x und Buskopplern (im K-Bus-Synchronen Mode) wäre also der folgende (siehe Kapitel [Synchronisierung](#) [► 12]):

- Am Anfang (I/O am Taskanfang) bzw. Ende (I/O nicht am Taskanfang) des Task-Zyklus werden die Ausgänge geschrieben
- Dadurch wird der PROFIBUS-Zyklus gestartet
- Am Anfang des PROFIBUS-Zyklus wird ein Sync/Freeze-Telegramm gesendet
- Dadurch starten die Buskoppler einen K-Bus-Zyklus mit den Ausgängen aus dem letzten Task-Zyklus und übergeben die Eingänge aus dem letzten K-Bus-Zyklus
- Dann sendet der Master zu jedem Slave die aktuellen Ausgänge und holt die übergebenen Eingänge
- Am Anfang des nächsten Task-Zyklus werden die Eingänge gelesen
- usw.

Ausgänge und Eingänge sind also immer einen Zyklus alt.

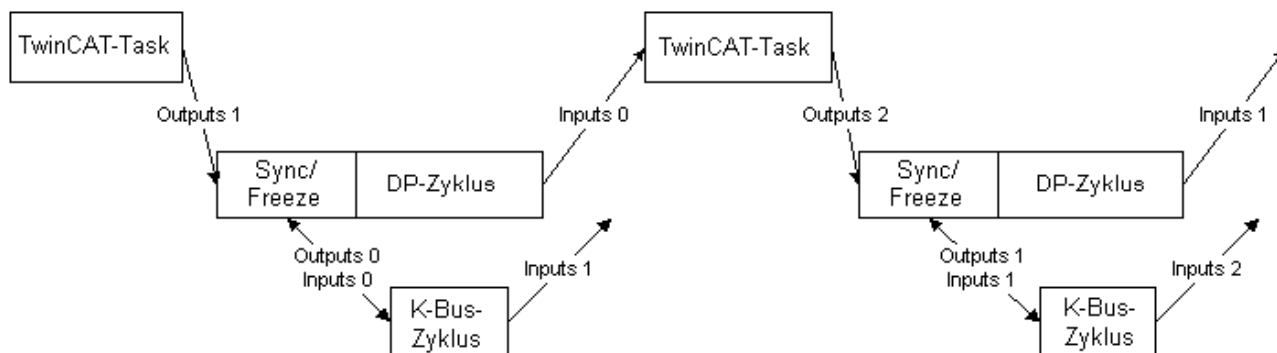


Abb. 9: Sync/Freeze-Funktionalität

An der FC310x ist der **Operation-Mode** auf dem Karteireiter "**FC310x**" (für [TwinCAT 2.8](#) [► 41] bzw. [TwinCAT 2.9](#) [► 49]) des Masters auf "DP/MC (Equidistant)" zu stellen, an den Boxen, die per Sync/Freeze betrieben werden sollen, ist auf dem Karteireiter [Profibus](#) [► 59] des Slaves das Flag **Sync/Freeze enable** anzuklicken. Der Master verwendet für die Synchronisierung per Sync/Freeze immer die Gruppe 1.

3.4 Diagnose

3.4.1 Übersicht

Im Abschnitt [Fehlerreaktionen](#) [► 17] ist beschrieben, wie auf nicht oder falsch antwortende Slaves, PLC-Stopp oder beim Start-Up reagiert werden soll. Das Kapitel [Slave-Diagnose](#) [► 24] erklärt, wie vom Slave gemeldete Diagnosedaten und Slave-Statistiken gelesen werden können, während im Kapitel [Master-Diagnose](#) [► 20] allgemeine Diagnoseinformationen und Statistiken beschrieben werden.

3.4.2 Fehlerreaktionen

Ausfall eines Slaves

Wenn ein Slave nicht oder gestört antwortet, wiederholt der Master das Telegramm entsprechend des **Max Retry-Limit** (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Bus-Parameter [▶ 50]) mehrere Male. Beim Empfang eines gestörten Telegramms wiederholt der Master sofort, im Timeout-Fall hat der Master die **Slot-Time** (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Bus-Parameter [▶ 50]) auf eine Antwort des Slaves gewartet. Bei 12 Mbit/s, einer Slot-Time von 1000 Bit-Zeiten und einem Max Retry-Limit von 4 (Defaultwerte) bei einem Data_Exchange-Telegramm verzögert sich das Senden des folgenden Telegramms um den Wert

$$TDelay = (4 \times ((15 + \text{Anzahl Outputs}) \times 11 + 1000) - (15 + \text{Anzahl Inputs}) \times 11) / 12 \mu\text{s}$$

Der DpState [▶ 26] des Slaves wird auf 0x02 (Timeout) bzw. 0x0B (gestörtes Telegramm) gesetzt. Die Auswirkung auf die DP-Verbindung kann eingestellt werden (s.u.).

Normaler DP-Zyklus (12 Mbit/s, 5 Slaves, im Schnitt 20 Bytes I, 20 Bytes O je Slaves)

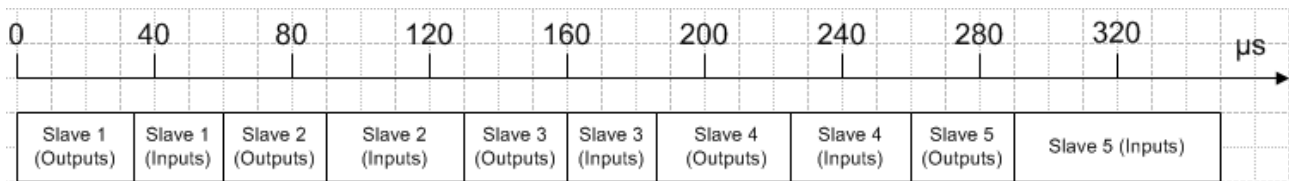


Abb. 10: Normaler DP-Zyklus

erstmalig gestörter DP-Zyklus (Slave 3 antwortet nicht)

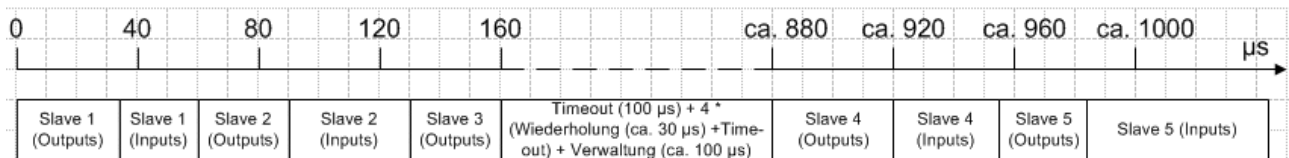


Abb. 11: Erstmalig gestörter DP-Zyklus

folgende DP-Zyklen (Slave 3 nicht mehr in der Poll-Liste)

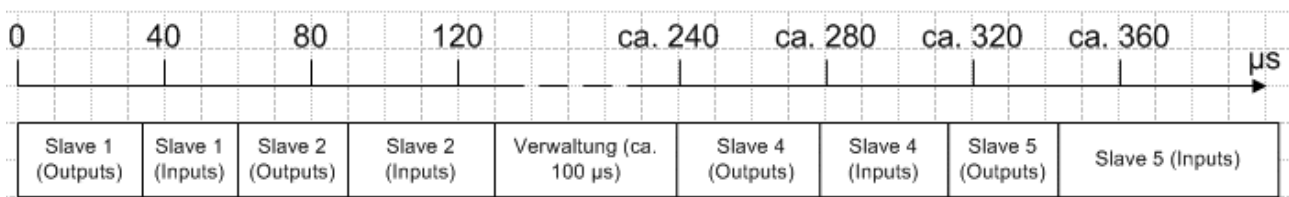


Abb. 12: Folgende DP-Zyklen

Weiterhin kann es noch passieren, dass der Slave falsch antwortet (z. B. weil aufgrund eines lokalen Ereignisses auf dem Slave die DP-Verbindung abgebaut wurde). In diesem Fall wird das Telegramm nicht wiederholt, sondern mit dem Senden des folgenden Telegramms fortgefahren. Der DpState [▶ 26] wird auf einen Wert ungleich 0 gesetzt, der Slave wird aus der Poll-Liste ausgetragen und im nächsten DP-Zyklus nicht mehr angesprochen (d.h. der Sendezeitpunkt des nachfolgenden Telegramms verändert sich), bis die DP-Verbindung wieder erneut aufgebaut wurde.

Reaktionen im Master

Die Reaktion im Master kann je Slave eingestellt werden (s. Karteireiter Features [▶ 60] des Slaves).

Auswirkung auf die DP-Verbindung (NoAnswer-Reaction), falls Slave nicht oder nicht ungestört antwortet

Hier wird festgelegt, ob der die DP-Verbindung zu dem Slave sofort oder erst nach Ablauf der DP-Watchdog-Zeit (s. Karteireiter PROFIBUS [▶ 59] des Slaves) ohne korrektes Empfangstelegramm abgebaut werden soll.

1. Wenn die DP-Verbindung sofort abgebaut werden soll (Leave Data-Exch, Default-Einstellung), wird der Slave aus der Poll-Liste ausgetragen und im nächsten DP-Zyklus nicht mehr angesprochen, bis die DP-Verbindung wieder erneut aufgebaut wurde. Um die DP-Verbindung zu dem Slave wieder aufzubauen werden mindestens 7 Telegramme gesendet, der Vorgang dauert in der Regel mindestens 10 bis 20 ms.
2. Wenn die DP-Verbindung erst abgebaut werden soll, wenn innerhalb der DP-Watchdog-Zeit der Slave nicht oder ungestört geantwortet hat (Stay in Data-Exch (for WD-Time)), wird im nächsten Pollzyklus erneut versucht, den Slave anzusprechen, falls der Slave aber nicht antwortet, wird keine Wiederholung gesendet.

Die Einstellung "Stay in Data-Exch (for WD-Time)" (2.) macht dann Sinn, wenn der PROFIBUS-Zyklus auch beim Ausfall eines Slaves zeitlich möglichst konstant ablaufen soll und der Ausfall eines Slave für einen oder mehrere Zyklen toleriert werden kann (z. B. in der Betriebsart DP/MC (Equidistant) [▶ 26]). In diesem Fall sollte die DP-Watchdog-Zeit des Slaves entsprechend der noch tolerierbaren Ausfall-Zeit des Slaves eingestellt und das **Max Retry-Limit (DX)** (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Bus-Parameter [▶ 50]) auf 0 gesetzt werden.

Normaler DP-Zyklus (12 Mbit/s, 5 Slaves, im Schnitt 20 Bytes I, 20 Bytes O je Slaves) im Mode "Stay in Data-Exch (for WD-Time)"

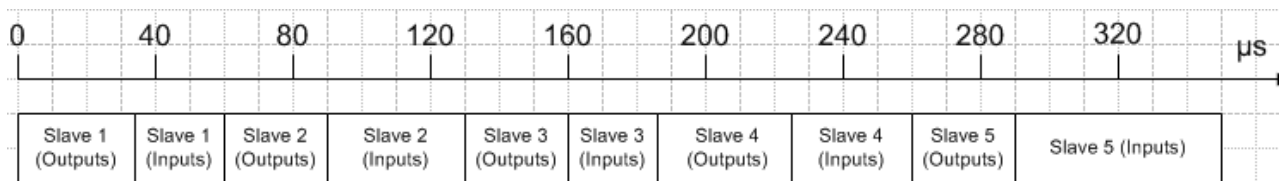


Abb. 13: Normaler DP-Zyklus bei Stay in Data-Exch (for WD-Time)

erster gestörter und folgende DP-Zyklen im Mode "Stay in Data-Exch (for WD-Time)" (Slave 3 antwortet nicht)

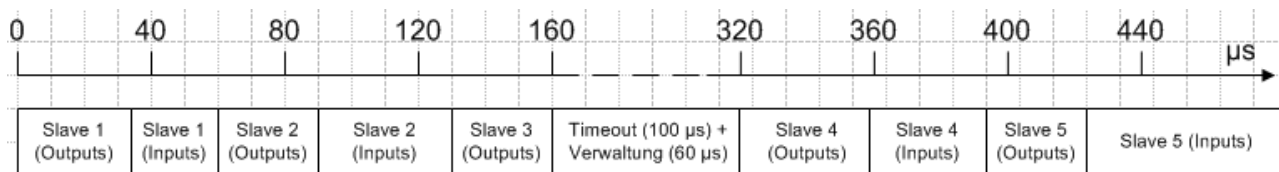


Abb. 14: Erster gestörter und folgende DP-Zyklen bei Stay in Data-Exch (for WD-Time)

Auswirkung auf die Input-Daten des Slaves (Changes of the Input Data), falls Slave nicht richtig antwortet

Hier wird festgelegt, ob die Input-Daten des Slaves bei dessen Ausfall auf 0 gesetzt ("Inputs will be set to 0", Default-Einstellung) oder der alte Wert erhalten bleiben soll ("No changes"). In beiden Fällen wird auf jeden Fall der DpState [▶ 26] des Slaves auf einen Wert ungleich 0 gesetzt, so dass die Task immer erkennen kann, ob die Daten gültig sind oder nicht. Wenn ein Slave gestört antwortet, werden die Inputdaten, unabhängig von der Einstellung **Changes of the Input Data** immer auf 0 gesetzt.

Einstellung des Wiederanlauf-Verhaltens des Slaves (Restart Behaviour of the Slave), falls die DP-Verbindung zu dem Slave abgebaut wurde

Hier wird festgelegt, ob die DP-Verbindung zu einem Slave, dessen DP-Verbindung abgebaut wurde, automatisch wieder aufgebaut wird, oder ob dieses manuell durch einen ADS-WriteControl-Aufruf (s. ADS-Interface [▶ 29]) passieren soll.

Auswirkung auf den Zustand des Masters (Reaction of the Master), falls die DP-Verbindung zu dem Slave abgebaut wurde

Hier wird festgelegt, ob der Abbau der DP-Verbindung zu einem Slave ohne weitere Auswirkungen bleiben (No Reaction, Default-Einstellung) oder ob der Master den STOP-Zustand einnehmen und damit die DP-Verbindungen zu allen Slaves abbauen soll.

Auswirkung auf den Zustand des Masters (Clear Mode), falls die DP-Verbindung zu dem Slave abgebaut wurde

Mit dem **Clear Mode** (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Fault-Settings [▶ 52]) kann eingestellt werden, dass der Master in den Zustand "Clear" wechselt bzw. bleibt, solange mindestens ein MC-Slave (Einstellung "Only MC-Slaves") bzw. irgendein Slave (Einstellung "All Slaves") nicht korrekt antwortet (einen `DpState` [▶ 26] ungleich 0 hat).

Die Einstellung **Reaction of the Master** (s. Karteireiter Features [▶ 60] des Slaves), die im vorangegangenen Kapitel beschrieben wurde, hat gegenüber dem **Auto-Clear Mode** Vorrang, d.h. wenn ein entsprechend eingestellter Slave ausfällt, geht der Master in den Zustand STOP.

Ausfall des Masters

Überwachung in der SPS/IO-Task

Bei dauerhaften Busstörungen kann sich der DP-Zyklus auch bei 12 Mbit/s auf bis zu 100 ms verlängern. Um den DP-Master zu überwachen, gibt es eine Status-Variable `CycleCounter`, die in der SPS verknüpft werden kann (s. Kapitel Master-Diagnose [▶ 20]). Diese Variable wird vom Master nach jedem DP-Zyklus inkrementiert, durch Überwachen dieser Variable in der SPS kann also ein Ausfall des Masters festgestellt werden.

Überwachung im Slave

Um den Ausfall des Masters bzw. die Übertragung auf dem PROFIBUS zu überwachen, kann beim Slave ein **Watchdog** (s. Karteireiter PROFIBUS [▶ 59] der Box) aktiviert werden (Default-Einstellung: Watchdog aktiviert mit 200 ms). Der **Watchdog** muss mindestens zweimal so groß wie das Maximum aus **Estimated Cycle Time** und **Cycle Time** (s. Karteireiter "FC310x" (für TwinCAT 2.8 [▶ 41] bzw. TwinCAT 2.9 [▶ 49]) des Masters) eingestellt werden.

Ausfall der SPS/IO-Task

Es wird zwischen den Fällen SPS-Stop, Erreichen eines Breakpoints und Task-Stop (IO-Task, NC-Task wird nur beim System-Stop gestoppt) unterschieden. Beim SPS-Stopp werden die Ausgangsdaten von der SPS noch auf 0 gesetzt, beim Erreichen eines Breakpoints bleiben die Daten zunächst unverändert.

Im Master wird mit einer Überwachungszeit (TwinCAT 2.8: entsprechend der Einstellung **Clear-Delay** x Task-Zykluszeit auf dem Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: entsprechend der Einstellung **Task-Watchdog** x Task-Zykluszeit auf dem Dialog Fault-Settings [▶ 52]) die Task überwacht. Wenn innerhalb dieser Überwachungszeit keine neue Datenübergabe erfolgt, geht der Master entsprechend der Einstellung **Reaction on PLC-Stop** bzw. **Reaction on Task-Stop** (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Fault-Settings [▶ 52]) in den Zustand "Clear" (Ausgänge werden auf 0 bzw. in den sicheren Zustand (Fail_Safe = 1 in der GSD-Datei) gesetzt, Default-Einstellung) oder bleibt im Zustand "Operate" (Ausgänge behalten den letzten Wert). Die Einstellung "Operate" macht Sinn, wenn beim Erreichen eines Breakpoints in der SPS die Ausgänge nicht abfallen sollen, bei einem SPS-Stopp würden die Ausgänge trotzdem auf 0 gesetzt (durch die SPS), auch wenn der Master in "Operate" bleibt. Dabei ist allerdings zu beachten, dass die Ausgänge nur genullt werden, wenn der vorangegangene DP-Zyklus rechtzeitig fertig war (s. Kapitel Synchronisierung [▶ 12]), es sollte daher nur während der Inbetriebnahmephase eingestellt werden.

Ausfall des Hosts

Um einen Absturz des Hosts (z. B. Blue Screen bei PC) zu überwachen, kann eine **Watchdog-Time** (TwinCAT 2.8: s. Karteireiter [FC310x \[► 41\]](#) des Masters, TwinCAT 2.9: s. Dialog [Fault-Settings \[► 52\]](#)) eingestellt werden. Wenn diese Watchdog-Time abläuft, geht der Master in den Zustand OFFLINE, d.h. die DP-Verbindungen zu allen Slaves werden abgebaut und der Master meldet sich vom PROFIBUS ab, d.h. er führt keine Buszugriffe mehr durch.

Start-Up Verhalten

Beim Start des TwinCAT Systems werden die DP-Verbindungen zu allen Slaves aufgebaut. Solange die höchstpriorige zugehörige Task noch nicht gestartet wurde, sendet der Master auch nach dem Aufbau der DP-Verbindung noch keine Data_Exchange-Telegramme sondern Diagnose-Telegramme. Sobald die höchstpriorige zugehörige Task einmal Daten übergeben hat und die DP-Verbindung des entsprechenden DP-Slaves aufgebaut ist, sendet der Master zyklisch (mit der höchstpriorigen zugeordneten Task) je ein Data_Exchange-Telegramm zu den entsprechenden Slaves.

Darüber hinaus kann über die Einstellungen **Operate-Delay** und **Clear Mode** (TwinCAT 2.8: s. Karteireiter [PROFIBUS \[► 43\]](#) des Masters, TwinCAT 2.9: s. Dialog [Fault-Settings \[► 52\]](#)) festgelegt werden, wann der Master vom "Clear"-Zustand (Ausgänge werden auf 0 bzw. in den sicheren Zustand (Fail_Safe = 1 in der GSD-Datei) gesetzt) in den "Operate"-Zustand (Ausgänge entsprechen den von der Task übergebenen Ausgängen) wechselt. Das **Operate-Delay** gibt an, wie lange der Master nach dem erstmaligen Übergeben der Daten mindestens noch im Zustand "Clear" bleiben soll. Mit dem **Clear Mode** wird, wie weiter oben beschrieben, eingestellt, ob der Master beim Ausfall eines allgemeinen bzw. eines MC-Slaves in den Zustand "Clear" wechselt bzw. bleibt.

Shut-Down Verhalten

Beim Stopp des TwinCAT Systems wird genauso reagiert, wie weiter oben im Kapitel "Ausfall des Hosts" beschrieben, die DP-Verbindungen aller Slaves werden abgebaut und der Master meldet sich vom Bus ab.

3.4.3 FC310x - Master-Diagnose

Diagnose Eingänge

Der Master verfügt über verschiedene Diagnosevariablen, die den Zustand der Karte sowie des Profibusses beschreiben und die in der SPS verknüpft werden können:

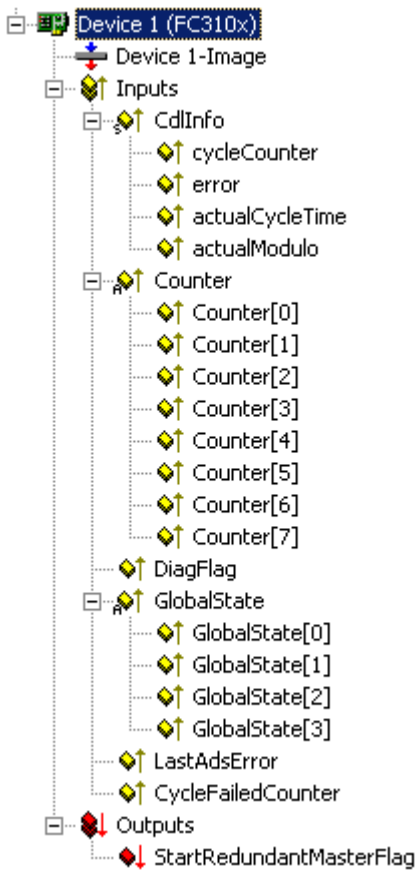


Abb. 15: Diagnosevariablen des PROFIBUS-masters

CdlInfo:

CdlInfo.error: Zeigt die Anzahl der Slaves an, mit denen kein Datenaustausch im letzten Zyklus durchgeführt werden konnte, nur wenn dieser Wert ungleich 0 ist, muss der BoxState der Slaves überprüft werden

CdlInfo.cycleCounter: wird am Ende jedes PROFIBUS-Zyklus inkrementiert, so dass man mit dieser Variable feststellen kann, ob der letzte Zyklus beendet war, bevor die Task gestartet wurde

CdlInfo.actualCycleTime: zeigt die aktuelle Zykluszeit in $4/25 \mu s$ an, diese Variable wird nur aktualisiert, falls alle Slaves im Datenaustausch sind (also CdlInfo.error gleich 0 ist)

CdlInfo.actualModulo: zeigt das aktuelle Modulo an, diese Variable spielt nur eine Rolle, wenn die Slaves priorisiert sind (s. Kapitel [Slave-Priorisierung/mehrere DP-Zyklen](#) [14])

Counter: werden für den [Redundanz-Mode](#) [39] benutzt

DiagFlag: Zeigt an, ob sich die Master-Diagnoseinformationen der Karte geändert haben, die dann per [ADS](#) [29] vom Steuerungsprogramm ausgelesen werden, woraufhin die Variable "DiagFlag" wieder zurückgesetzt wird.

GlobalState: GlobalState[0] gibt den Zustand der FC310x an, GlobalState[1-2] zeigen globale Bus-Stati an, GlobalState[3] ist reserviert für Erweiterungen:

RESET (1): Router der Karte wurde nicht gestartet (nach dem Hochlauf des PCs)

INIT (2): Router gestartet, Karte aber nicht am PROFIBUS aktiv

STOP (3): Karte am PROFIBUS aktiv, aber kein zyklischer Datenaustausch

STOPPING (4): Karte beendet den zyklischen Datenaustausch

RUN (0): Karte im zyklischen Datenaustausch

GlobalState[1] zählt die festgestellten Bus-Errors (ab FC310x, Version 1)

GlobalState[2]: Bit 0 ist gesetzt, wenn keine 11 Bit-Ruhezeit auf dem PROFIBUS erkannt wird (-> Verkablung prüfen), Bit 1 enthält die Betriebsart CLEAR (Bit 1 = 1) oder OPERATE (Bit 1 = 0), die anderen Bits (2 bis 7) sind reserviert für Erweiterungen (ab FC310x, Version 1)

GlobalState[3] ist reserviert für Erweiterungen

CycleFailedCounter: Dieser Zähler zeigt an, wie oft beim Start der TwinCAT-Task der PROFIBUS-Zyklus der FC310x noch nicht fertig war.

StartRedundantMasterFlag: wird für den Redundanz-Mode [► 39] benutzt

Master-Diagnosedaten

Die Master-Diagnosedaten können per ADS [► 29] ausgelesen werden:

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter <u>ADS</u> [► 45] des Devices)
Port	200
IndexGroup	0x0000F100
IndexOffset	Offset innerhalb der Diagnosedaten
Length	Länge der auszulesenden Diagnosedaten
Data	Diagnosedaten

Die Master-Diagnosedaten sind wie folgt aufgebaut:

Offset	Beschreibung
0 - 125	BusStatus-Liste, je Stationsadresse 0-125 ein Byte, das den Status der Station enthält (s. BoxState bei den PROFIBUS-Boxen, zusätzlich für nicht projektierte Stationen: 0x80 - nicht vorhanden, 0x81 - Slave, 0x82 - Master nicht bereit für Token-Ring, 0x83 - Master bereit für Token-Ring, 0x84 - Master im Token-Ring)
126 - 127	reserviert
128 - 135	Zustand der FC310x (->GlobalState)
136 - 137	Sende-Fehler-Zähler über alle gesendeten Telegramme
138 - 139	Empfangs-Fehler-Zähler über alle empfangenen Telegramme
140 - 255	reserviert für Erweiterungen
256 - 257	Sync-Failed-Counter (s. Karteireiter <u>EquiDiag</u> [► 47] (TwinCAT 2.8) bzw. <u>MC-Diag</u> [► 47] (TwinCAT 2.9))
258 - 259	Zyklus-Start-Fehler-Zähler, zählt hoch, wenn der PROFIBUS-Zyklus erneut gestartet wird bevor der alte Zyklus beendet wurde (wird vom TwinCAT-IO-Treiber abgefangen, nur bei kundenspezifischen Treibern möglich)
260 - 261	Time-Control-Failed-Counter (s. Karteireiter <u>EquiDiag</u> [► 47] (TwinCAT 2.8) bzw. <u>MC-Diag</u> [► 47] (TwinCAT 2.9))
262 - 263	reserviert für Erweiterungen
264 - 265	minimaler Nachladewert des Echtzeittimers
266 - 267	maximaler Nachladewert des Echtzeittimers (max. FCxxxx-Jitter (s. Karteireiter <u>EquiDiag</u> [► 47] (TwinCAT 2.8) bzw. <u>MC-Diag</u> [► 47] (TwinCAT 2.9)) = max. Nachladewert - min. Nachladewert)
268 - 269	PLL-Overflow-Counter (s. Karteireiter <u>EquiDiag</u> [► 47] (TwinCAT 2.8) bzw. <u>MC-Diag</u> [► 47] (TwinCAT 2.9))
270 - 271	PLL-Underflow-Counter (s. Karteireiter <u>EquiDiag</u> [► 47] (TwinCAT 2.8) bzw. <u>MC-Diag</u> [► 47] (TwinCAT 2.9))

Tclo-Diagnosedaten

Der Tclo-Treiber generiert ebenfalls Diagnosedaten, die per ADS ausgelesen, aktiviert, deaktiviert und zurückgesetzt werden können, aber defaultmäßig deaktiviert sind. Sie werden aktiviert, wenn der Karteireiter [EquiDiag \[▶ 47\]](#) oder [GeneralDiag \[▶ 45\]](#) (TwinCAT 2.8) [MC-Diag \[▶ 47\]](#) bzw. [DP-Diag \[▶ 55\]](#) oder des Devices angewählt wird und deaktiviert, wenn der Karteireiter abgewählt wird.

Aktivieren, Deaktivieren und Zurücksetzen der Tclo-Diagnosedaten

ADS-Write-Parameter	Bedeutung
Net-ID	Net-ID des PCs
Port	300
IndexGroup	0x00005000 + Device-Id (Karteireiter Allgemein des Devices)
IndexOffset	0xFFFFF100
Length	2
Data	0: Deaktivieren der Tclo-Diagnosedaten 1: Aktivieren der Tclo-Diagnosedaten 2: Zurücksetzen der Tclo-Diagnosedaten

Auslesen der Tclo-Diagnosedaten

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des PCs
Port	300
IndexGroup	0x00005000 + Device-Id (Karteireiter Allgemein des Devices)
IndexOffset	0xFFFFF100
Length	Länge der Tclo-Diagnosedaten
Data	Tclo-Diagnosedaten

Die Tclo-Diagnosedaten sind wie folgt aufgebaut:

Offset	Beschreibung
0 - 3	Max. TwinCAT-Jitter (in 100 ns, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
4 - 7	Min. Mapping-Time (in 100 ns, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
8 - 11	Max. Mapping-Time (in 100 ns, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
12 - 15	Max. FC310x-Jitter (in FC310x-Ticks, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
16 - 19	CycleWithNoDxch-Counter (s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
20 - 23	CycleWithRepeat-Counter (s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
24 - 27	Max. Repeat/Cycle (s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
28 - 31	Actual Cycle-Time (in 4/25 µs, s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
32 - 35	Max. Cycle-Time (in 4/25 µs, s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
36 - 39	Min. Cycle-Time (in 4/25 µs, s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
40 - 43	RealFailedCycle-Counter (s. Karteireiter GeneralDiag [▶ 45] (TwinCAT 2.8) bzw. DP-Diag [▶ 45] (TwinCAT 2.9))
44 - 47	EquiCycleNoDxch-Counter (s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
48 - 51	EquiCycleRepeat-Counter (s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
52 - 55	Max. Repeats/Equi-Cycle (s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
56 - 59	Actual Equi-Cycle-Time (in 4/25 µs, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
60 - 63	Max. Equi-Cycle-Time (in 4/25 µs, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))
64 - 67	Min. Equi-Cycle-Time (in 4/25 µs, s. Karteireiter EquiDiag [▶ 47] (TwinCAT 2.8) bzw. MC-Diag [▶ 47] (TwinCAT 2.9))

3.4.4 Slave-Diagnose

DP-State

Je DP-Slave gibt es eine Status-Variable, die den aktuellen Zustand des DP-Slaves anzeigt. Dieser Status ist ein Echtzeit-Status, d.h. er paßt immer zu den aktuellen Daten des DP-Slaves und kann mit einer PLC-Variable verknüpft werden (-> [DpState \[▶ 26\]](#) des Slaves):



Abb. 16: Slave-Diagnose - DP-State

Diagnosedaten

Jeder DP-Slave kann während des Data_Exchange-Betriebs azyklisch DP-Diagnosedaten melden. Dabei setzt der Slave in der Antwort des zyklischen Data_Exchange-Telegramms das Diag_Flag, woraufhin der DP-Master automatisch die DP-Diagnosedaten beim Slave ausliest. Der Data-Exchange-Zyklus wird beim Beckhoff-DP-Master dabei nicht beeinflusst, da das DP-Diagnosetelegramm am Ende des zyklischen Data-Exchange-Zyklus (vor dem Anfang des nächsten Zyklus) gesendet wird. Wenn sich die beim Slave ausgelesenen DP-Diagnosedaten gegenüber dem letzten Zustand geändert haben, setzt der DP-Master die Variable "ExtDiagFlag", die mit einer Variable des Steuerungsprogramms verknüpft werden kann.

Die aktuellen Diagnosedaten des DP-Slaves werden im System Manager Karteireiter [Diag \[▶ 63\]](#) des Slaves angezeigt. Außerdem können sie per [ADS \[▶ 29\]](#) vom Steuerungsprogramm ausgelesen werden, woraufhin die Variable "ExtDiagFlag" wieder zurückgesetzt wird:

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	200
IndexGroup	0x00yyF181 (yy = Stationsadresse des Slaves)
IndexOffset	Offset innerhalb der Diagnosedaten
Length	Länge der auszulesenden Diagnosedaten
Data	Diagnosedaten

Die Diagnosedaten beinhalten die Slave-Statistiken (32 Bytes) und die vom Slave gesendeten DP-Diagnosedaten (bis zu 244 Bytes) und sind wie folgt aufgebaut:

Offset	Bedeutung
Slave-Statistiken	
0	Receive-Error-Counter (WORD): Anzahl der fehlerhaften Telegramme bei der Kommunikation mit diesem Slave
2	Repeat-Counter[8] (WORD): Die Repeat-Counter zeigen an, wie oft wie viele Repeats gemacht werden mussten. Repeat-Counter[0] zeigt an, wie oft ein Telegramm zu diesem Slave einmal wiederholt werden musste, Repeat-Counter[1], wie oft ein Telegramm zu diesem Slave zweimal wiederholt werden musste, etc. Die maximale Anzahl der Wiederholungen wird mit dem Parameter Max Retry-Limit (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Bus-Parameter [▶ 50]) eingestellt, der Wertebereich geht von 0 bis 8, daher gibt es hier 8 Repeat-Counter (für 1 bis 8 Wiederholungen)
18	reserviert für Erweiterungen
20	NoAnswer-Counter (DWORD): Anzahl der Telegramme bei der Kommunikation mit diesem Slave, auf die nicht geantwortet wurde. Wenn ein Slave das erste Mal nicht antwortet, wird entsprechend des MaxRetryLimit das Telegramm wiederholt, wenn er auch dann nicht geantwortet hat, wird beim nächsten Mal keine Wiederholung mehr durchgeführt.
24-27	Last-DPV1-Error[4] (BYTE): Hier wird die letzte fehlerhafte DPV1-Antwort eingetragen (Byte 0: DPV1-Dienst (Bit 7 ist gesetzt und zeigt damit einen Fehler an), Byte 1: Error_Decode, Byte 2: Error_Code_1 (Error_Class/Error_Code), Byte 3: Error_Code_2), s. Beschreibung DPV1-Fehlercodes [▶ 73]
27-31	reserved for future use
ab 32	DP-Diagnosedaten [▶ 70]

3.4.5 DP-State der Slaves

Wert	Beschreibung
0	No Error - Station ist im Datenaustausch
1	Station deactivated - Slave wurde deaktiviert, tritt zur Zeit nur temporär beim Hochlauf auf
2	Station not exists - Slave antwortet nicht am Bus -> Prüfen, ob Slave eingeschaltet, ob PROFIBUS-Stecker angeschlossen, ob Stationsadresse korrekt oder ob Busverkablung korrekt
3	Master lock - Slave ist mit anderem Master im Datenaustausch -> anderen Master vom Bus nehmen oder Slave mit anderem Master wieder freigeben
4	Invalid slave response - falsche Antwort des Slaves, tritt temporär auf, wenn Slave aufgrund eines lokalen Ereignisses den Datenaustausch beendet hat
5	Parameter fault - Parametrierungsfehler -> Prüfen, ob Buskoppler bzw. GSD-Datei korrekt, ob Stationsadresse korrekt oder ob UserPrmData-Einstellungen korrekt
6	Not supported - DP-Funktion wird nicht unterstützt -> Prüfen, ob GSD-Datei korrekt oder ob Stationsadresse korrekt
7	Config fault - Konfigurationsfehler -> Prüfen, ob die angefügten Klemmen bzw. Module korrekt sind
8	Station not ready -> Station im Hochlauf, wird während des Hochlaufs temporär angezeigt
9	Static diagnosis - Slave meldet statische Diagnose und kann zur Zeit keine gültigen Daten liefern -> Betriebszustand am Slave prüfen
10	Diagnosis overflow - Slave meldet einen Diagnose-Überlauf -> Diagnosedaten (mit ADS-Read, s.u.) und Betriebszustand am Slave prüfen
11	Physical fault - physikalischer Fehler bei Antwort eines Slaves -> Verkabelung prüfen
13	Severe Bus fault - schwerer Busfehler -> Verkabelung prüfen
14	Telegram fault - Slave antwortet mit einem ungültigen Telegramm -> darf nicht auftreten
15	Station has no resources -> Slave hat nicht genügend Ressourcen für das Telegramm -> Prüfen, ob GSD-Datei korrekt
16	Service not activated -> temporärer Fehler, falls Slave aufgrund eines lokalen Ereignisses den Datenaustausch beendet, oder prüfen, ob DP-Funktionalität beim Slave disabled ist
17	Unexpected telegram -> unerwartetes Telegramm, kann temporär auftreten, falls zwei PROFIBUS-Netze zusammengesteckt werden oder prüfen, ob Buszeiten bei zweitem Master gleich eingestellt sind.
18	Station ready -> tritt temporär beim Hochlauf auf und solange die Task noch nicht gestartet ist
19	DPV1 StartUp -> tritt temporär nach dem DP-Hochlauf auf, wenn noch Daten per DPV1-Write gesendet werden
128	FC310x in Slave-Mode, waiting for data transfer -> Slave wurde parametrierung und konfiguriert hat aber noch keine Data_Exchange-Telegramm empfangen
129	FC310x in Slave-Mode, waiting for configuration -> Slave wurde parametrierung, hat aber noch kein Chk_Cfg-Telegramm empfangen
130	FC310x in Slave-Mode, waiting for parameter -> Slave wurde noch nicht parametrierung, wartet auf Set_Prm (Lock)-Telegramm

3.5 PROFIBUS MC

PROFIBUS MC unterscheidet sich von PROFIBUS DP dadurch, dass der PROFIBUS-Zyklus mit einem konstanten Zyklus mit wenigen Mikrosekunden Jitter erfolgt (bei PROFIBUS DP ist der Jitter größer 100 µs) und am Anfang des Zyklus ein Broadcast-Global-Control-Telegramm gesendet wird, auf das sich die MC-Slaves synchronisieren können. Dadurch ist es möglich, Antriebsregelkreise genau mit der NC zu synchronisieren.

Die genaue Synchronisierung hat aber zur Folge, dass Busstörungen, Ausschalten von Slaves, Abziehen von Bussteckern, etc. in der Regel dazu führen, dass die Synchronität zwischen Master und Slave verloren geht, da sich das Bus-Timing dadurch verändert.

FC310x mit Simodrive 611U ist Plug&Play

Um einen Simodrive 611U an einer FC310x in Betrieb zu nehmen, ist das Folgende zu tun:

1. FC310x auf Betriebsart "DP/MC (Equidistant)" stellen.
2. Box "Siemens AG, Profidrive MC" anfügen.
3. Stationsadresse des 611Us anpassen (Box-Karteireiter "PROFIBUS").
4. Achse (bzw. 2 Achsen bei 611U mit 2 Achsen) an NC-Task anhängen, Achstyp "kontinuierliche Achse" auswählen.
5. Achse (bzw. Achsen) mit 611U verknüpfen (in Achs-Karteireiter "Settings" den Achstyp "ProfiDrive MC" auswählen, dann mit 611U verknüpfen, bei einem 2-Achs-611U sind immer beide Achsen zu verknüpfen, sonst gibt es einen Fehler auf dem 611U).
6. An der FC310x auf dem Karteireiter "FC310x" den Button "Calculate Equi-Times" drücken.
7. Projekt in der Registry speichern, TwinCAT starten, der 611U müsste dann auf RUN gehen, die Achse kann im NC-Online-Menü bedient werden.

Falls das nicht Fall ist, sollten die folgenden Dinge überprüft werden:

- DpState des 611Us in TwinCAT ist 2: Stationsadresse des 611Us prüfen.
- DpState des 611Us in TwinCAT ist 5: Prüfen, ob das richtige PROFIBUS-Modul am 611U gesteckt ist.
- DpState des 611Us in TwinCAT ist 7: Prüfen, ob P922 auf dem richtigen Standardtelegramm steht (entsprechend Karteireiter ProcessData beim 611U im System Manager).
- DpState des 611Us in TwinCAT ist 0, aber 611U trotzdem nicht im RUN: die Firmware-Version des 611Us überprüft werden, bei Firmware-Versionen kleiner 3.4.3 lässt sich ein Synchronisierungsfehler auf dem 611U (Fehler 597 bzw. 598) nur durch einen Hardware-Reset des 611Us beheben, ansonsten Fehlercode im Siemens-Handbuch nachschauen.

Wenn mehrere 611Us konfiguriert werden, müssen ggf. die Equidistant-Zeiten angepasst werden (s.u.).

DP/MC-Equidistant-Mode

Um die FC310x mit PROFIBUS MC zu betreiben, muss auf dem Karteireiter "**FC310x**" (für TwinCAT 2.8 [▶ 41] bzw. TwinCAT 2.9 [▶ 49]) des Masters der **Operation Mode** "DP/MC(Equidistant)" eingestellt werden. Die Task, die die Äquidistanz-Funktionalität der FC310x nutzt (in der Regel die NC-Task), sollte die höchste Priorität haben, ansonsten kann es zu Störungen in der Synchronität kommen. Zusätzlich kann noch der **Sync-Mode** ausgewählt werden, die festlegt, wo das Synchronisationssignal generiert wird:

Disabled (PC ist Sync-Master)

Das Synchronisationssignal wird vom PC generiert, die FC310x synchronisiert sich auf den PC (Jitter des PROFIBUS-Zyklus ca. 2-4 µs).

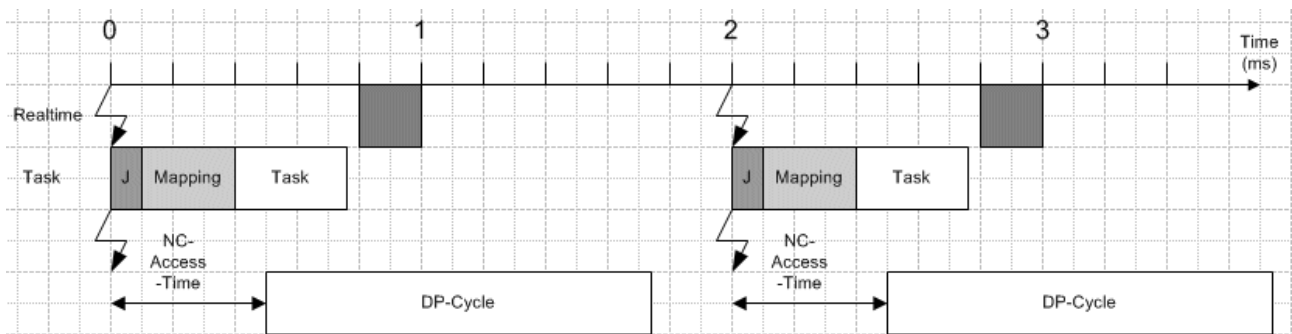


Abb. 17: PC ist Sync-Master

Die **NC-Access-Time** gibt an, um wie viel der PROFIBUS-Zyklus zum TwinCAT-Zyklus verschoben ist, die **PLL-Sync-Time** sollte auf ca. 10% der **NC-Access-Time** eingestellt werden (max. 50 µs).

Sync-Slave

Das Synchronisierungssignal kommt von einem anderen Device, dessen Sync-Mode auf "Sync-Master" eingestellt sein muss. Die Verbindung zwischen Sync-Master und Sync-Slave erfolgt durch einen Hardware-Link, der erst ab Hardware-Version 4 und Firmware-Version 3.00 der FC310x unterstützt wird. Zeiten sind keine einzustellen.

Sync-Master

Das Synchronisationssignal wird von der FC310x generiert, der PC synchronisiert sich auf die FC310x (Jitter des PROFIBUS-Zyklus ca. 1 µs).

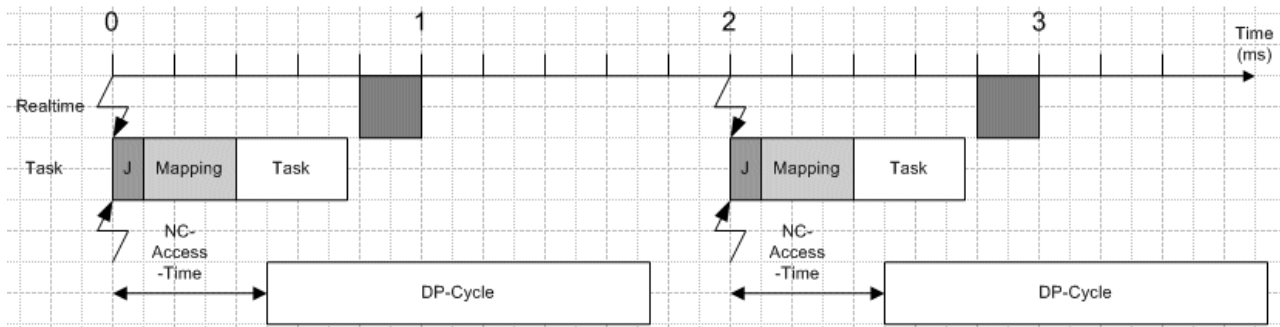


Abb. 18: FC310x ist Sync-Master

Die **NC-Access-Time** gibt wiederum an, um wie viel der TwinCAT-Zyklus zum PROFIBUS-Zyklus verschoben ist.

Einstellung der Equidistant-Zeiten

Mit dem Button **Calculate Equi-Times** (TwinCAT 2.8: s. Karteireiter [FC310x](#) [► 41], TwinCAT 2.9: s. Karteireiter [MC](#) [► 53]) können alle Equidistant-Parameter automatisch eingestellt werden. Der einzige Parameter, der danach ggf. noch angepasst werden muss, ist die **NC-Access-Time**, da diese vom maximalen TwinCAT-Jitter und der maximalen Mapping-Time (geht über alle Devices, d.h. das Hinzufügen von und Verknüpfen von Boxes an anderen Devices führt dazu, dass trotzdem die **NC-Access-Time** eines unveränderten Devices zu ändern ist) abhängt. Falls "I/O nicht am Task-Anfang" angewählt ist (s. Kapitel [Synchronisierung](#) [► 12]), hängt die **NC-Access-Time** auch noch von der Task-Laufzeit ab. Damit nach dem Drücken des Buttons **Calculate Equi-Times** die **NC-Access-Time** nicht jedes Mal manuell angepasst werden muss, kann das Verhältnis von **NC-Access-Time** zu **Cycle-Time** festgelegt werden (im Auslieferungszustand auf 15%).

Disabled (PC ist Sync-Master) bzw. Sync-Master

Die **NC-Access-Time** muss größer als Max. TwinCAT-Jitter plus Max. Mapping-Time (plus Task-Laufzeit der höchstpriorären mit der FC310x verknüpften Task (falls "I/O nicht am Task-Anfang" bei dieser Task angewählt ist)) sein.

Diagnose der Equidistant-Zeiten

Die Diagnose der Equidistanz-Zeiten kann mit dem Karteireiter [EquiDiag](#) [► 47] (TwinCAT 2.8) bzw. [MC-Diag](#) [► 57] (TwinCAT 2.9) im System Manager bzw. per ADS im Steuerungsprogramm erfolgen (s. Kapitel [Master-Diagnose](#) [► 20]).

3.6 ADS (azyklische Kommunikation)

3.6.1 ADS-Interface

Sämtliche azyklischen Daten werden mit ADS-Read, ADS-Write oder ADS-Write-Control an die bzw. von der FC310x übertragen. Die FC310x hat eine eigene Net-ID und unterstützt die folgenden Ports:

Port	Beschreibung
200	damit wird die FC310x selbst adressiert, d.h. Daten, die lokal auf der FC310x liegen, bei denen in der Regel kein zusätzlicher Buszugriff nötig ist
0x1000 - 0x107E	damit wird ein angeschlossener PROFIBUS-Teilnehmer adressiert, wobei sich Adresse aus Port-0x1000 berechnet, es wird auch immer ein Buszugriff durchgeführt

ADS-Read

Es folgt eine Übersicht der von der FC310x unterstützten IndexGroup/IndexOffset bei ADS-Read.

IndexGroup bei lokaler Adressierung der FC310x (Port 200)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Beschreibung
0xF100	0x00	BYTE-Offset innerhalb der Daten	Damit werden die Diagnosedaten der FC310x ausgelesen. Wenn der ADS-Read ohne Fehler (Error-Code = 0) beantwortet wird, enthalten die Daten die im Kapitel Master-Diagnose [▶ 20] beschriebenen Diagnosedaten der FC310x. Die FC310x setzt dabei das DiagFlag der FC310x zurück, das dann wieder gesetzt wird, wenn sich die Diagnosedaten der FC310x erneut ändern.
0xF181	0x00-0x7E	BYTE-Offset innerhalb der Daten	Damit werden die Diagnosedaten eines projektierten DP-Slaves ausgelesen, die Stationsadresse berechnet sich aus dem IndexGroup(Hi-Word). Wenn der ADS-Read ohne Fehler (Error-Code = 0) beantwortet wird, enthalten die Daten die im Kapitel Slave-Diagnose [▶ 24] beschriebenen Diagnosedaten eines projektierten DP-Slaves.
0xF830	0x8000-0x807E	immer 0	Damit kann festgestellt werden, welche DP-Slaves am PROFIBUS vorhanden sind, unabhängig davon, ob sie projektiert wurden oder nicht, die Stationsadresse berechnet sich aus IndexGroup(Hi-Word)-0x8000. Wenn der ADS-Read ohne Fehler (Error-Code = 0) beantwortet wird, hat der entsprechende DP-Slave korrekt geantwortet, die Daten enthalten die Ident-Nummer des Slaves (BYTE-Offset 0-1) und die ausgelesenen CfgData (ab BYTE-Offset 2) (s. Kapitel Upload Configuration [▶ 35]).
0xF840	0	BYTE-Offset innerhalb der Daten	Damit werden Firmware-Version und Stationsadresse der FC310x ausgelesen. Wenn der ADS-Read ohne Fehler (Error-Code = 0) beantwortet wird, enthalten die Daten die Firmware-Version (BYTE-Offset 0-1) und die Stationsadresse der FC310x (BYTE-Offset 2).

IndexGroup bei Adressierung eines projektierten PROFIBUS-Teilnehmers (Port 0x1000-0x107E)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Beschreibung
0x00-0xFF	0x00	0x00-0xFF	Damit wird ein DPV1-Read an den entsprechenden, projektierten DPV1-Slave über eine Class 1-Verbindung gesendet, die DPV1-Slot-Nummer entspricht der IndexGroup, der DPV1-Index entspricht dem IndexOffset. Wenn der ADS-Read ohne Fehler (Error-Code = 0) beantwortet wird, enthalten die Daten die ausgelesenen DPV1-Daten (s. Kapitel DPV1 [▶ 32])
0x100-0x1FF	0x00	0x00-0xFF	Damit wird ein DPV1-Read an den entsprechenden, projektierten DPV1-Slave über eine Class 2-Verbindung gesendet, die DPV1-Slot-Nummer entspricht der IndexGroup - 0x100, der DPV1-Index entspricht dem IndexOffset. Wenn der ADS-Read ohne Fehler (Error-Code = 0) beantwortet wird, enthalten die Daten die ausgelesenen DPV1-Daten (s. Kapitel DPV1 [▶ 32])
0x0000 - 0xFFFF	0x10000000 - 0xF0000000	0x00-0xFF	Damit wird ein PKW-Read an den entsprechenden, projektierten PROFIDRIVE-Slave gesendet, die Parameternummer (PNU) steht im Lo-Word der IndexGroup, der Sub-Index bei Zugriff auf ein Array steht im IndexOffset, die angesprochene Achse steht in den Bits 28-31 der IndexGroup (bei einem 1-Achser muss hier eine 1 stehen), in Bit 26,27 kann noch die PKW-Kompatibilität angepasst werden (leider sind nicht alle PROFIDRIVE-Slaves hier kompatibel, s. Kapitel PKW-Interface [▶ 36]).
0	0x01000000	0	Damit wird FDL-Read für die Siemens AG-Kopplung an die entsprechende projektierte FDL-Station gesendet (s. Kapitel S5-FDL [▶ 37]).

ADS-Write

Es folgt eine Übersicht der von der FC310x unterstützten IndexGroup/IndexOffset bei ADS-Write.

IndexGroup bei lokaler Adressierung der FC310x (Port 200)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Beschreibung
0xF100	0x00	0 -2	Damit werden die Equidistant-Diagnosedaten (IndexOffset = 0), die Repeat-Counter (IndexOffset = 1) oder die NoAnswer-Counter (IndexOffset = 2) der FC310x zurückgesetzt.

IndexGroup bei Adressierung eines projektierten PROFIBUS-Teilnehmers (Port 0x1000-0x107E)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Beschreibung
0x00-0xFF	0x00	0x00-0xFF	Damit wird ein DPV1-Write an den entsprechenden, projektierten DPV1-Slave über eine Class 1-Verbindung gesendet, die DPV1-Slot-Nummer entspricht der IndexGroup, der DPV1-Index entspricht dem IndexOffset (-> Kapitel DPV1 [► 32]).
0x100-0x1FF	0x00	0x00-0xFF	Damit wird ein DPV1-Write an den entsprechenden, projektierten DPV1-Slave über eine Class 2-Verbindung gesendet, die DPV1-Slot-Nummer entspricht der IndexGroup - 0x100, der DPV1-Index entspricht dem IndexOffset (-> Kapitel DPV1 [► 32]).
0x400	0x00	0x00	Damit wird ein DPV1-Abort an den entsprechenden, projektierten DPV1-Slave über eine Class 2-Verbindung gesendet, die Abort-Parameter befinden sich in den Daten (-> Kapitel DPV1 [► 32]).
0x0000 - 0xFFFF	0x10000000 - 0xF0000000	0x00-0xFF	Damit wird ein PKW-Write an den entsprechenden, projektierten PROFIDRIVE-Slave gesendet, die Parameternummer (PNU) steht im Lo-Word der IndexGroup, der Sub-Index bei Zugriff auf ein Array steht im IndexOffset, die angesprochene Achse steht in den Bits 28-31 der IndexGroup (bei einem 1-Achser muss hier eine 1 stehen), in Bit 26,27 kann noch die PKW-Kompatibilität angepasst werden (leider sind nicht alle PROFIDRIVE-Slaves hier kompatibel, -> Kapitel PKW-Interface [► 36]).
0	0x01000000	0	Damit wird FDL-Write für die Siemens AG-Kopplung an die entsprechende projektierte FDL-Station gesendet (s. Kapitel S5-FDL [► 37]).
0	0x02000000	0	Damit wird ein SetSlaveAddress-Kommando an einen projektierten DP-Slave gesendet, wobei der DP-Slave mit der neuen Stationsadresse projektiert werden muss, die alte Stationsadresse ist dann am BYTE-Offset 0 der ADS-Write-Daten einzutragen. Weiterhin muss unter BYTE-Offset 1 und 2 die Ident-Nummer des Slaves stehen und unter BYTE-Offset 3, ob die Slave später noch einmal geändert werden darf (0) oder nicht (ungleich 0). Insgesamt sind also 4 Bytes ADS-Write-Daten zu übergeben.

ADS-ReadWrite

Es folgt eine Übersicht der von der FC310x unterstützten IndexGroup/IndexOffset bei ADS-ReadWrite.

IndexGroup bei Adressierung eines projektierten PROFIBUS-Teilnehmers (Port 0x1000-0x107E)

Index-Group (Lo-Word)	IndexGroup (Hi-Word)	IndexOffset	Beschreibung
0x100-0x1FF	0x00	0x00-0xFF	Damit wird ein DPV1-Data_Transport an den entsprechenden, projektierten DPV1-Slave über eine Class 2-Verbindung gesendet, die DPV1-Slot-Nummer entspricht der IndexGroup - 0x100, der DPV1-Index entspricht dem IndexOffset (-> Kapitel DPV1 [► 32]).
0x200	0x00	0x00	Damit wird ein DPV1-Initiate an den entsprechenden, projektierten DPV1-Slave über eine Class 2-Verbindung gesendet, die Initiate-Parameter befinden sich in den Daten (-> Kapitel DPV1 [► 32]).

ADS-WriteControl

Es folgt eine Übersicht der von der FC310x unterstützten ADS-Write-Control-Kommandos.

ADS-WriteControl bei lokaler Adressierung der FC310x (Port 200)

AdsState	DeviceState	State der FC310x	Beschreibung
STOP (6)	0x00	RUN (5)	Damit wird die FC310x gestoppt, d.h. die Prozessdatenverbindungen zu allen DP-Slaves (Data_Exchange) werden abgebaut (mit SetPrm, Unlock).
RUN (5)	0x00	STOP (6)	Damit wird die FC310x nach einem Stoppen erneut gestartet, d.h. die Prozessdatenverbindungen zu allen DP-Slaves (Data_Exchange) werden wieder aufgebaut (normaler DP-Hochlauf).

ADS-WriteControl bei Adressierung eines projektierten PROFIBUS-Teilnehmer (Port 0x1000-0x107E)

AdsState	DeviceState	State der FC310x	Beschreibung
STOP (6)	0x00	RUN (5)	Damit wird der Slave gestoppt, d.h. die Prozessdatenverbindung zu dem entsprechenden DP-Slave (Data_Exchange) wird abgebaut (mit SetPrm,Unlock).
RUN (5)	0x00	STOP (6)	Damit wird der Slave nach einem Stoppen erneut gestartet, d.h. die Prozessdatenverbindung zu dem entsprechenden DP-Slave (Data_Exchange) wird wieder aufgebaut (normaler DP-Hochlauf).

ADS-Fehlercodes

Der 32-Bit-ADS-Fehlercode besteht immer aus einem allgemeinen ADS-Fehlercode (Lo-Word, siehe ADS-Dokumentation) und einem FC310x-spezifischen, eindeutigen Fehlercode (Hi-Word, -> Kapitel [FC310x-ADS-Fehlercodes](#) [▶ 38]). Die entsprechende Meldung wird auch im Logger des TwinCAT System Managers textuell angezeigt.

3.6.2 PROFIBUS DPV1

Der Master unterstützt auf einer C1-Verbindung die Dienste Read und Write sowie auf einer C2-Verbindung die Dienste Read, Write, Data_Transport, Initiate und Abort.

C1-Verbindung (MSAC-C1)

Die C1-Verbindung ist für den Master reserviert, die mit dem Slave zyklischen Datenaustausch durchführt (C1-Master). Um die C1-Verbindung bei einem Slave nutzen zu können, muss der Slave DPV1 unterstützen (in der GSD-Datei müssen dafür die Zeile "DPV1_Slave = 1" und das Schlüsselwort "C1_Max_Data_Len" mit einer entsprechenden Länge eingetragen sein). Weiterhin ist in der Regel die C1-Funktionalität durch Setzen des Bit 7 von **PrmData**-Byte 0 (s. Karteireiter [PROFIBUS](#) [▶ 59] des Slaves) bei dem entsprechenden Slave zu aktivieren (bei Beckhoff-Geräten, die DPV1 unterstützen, passiert das automatisch).

MSAC-C1-Read ist auf ADS-Read, MSAC-C1-Write auf ADS-Write abgebildet:

MSAC-C1 Read

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	Slot-Number (DPV1-Parameter)
IndexOffset	Index (DPV1-Parameter)
Length	Länge der auszulesenden Daten
Data	bei Response: ausgelesene Daten

MSAC-C1 Write

ADS-Write-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	Slot-Number (DPV1-Parameter)
IndexOffset	Index (DPV1-Parameter)
Length	Länge der zu schreibenden Daten
Data	bei Request: zu schreibende Daten

C2-Verbindung (MSAC-C2)

Die C2-Verbindung ist in der Regel für einen zweiten Master (C2-Master) gedacht, der nicht zyklisch mit dem Slave kommuniziert, aber auch der C1-Master kann die C2-Verbindung nutzen. Um die C2-Verbindung bei einem Slave nutzen zu können, muss der Slave DPV1 unterstützen (in der GSD-Datei müssen dafür die Zeile "DPV1_Slave = 1" und das Schlüsselwort "C2_Max_Data_Len" mit einer entsprechenden Länge eingetragen sein).

Die Verbindung wird von dem Master automatisch aufgebaut, sobald ein Read, Write oder Data_Transport-Zugriff gefordert wird, sie kann aber auch explizit per Initiate aufgebaut werden. Beim automatischen Aufbau sendet der Master die zuletzt übergebenen Initiate-Parameter (s. Beschreibung Initiate), wobei er die Initiate-Parameter nach einem TwinCAT Start (Restart) mit 0 initialisiert (Ausnahme ist die Verbindung-Überwachung, die entsprechend des im System Manager eingestellten Wertes (**Watchdog** unter **DPV1-Class 2** auf dem Karteireiter [PROFIBUS \[▶ 59\]](#) des Slaves) initialisiert wird).

Weiterhin muss die C2-Funktionalität durch Anwählen der **Enable**-Check-Box unter **DPV1-Class 2** (s. Karteireiter [PROFIBUS \[▶ 59\]](#) des Slaves) bei jedem Slave, der mit C2-Diensten angesprochen werden soll, aktiviert werden.

Wenn ein anderer Master mit dem Slave zyklischen Datenaustausch macht, muss unter **DP-Class 2** die Einstellung "No cyclic connection" angewählt werden (s. Karteireiter [PROFIBUS \[▶ 59\]](#) des Slaves). Das ist z. B. sinnvoll, um einen BC3100/IL23xx-C310 per PROFIBUS debuggen zu können, obwohl er an einer Fremd-Steuerung hängt.

MSAC-C2-Read ist auf ADS-Read, MSAC-C2-Write auf ADS-Write, MSAC-C2-Data_Transport auf ADS-ReadWrite, MSAC-C2-Initiate auf ADS-ReadWrite und MSAC-C2-Abort auf ADS-Write abgebildet:

MSAC-C2 Read

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	0x100 + Slot-Number (DPV1-Parameter)
IndexOffset	Index (DPV1-Parameter)
Length	Länge der auszulesenden Daten
Data	bei Response: ausgelesene Daten

MSAC-C2 Write

ADS-Write-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	0x100 + Slot-Number (DPV1-Parameter)
IndexOffset	Index (DPV1-Parameter)
Length	Länge der zu schreibenden Daten
Data	bei Request: zu schreibende Daten

MSAC-C2 Data_Transport

ADS-ReadWrite-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	0x100 + Slot-Number (DPV1-Parameter)
IndexOffset	Index (DPV1-Parameter)
Write-Length	Länge der zu schreibenden Daten
Read-Length	Länge der auszulesenden Daten
Data	bei Request: zu schreibende Daten, bei Response: ausgelesene Daten

MSAC-C2 Initiate

Mit dem MSAC-C2-Initiate-Dienst kann die C2-Verbindung zu dem Slave aufgebaut bzw. neue Initiate-Parameter übergeben werden, falls diese schon aufgebaut ist.

ADS-ReadWrite-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	0x200 + Slot-Number (DPV1-Parameter)
IndexOffset	0
Read-Length	Länge der Initiate-Response-Parameter (6)
Write-Length	Länge der Initiate-Request-Parameter (10 - 42)
Data	Initiate-Request-Parameter bzw. Initiate-Response-Parameter

Initiate-Request-Parameter

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

Initiate-Response-Parameter

0x00 - 0x01	Feature_Supported (von Slave empfangener Wert)
0x02 - 0x03	Profile_Feature_Supported (von Slave empfangener Wert)
0x04 - 0x05	Profile_Ident_number (von Slave empfangener Wert)

MSAC-C2 Abort

Mit dem MSAC-C2-Abort-Dienst kann die C2-Verbindung zu dem Slave wieder abgebaut werden.

ADS-Write-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	0x400 + Slot-Number (DPV1-Parameter)
IndexOffset	0
Length	Länge der Abort-Parameter (3)
Data	bei Request: Abort-Parameter

Abort-Parameter

0x00	Reason_Code
0x01 - 0x02	Additional_Detail

3.6.3 Upload der Konfiguration

Während des Betriebs kann der PROFIBUS per [ADS \[▶ 29\]](#)-Read nach neuen Geräten gescannt werden:

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [▶ 45] des Devices)
Port	200
IndexGroup	0xzzyyF830 (yy = Stationsadresse, zz = 0: bei Beckhoff-Geräten werden die Tabellen 0,1 und 9 ausgelesen, zz = 0x80: Beckhoff-Geräte liefern die gleichen Informationen wie Fremd-Geräte)
IndexOffset	0
Length	1538
Data	Konfigurationsdaten des Slaves

Falls in der IndexGroup angegeben ist, dass bei Beckhoff-Geräten die Tabellen 0, 1 und 9 ausgelesen werden sollen, werden die folgenden Daten geliefert, sofern es sich um ein Beckhoff-Gerät handelt:

Offset	Beschreibung
0 -1	0
2 -513	Tabelle 0 (enthält u.a. die genauen Koppler-Typ und die Firmware-Version)
514 - 1025	Tabelle 9 (enthält die Koppler-Nummer und die Klemmen-Nummern)
1026 - 1537	Tabelle 1 (nur bei Buscontrollern relevant, enthält u.a. die Zuordnung der Klemmen)

Falls es sich um Fremdgeräte handelt oder in der IndexGroup angegeben ist, dass sich Beckhoff-Geräte genauso verhalten sollen wie Fremd-Geräte, werden die folgenden Informationen in der ADS-Read-Response zurückgeliefert:

Offset	Beschreibung
0 -1	1
2 - 7	DP-Diagnosedaten Byte 0-5 (s. Slave-Diagnose [▶ 24])
8 - 251	DP-Konfigurationsdaten (CfgData [▶ 74])

3.6.4 PKW-Interface von PROFIDRIVE-Slaves

Das PKW-Interface ist in der FC310x integriert, es kann dann aus dem Steuerungsprogramm per ADS [► 29] darauf zugegriffen werden. PKW-Read wird dabei auf ADS-Read, PKW-Write auf ADS-Write und PKW-Read No Of Array Elements auf ADS-Read abgebildet:

PKW Read

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	Bit 0-11: Parameter-Nummer (PNU)
	Bit 12-25: 0
	Bit 26: 1 = Subindex in Octet 3 (Standard), 0 = Subindex in Octet 4 (Simodrive 611U)
	Bit 27: 1 = ARRAY-Codes werden nicht unterstützt von PROFIDRIVE-Slave
	Bit 28-31: Nummer der Achse (bei 1-Achsen-Modulen immer 1)
IndexOffset	Subindex (bei ARRAY-Zugriffen)
Length	Parameterlänge: 2 oder 4
Data	bei Response: Parameterwert

PKW Write

ADS-Write-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	Bit 0-11: Parameter-Nummer (PNU)
	Bit 12-25: 0
	Bit 26: 1 = Subindex in Octet 3 (Standard), 0 = Subindex in Octet 4 (Simodrive 611U)
	Bit 27: 1 = ARRAY-Codes werden nicht unterstützt von PROFIDRIVE-Slave
	Bit 28-31: Nummer der Achse (bei 1-Achsen-Modulen immer 1)
IndexOffset	Subindex (bei ARRAY-Zugriffen)
Length	Parameterlänge: 2 oder 4
Data	bei Request: Parameterwert

PKW ReadNoOfArrayElements

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 45] des Devices)
Port	0x1000 + Stationsadresse des Slaves
IndexGroup	Bit 0-11: Parameter-Nummer (PNU)
	Bit 12-15: 0
	Bit 16: 1
	Bit 17-25: 0
	Bit 26: 1 = Subindex in Octet 3 (Standard), 0 = Subindex in Octet 4 (Simodrive 611U)
	Bit 27: 1 = ARRAY-Codes werden nicht unterstützt von PROFIDRIVE-Slave
	Bit 28-31: Nummer der Achse (bei 1-Achsen-Modulen immer 1)
IndexOffset	0
Length	Parameterlänge: 1
Data	bei Response: Anzahl der Array-Elemente des Parameters

3.6.5 S5-FDL-Kommunikation

Mit S5-Steuerungen bzw. anderen PROFIBUS-FDL-Geräten kann per FDL-AGAG-Kommunikation kommuniziert werden. Dabei gibt es die folgenden Festlegungen für die PROFIBUS-SAPs:

FC310x sendet

SDA-Request mit DSAP = Stationsadresse + 1 der FC310x und SSAP = Stationsadresse + 1 des anderen FDL-Geräts, Daten entsprechend der mit ADS-Write übergebenen Datenlänge, das SDA-Telegramm wird nur abgeschickt, wenn der ADS-Write-Aufruf erfolgt:

ADS-Write-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 29] des Devices)
Port	0x1000 + Stationsadresse des anderen FDL-Geräts
IndexGroup	0x01000000
IndexOffset	0
Length	Länge der zu schreibenden Daten
Data	bei Request: zu schreibende Daten

Anderes FDL-Gerät sendet:

SDA-Request mit DSAP = Stationsadresse + 1 des anderen FDL-Geräts und SSAP = Stationsadresse + 1 der FC310x, Daten entsprechend der beim anderen FDL-Gerät eingestellten Datenlänge, FC310x speichert die empfangenen Daten zwischen, die Daten können per ADS-Read ausgelesen werden:

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 29] des Devices)
Port	0x1000 + Stationsadresse des anderen FDL-Geräts
IndexGroup	0x01000000
IndexOffset	0
Length	Länge der empfangenen Daten
Data	bei Response: empfangene Daten, am Ende der empfangenen Daten steht ein Receive-Counter, der bei jedem Empfang inkrementiert wird

ADS-Read-Parameter	Bedeutung
Net-ID	Net-ID des Masters (s. Karteireiter ADS [► 29] des Devices)
Port	0x1000 + Stationsadresse des anderen FDL-Geräts
IndexGroup	0x04000000
IndexOffset	Stationsadresse des anderen FDL-Geräts * 2
Length	2
Data	bei Response: Receive-Counter, der bei jedem Empfang inkrementiert wird

3.6.6 ADS-Error-Codes der FC310x

Fehlercode	Bedeutung
0x1129	IndexOffset zu groß beim Lesen der FC310x-Diagnosedaten
0x112B	IndexOffset zu groß beim Lesen der Slave-Diagnosedaten
0x112D	ungültige Stationsadresse beim Lesen der Slave-Diagnosedaten
0x2023	ungültiger IndexOffset beim Rücksetzen der FC310x-Diagnosedaten
0x2024	ungültige Daten beim Rücksetzen der FC310x-Diagnosedaten
0x2025	ungültige Datenlänge beim Rücksetzen der FC310x-Diagnosedaten
0x2101	DPV1-C1-Read: zyklische Verbindung zu Slave noch nicht aufgebaut
0x2102	PKW-Read: nur die Datenlängen 2 und 4 sind erlaubt
0x2103	PKW-Read: Slave nicht im Datenaustausch
0x2105	PKW-Read: Slave unterstützt kein PKW
0x2106	PKW-Read: falscher IndexOffset
0x2107	PKW-Read: falsche IndexGroup
0x2109	DPV1-C1-Read: FDL-Fehler (antwortet nicht, etc.)
0x210A	DPV1-C1-Read: Syntax-Fehler (DPV1-Syntax nicht korrekt)
0x210B	DPV1-C1-Read: DPV1-Fehler (4 Byte Fehlercode in den Diagnosedaten des Slave)
0x210C	PKW-Read: Syntax-Fehler
0x210D	PKW-Read: PKW-Fehler
0x210E	PKW-Read: falscher Datentyp
0x210F	DPV1-C1-Write: zyklische Verbindung zu Slave noch nicht aufgebaut
0x2110	PKW-Write: nur die Datenlängen 2 und 4 sind erlaubt
0x2111	PKW-Write: falscher IndexOffset
0x2112	PKW-Write: Slave unterstützt kein PKW
0x2113	PKW-Write: falsche IndexGroup
0x2114	Read allgemein: falsche IndexGroup
0x2115	DPV1-C1-Write: FDL-Fehler (antwortet nicht, etc.)
0x2116	DPV1-C1-Write: Syntax-Fehler (DPV1-Syntax nicht korrekt)
0x2117	DPV1-C1-Write: DPV1-Fehler (4 Byte Fehlercode in den Diagnosedaten des Slave)
0x211C	Read allgemein: falsche IndexGroup
0x211D	SetSlaveAdress: falscher IndexOffset
0x211E	FDL-AGAG-Write: falscher IndexOffset
0x211F	FDL-AGAG-Read: falscher IndexOffset
0x2120	FDL-AGAG-Write: falsche Länge
0x2121	SetSlaveAddress: falsche Länge
0x2122	FDL-AGAG-Read: falsche Länge

Fehlercode	Bedeutung
0x2131	Write allgemein: falsche indexGroup
0x2132	Write allgemein: falsche indexGroup
0x2137	PKW-Read: WORD empfangen, aber Read-Datenlänge ungleich 2
0x2138	PKW-Read: DWORD empfangen, aber Read-Datenlänge ungleich 4
0x2139	PKW-Read: unbekannte AK empfangen (1,2 oder 7 erwartet)
0x213A	PKW-Read-Array: WORD empfangen, aber Read-Datenlänge ungleich 2
0x213B	PKW-Read-Array: DWORD empfangen, aber Read-Datenlänge ungleich 4
0x213C	PKW-Read-Array: unbekannte AK empfangen (4,5 oder 7 erwartet)
0x213D	PKW-Write-Array: unbekannte AK empfangen (2 oder 7 erwartet)
0x213E	PKW-Write: unbekannte AK empfangen (1 oder 7 erwartet)
0x213F	PKW-Write: unbekannte AK empfangen (2 oder 7 erwartet)
0x2140	PKW-Write-Array: unbekannte AK empfangen (1 oder 7 erwartet)
0x2142	SetSlaveAddress: falscher Parameter beim Setzen der Adresse im Slave-Mode
0x2144	falsche IndexGroup beim ReadWrite
0x2147	DPV1-C2-Initiate: MSAC_C2 ist nicht aktiviert
0x2148	falsche IndexGroup beim Read
0x2149	falsche IndexGroup beim Write
0x214E	DPV1-C2-Read: MSAC_C2 ist nicht aktiviert
0x214F	DPV1-C2-Write: MSAC_C2 ist nicht aktiviert
0x2150	DPV1-C2-DataTransport: MSAC_C2 ist nicht aktiviert
0x2151	DPV1-C2-Read: FDL-Fehler (antwortet nicht, etc.)
0x2152	DPV1-C2-Read: Abort der Verbindung
0x2153	DPV1-C2-Read: DPV1-Fehler (4 Byte Fehlercode in den Diagnosedaten des Slave)
0x2154	PKW-ReadNoOfElements: Länge muss gleich 1 sein
0x2155	PKW-ReadNoOfElements: PKW ist nicht aktiviert
0x2156	PKW-ReadNoOfElements: Achs-Nummer ist zu groß
0x2157	PKW-ReadNoOfElements: Slave nicht im Datenaustausch
0x2158	PKW-ReadNoOfElements: unbekannte AK empfangen (6 oder 7 erwartet)
0x215A	DPV1-C2-Write: FDL-Fehler (antwortet nicht, etc.)
0x215B	DPV1-C2-Write: Abort der Verbindung
0x215C	DPV1-C2-Write: DPV1-Fehler (4 Byte Fehlercode in den Diagnosedaten des Slave)
0x215D	DPV1-C2-DataTransport: FDL-Fehler (antwortet nicht, etc.)
0x215E	DPV1-C2-DataTransport: Abort der Verbindung
0x215F	DPV1-C2-DataTransport: DPV1-Fehler (4 Byte Fehlercode in den Diagnosedaten des Slave)
0x2163	DPV1-C2-DataTransport: falscher IndexOffset
0x2600-0x26FF	AK 7 (Fehler) bei PKW-Bearbeitung erhalten, Fehlercode in Lo-Byte
0x2700-0x27FF	Fehler bei DPV1-Bearbeitung, 4 Bytes Fehlercode in den Slave-Diagnosedaten, Byte 3 des Fehlercodes (Error-Class, Error-Code) steht in Lo-Byte

3.7 Master-Redundanz

Um ein redundantes Steuerungssystem aufzubauen, ist es möglich, den DP-Master im Redundancy-Mode zu starten. In diesem Fall hört der DP-Master nur auf dem Bus mit, ist aber nicht am Bus aktiv.

Um ein redundantes Steuerungssystem aufzubauen, hängen am PROFIBUS zwei Master, die beide die gleiche Konfiguration haben, der Primary-Master (der im Normalfall die Kommunikation übernimmt) und der Redundancy-Master (der nur auf dem Bus mithört, aber nicht sendet). Der einzige Unterschied in der PROFIBUS-Konfiguration zwischen Primary- und Redundancy-Master sollte in den Einstellungen **Redundancy-Mode** und **SetPrm-Unlock before DP-Start-Up** bzw. **SetPrm-Unlock at Shutdown**

(TwinCAT 2.8: s. Karteireiter [PROFIBUS](#) [► 43] des Masters, TwinCAT 2.9: s. Dialog [Fault-Settings](#) [► 52]) sowie ggf. dem **Watchdog** des Devices bestehen (TwinCAT 2.8: s. Karteireiter [FC310x](#) [► 41] des Masters, TwinCAT 2.9: s. Dialog [Fault-Settings](#) [► 52]).

Primary-Master: der **Redundancy-Mode** ist nicht aktiv, die Einstellungen **SetPrm-Unlock before DP-Start-Up** sowie **SetPrm-Unlock at Shutdown** sollten deaktiviert werden, wenn ein Starten oder Stoppen des Primary-Masters rückwirkungsfrei auf die DP-Slaves sein soll (Ausgänge bleiben unverändert). Außerdem muss der **Watchdog** des Devices eingestellt werden (TwinCAT 2.8: s. Karteireiter [FC310x](#) [► 41] des Masters, TwinCAT 2.9: s. Dialog [Fault-Settings](#) [► 52]), damit ein Absturz des PCs dazu führt, dass der Primary-Master sich vom Bus abmeldet.

Redundancy-Master: der **Redundanz-Mode** ist aktiv, die Einstellungen **SetPrm-Unlock before DP-Start-Up** sowie **SetPrm-Unlock at Shutdown** sollten deaktiviert werden, wenn ein Starten oder Stoppen des Primary-Masters rückwirkungsfrei auf die DP-Slaves sein soll (Ausgänge bleiben unverändert).

Als Schnittstelle zum PC gibt es zusätzlich drei **Counter** und ein **StartRedundancyMasterFlag**:

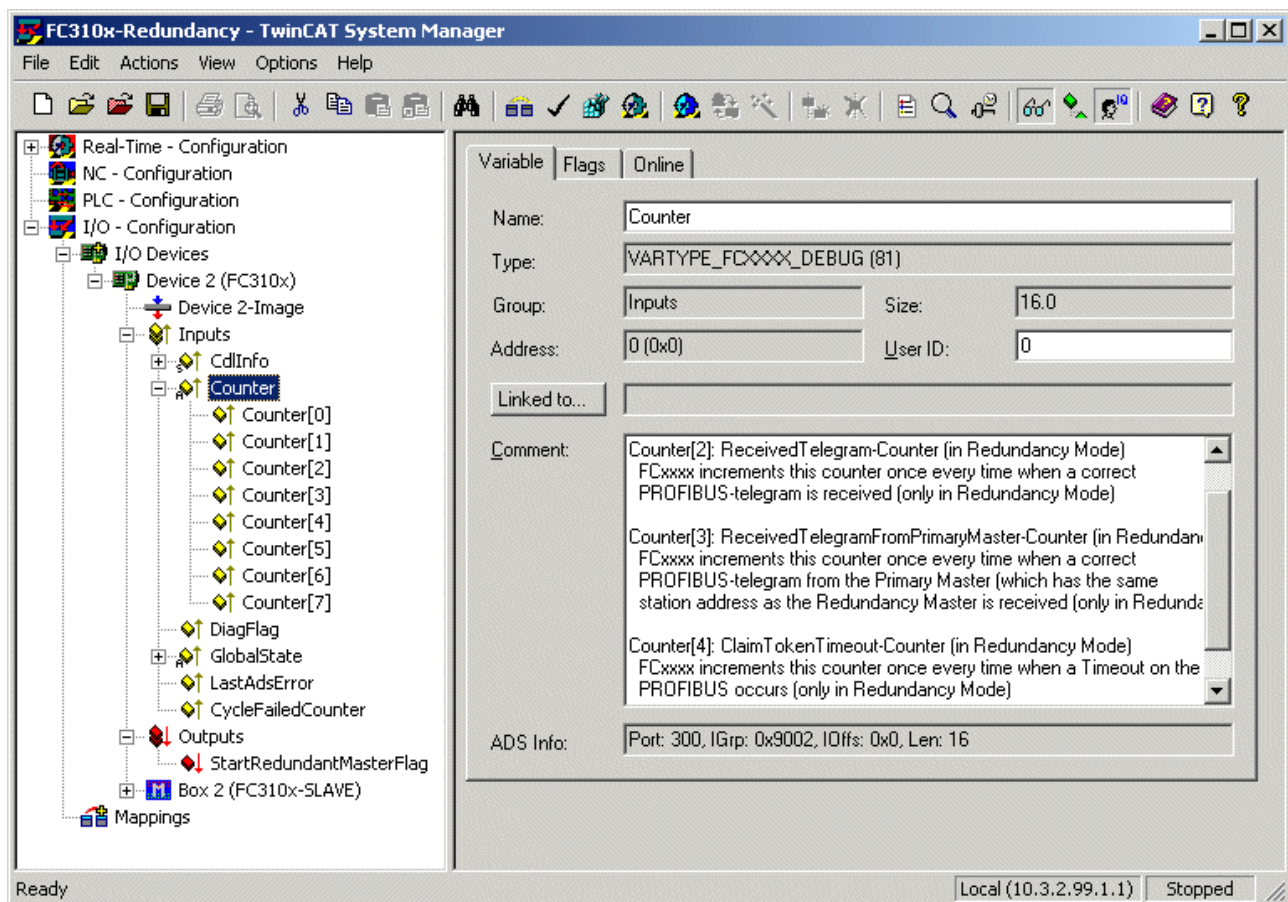


Abb. 19: Counter und StartRedundancyMasterFlag

Counter[2] (ReceivedTelegram-Counter): Dieser Counter zählt jedes Mal hoch, wenn ein gültiges PROFIBUS-Telegramm empfangen wurde

Counter[3] (ReceivedTelegramFromPrimary-Counter): Dieser Counter zählt jedes Mal hoch, wenn ein gültiges PROFIBUS-Telegramm von dem Primary-Master (der die gleiche Stationsadresse wie der Redundancy-Master hat) empfangen wurde

Counter[4] (ClaimTokenTimeout-Counter): Dieser Counter zählt jedes Mal hoch, wenn der Redundancy-Master einen Timeout auf dem Bus erkennt, nachdem er im Normalfall (**Redundancy-Mode** deaktiviert) die Busaktivität übernehmen würde (ClaimTokenTimeout-Time = $(6 + 2 * \text{Stationsadresse des DP-Master}) * \text{Slot-Time}$).

StartRedundancyMasterFlag: Damit kann der Redundancy-Master gestartet bzw. gestoppt werden.

Die Anwendung (SPS-Task oder anderes Programm) ist also selbst dafür verantwortlich, den Ausfall des Primary-Masters zu diagnostizieren (z. B. dadurch, dass der **ReciveTelegram-Counter** und der **ReceivedTelegramFromPrimaryMaster-Counter** nicht mehr hochzählen, der **ClaimTokenTimeout-Counter** hochzählt oder die anwendungsspezifische Überwachung der beiden PCs zuschlägt), der Redundancy-Master wird erst am Bus aktiv, wenn das **StartRedundancyMaster-Flag** gesetzt ist (das Starten dauert ca. 10 x das Min-Slave-Int. (TwinCAT 2.8: s. Karteireiter PROFIBUS [▶ 43] des Masters, TwinCAT 2.9: s. Dialog Bus-Parameter [▶ 50]), wenn das **StartRedundancyMaster-Flag** wieder zurückgesetzt wird, beendet der Redundancy-Master beim nächsten Senden eines Tokens (am Ende des DP-Zyklus, maximal nach der **Estimated Cycle Time** (s. Karteireiter "**FC310x**" (für TwinCAT 2.8 [▶ 41] bzw. TwinCAT 2.9 [▶ 49]) des Devices)) seine Busaktivität, ohne die Verbindung zu den Slaves zu abzubauen (unabhängig von der Einstellung **SetPrm-Unlock at Shutdown**).

Bei der Einstellung des DP-Watchdogs der DP-Slaves (s. Karteireiter PROFIBUS [▶ 59] der Box) ist zu beachten, dass die DP-Watchdog-Zeit größer als die Überwachungszeit des Primary-Masters der Anwendung plus die Start-Up-Zeit des Redundancy-Masters ist, falls die Übernahme der DP-Slaves durch den Redundancy-Masters rückwirkungsfrei erfolgen soll.

Weiterhin aktualisiert der Redundancy-Master keine Prozessdaten, solange er am Bus nur mithört, wenn er gestartet wird, sollte der DpState [▶ 26] der Boxen ausgewertet werden, wenn dieser 0 ist, sind die Prozessdaten auch aktuell.

3.8 Karteireiter Device

3.8.1 TwinCAT 2.8

3.8.1.1 Karteireiter FC310x

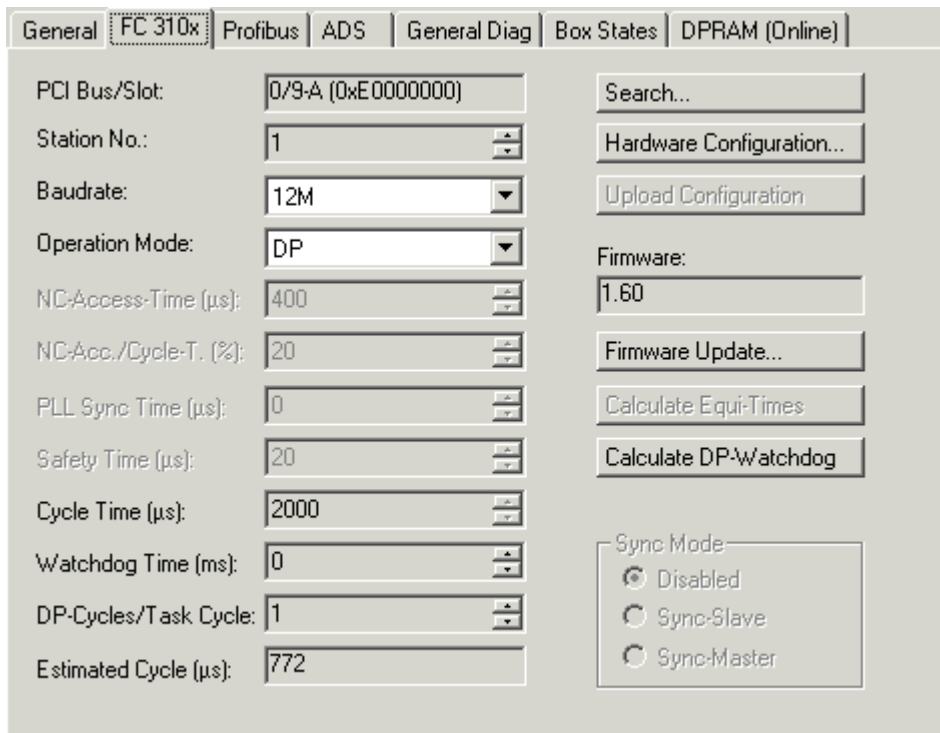


Abb. 20: TwinCAT 2.8 - Karteireiter FC310

PCI Slot/Irq: Zeigt an in welchem logischen PCI-Slot die Karte gefunden wurde.

Search...: Hierüber werden alle gesteckten FC310x-Kanäle gesucht, und es kann der gewünschte ausgewählt werden. Bei einer FC3102 erscheinen beide Kanäle A und B, die sich logisch wie zwei FC3101-Karten verhalten.

Hardware Configuration...: Hier kann die Hardware-Versionsnummer der FC310x angezeigt werden

Upload Configuration...: Hiermit wird der PROFIBUS gescannt und alle gefundenen Geräte werden dem FC310x-Device hinzugefügt (bei TwinCAT 2.8 darf keine Box angefügt sein, ab TwinCAT 2.9 kann auch bei angefügten Boxen gescannt werden, die FC310x übernimmt dann die neue Konfiguration, zeigt aber Änderungen an). Bei Beckhoff-Boxen wird die Konfiguration genau ausgelesen, bei Fremdgeräten wird versucht, die zugehörige GSD-Datei zu finden.

Verify Configuration...: Hiermit wird der PROFIBUS gescannt und mit den aktuell angefügten Boxen verglichen, Änderungen werden angezeigt (ab TwinCAT 2.9).

Firmware: Hier wird die aktuelle Firmware-Version der FC310x angezeigt.

Firmware Update...: Hierüber kann die Firmware der FC310x-Karte aktualisiert werden.

Stations-No.: Jeder Profibusteilnehmer benötigt eine eindeutige Stations-Nr. - auch der Master.

Baudrate: Hier wird die Profibus Baudrate eingestellt.

Operation Mode: Bei allen drei Betriebsarten übernimmt die höchstprioräre Task, die mit dem entsprechenden Gerät verknüpft ist, die Ansteuerung des PROFIBUS-Zyklus und ist somit mit dem DP-Zyklus synchronisiert (s. Kapitel [Synchronisierung](#) [▶ 12]). Wenn diese Task gestoppt wird oder auf einen Breakpoint läuft, schaltet die FC310x in den CLEAR-Mode (Ausgänge gehen bei den Slaves auf 0 oder auf sichere Werte) (s. Kapitel [Fehlerreaktionen](#) [▶ 17]). Alle anderen Tasks werden asynchron über entsprechende Puffer bedient. Wenn eine dieser Tasks gestoppt wird oder auf einen Breakpoint läuft, kommt in der Regel im System Manager die Meldung, dass der Watchdog des entsprechenden asynchronen Mappings zugeschlagen hat und die entsprechenden Ausgänge werden auf 0 gesetzt. Bei allen Betriebsarten kann je Slave eine Poll-Rate eingestellt werden (im Karteireiter [Features](#) [▶ 60] der Box). Die Reihenfolge der Slaves im PROFIBUS-Zyklus entspricht der Reihenfolge, in der sie in dem Baum unter dem FC310x-Device hängen. Die Betriebsart "DP" ist für den Standard-DP-Betrieb, die Betriebsarten "DP/MC (Equidistant)" und "Equidistant (no GC)" sind im Kapitel [PROFIBUS-MC](#) [▶ 26] beschrieben, ebenso wie die Parameter **NC-Access-Time**, **Relation NC-Access-Time/Cycle-Time**, **PLL-Sync-Time** und **Safety-Time** sowie die **Sync-Mode**-Einstellungen und der Button **Calculate Equi-Times**, die nur für PROFIBUS MC von Bedeutung sind.

Cycle-Time: Hier wird die Zykluszeit der zugehörigen höchstpriorären Task angezeigt.

Estimated Cycle: Hier wird die voraussichtliche PROFIBUS-Zykluszeit angezeigt.

Watchdog Time: Hier kann ein Watchdog aktiviert werden, der dazu führt, dass bei einem Absturz des PCs die FC310x in den STOP-Zustand geht und den Datenaustausch mit allen projektierten Slaves beenden würde (s. [Fehlerreaktionen](#) [▶ 17]). Die Zeit ist wichtig im [Redundanz-Mode](#) [▶ 39] beim Primary-Master.

Calculate DP-Slave Watchdog Time: Hier wird die DP-Watchdog-Time aller DP-Slaves auf einen vernünftigen Wert gestellt, nach der Formel **Estimated-Cycle-Time * 10**

3.8.1.2 Karteireiter Profibus

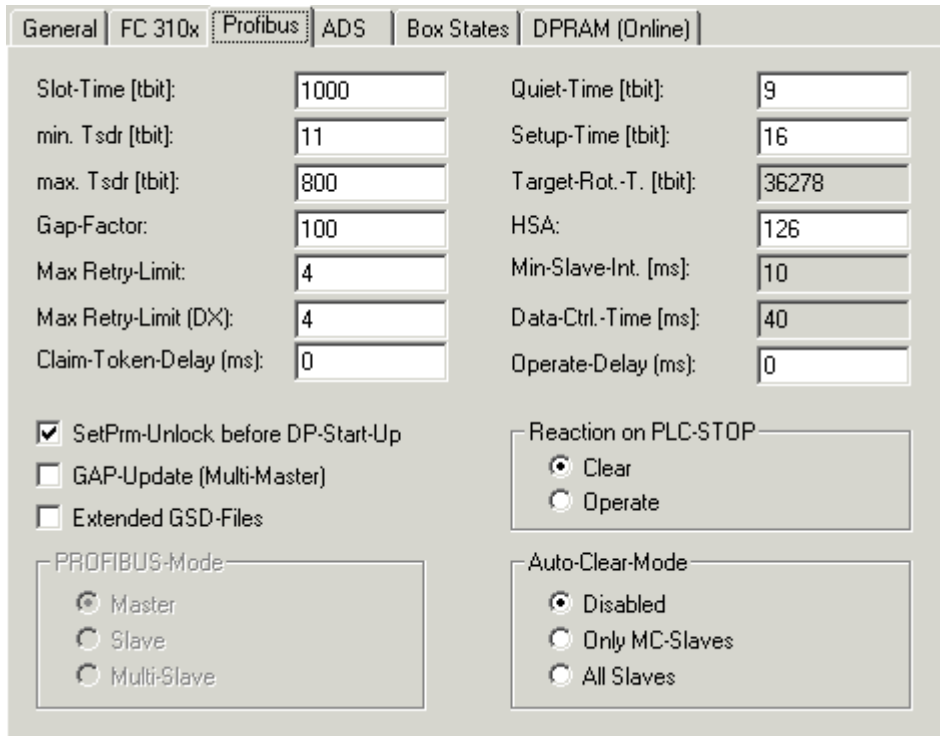


Abb. 21: TwinCAT 2.8 - Karteireiter Profibus

Slot-Time: Die Slot-Time gibt an, wie lange der DP-Master auf eine Antwort des DP-Slaves wartet, bevor er eine Wiederholung oder das nächste Telegramm sendet.

min. Tsd: Die min. Tsd gibt an, wie lange der DP-Slave mindestens mit einer Antwort wartet. Diese Zeit wird bei allen DP-Slaves während des DP-Start-Ups eingestellt (Wertebereich 11-255 Bit-Zeiten). Die min. Tsd muss kleiner als die max. Tsd sein.

max. Tsd: Die max. Tsd gibt an, wie lange der DP-Slave höchstens mit einer Antwort warten darf. Diese Zeit wird entsprechend der GSD-Datei-Einträge der DP-Slaves eingestellt. Die max. Tsd muss kleiner als die Slot-Time sein.

Max-Retry-Limit: Das Max-Retry-Limit gibt an, wie oft ein Telegramm zu wiederholen ist, wenn der adressierte Teilnehmer nicht antwortet. Hier sollte minimal ein Wert von 1 eingestellt werden, damit bei azyklischen Telegrammen im Fehlerfall mindestens einmal wiederholt wird (s. Kapitel [Fehlerreaktionen](#) [▶ 17]).

Max-Retry-Limit (DX): Da das Data_Exchange-Telegramm zyklisch wiederholt wird, könnte man für die Wiederholung des Data_Exchange-Telegrammes hier den Wert 0 eintragen, um im Equidistant-Betrieb den Zyklus auch bei einem nicht antwortenden Teilnehmer einigermaßen konstant zu halten. In diesem Fall sollte aber sinnvollerweise auf dem Karteireiter [Features](#) [▶ 60] der Box eingestellt werden, dass ein Nicht-Antworten des Slaves nicht zum Verlassen des DATA-EXCH führt. Dass ein Teilnehmer nicht geantwortet hat, kann man immer am [DpState](#) [▶ 26] erkennen, der dann für einen Zyklus auf ungleich 0 wäre (s. Kapitel [Fehlerreaktionen](#) [▶ 17]).

GAP-Update: Das GAP-Update, mit dem alle Stationen bis zur HSA ab und zu angefragt werden, ob sie vorhanden sind, kann ein- oder ausgeschaltet werden. Das GAP-Update ist eigentlich nur bei Multi-Master-Betrieb von Bedeutung, bei Single-Master-Betrieb hat es zur Folge, dass der Jitter des PROFIBUS-Zyklus größer ist, daher ist es defaultmäßig ausgeschaltet.

GAP-Factor: Der GAP-Factor bestimmt, wie häufig das GAP-Update durchgeführt wird (falls es aktiviert ist), der Abstand zwischen zwei GAP-Update-Zyklen ist **GAP-Factor * Target-Rot.-T.**

HSA: Die HSA bestimmt die höchste aktive Adresse, bis zu der das GAP-Update durchgeführt wird (falls es aktiviert ist).

Min. Slave-Int.: Das MinSlaveIntervall gibt an mit welcher minimalen Zykluszeit die DP-StartUp-Telegramme an die DP-Slaves gesendet werden (wird aus den Einstellungen in der GSD-Datei bestimmt).

PROFIBUS-Mode: Hier kann zwischen Master [► 10]-Funktionalität (Defaulteinstellung) und Slave [► 65] unterschieden werden.

Auto-Clear-Mode: Hier kann eingestellt werden, dass der Master in den Zustand "Clear" wechselt bzw. bleibt, solange mindestens ein MC-Slave (Einstellung "Only MC-Slaves") bzw. irgendein Slave (Einstellung "All Slaves") nicht korrekt antwortet (einen DpState [► 26] ungleich 0 hat) (s. Kapitel Fehlerreaktionen [► 17]).

Clear-Delay: Der DP-Master wechselt automatisch in den Clear-Mode (Ausgänge der Slaves werden auf 0 bzw. Fail_Safe-Werte gesetzt), wenn er keinen Interrupt mehr von der ihm zugehörigen Task bekommt (z. B. PLC-Breakpoint oder -Absturz). Hier kann angegeben werden, wie viele fehlende Task-Zyklen toleriert werden, bevor der Master in den Clear-Mode schaltet. Diese Einstellung ist unabhängig von der Einstellung des **Auto-Clear-Modes**.

Operate-Delay: Der DP-Master wechselt automatisch bei Beachtung des Auto-Clear-Modes in Operate-Zustand, wenn die Task gestartet wurde. Der Übergang von Clear nach Operate kann mit der Operate-Delay-Zeit verzögert werden. Im Zustand Clear werden alle Ausgänge auf 0 (DP-Slave unterstützt kein Fail_Safe) bzw. auf den Fail_Safe-Wert (DP-Slave unterstützt Fail_Safe) gesetzt, im Zustand Operate haben die Ausgänge den von der Task vorgegebenen Wert.

Reaction on PLC-STOP: Hier kann eingestellt werden, ob der DP-Master beim PLC-STOP/-Breakpoint die Ausgänge auf 0 setzen und unverändert lassen soll (s. Kapitel Fehlerreaktionen [► 17]).

Redundancy-Mode: Hier kann beim DP-Master der Redundancy-Mode eingestellt werden, in dem Fall hört er nur auf dem Bus mit (s. Kapitel Master-Redundanz [► 39]).

SetPrm-Unlock before DP-Start-Up: Normalerweise führt der DP-Master beim DP-Start-Up einen Verbindungsabbau der zyklischen Verbindung durch, damit der DP-Slave immer mitbekommt, dass der DP-Master neu gestartet wurde. Im Redundancy-Mode kann es aber gewünscht sein, dass der DP-Slave das gerade nicht mitbekommt, da das Umschalten vom Primary- auf den Redundancy-Master rückwirkungsfrei für den DP-Slave erfolgen soll (s. Kapitel Master-Redundanz [► 39]).

SetPrm-Unlock at DP-Shutdown: Normalerweise führt der DP-Master beim DP-Shutdown einen Verbindungsabbau der zyklischen Verbindung durch, damit der DP-Slave immer mitbekommt, dass der DP-Master gestoppt wurde. Im Redundancy-Mode kann es aber gewünscht sein, dass der DP-Slave das gerade nicht mitbekommt, da das Umschalten vom Primary- auf den Redundancy-Master rückwirkungsfrei für den DP-Slave erfolgen soll (s. Kapitel Master-Redundanz [► 39]).

3.8.1.3 Karteireiter ADS

Die FC310x ist ein ADS-Device mit einer eigenen Net-ID, die hier verändert werden kann. Alle ADS-Dienste (Diagnose, azyklische Kommunikation), die an die FC310x gehen, müssen diese Net-ID adressieren.

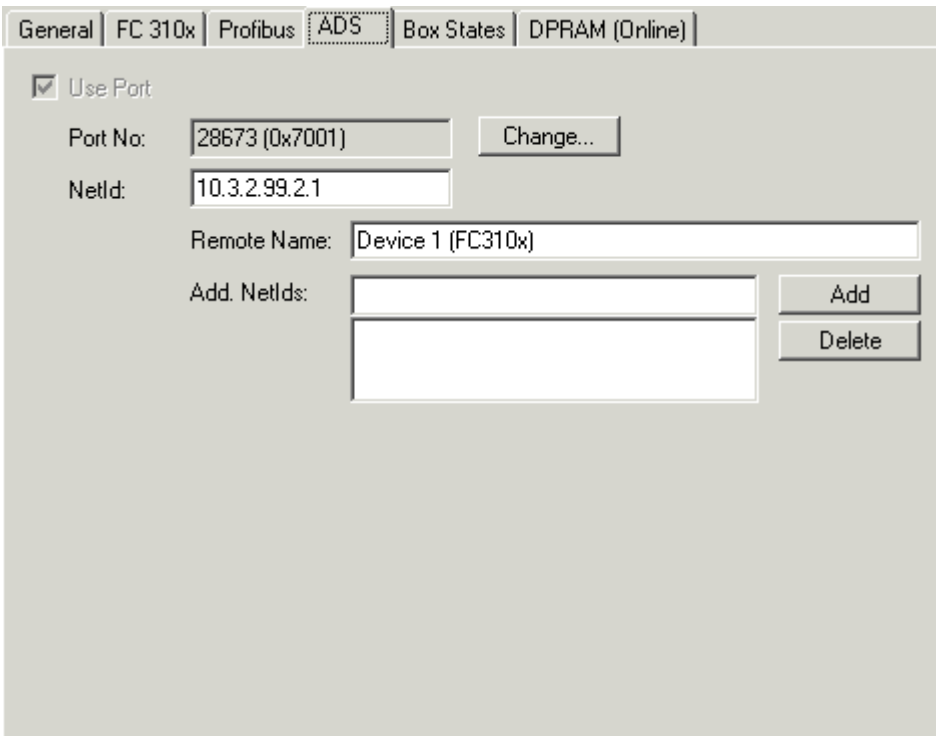


Abb. 22: TwinCAT 2.8 - Karteireiter ADS

3.8.1.4 Karteireiter General Diag

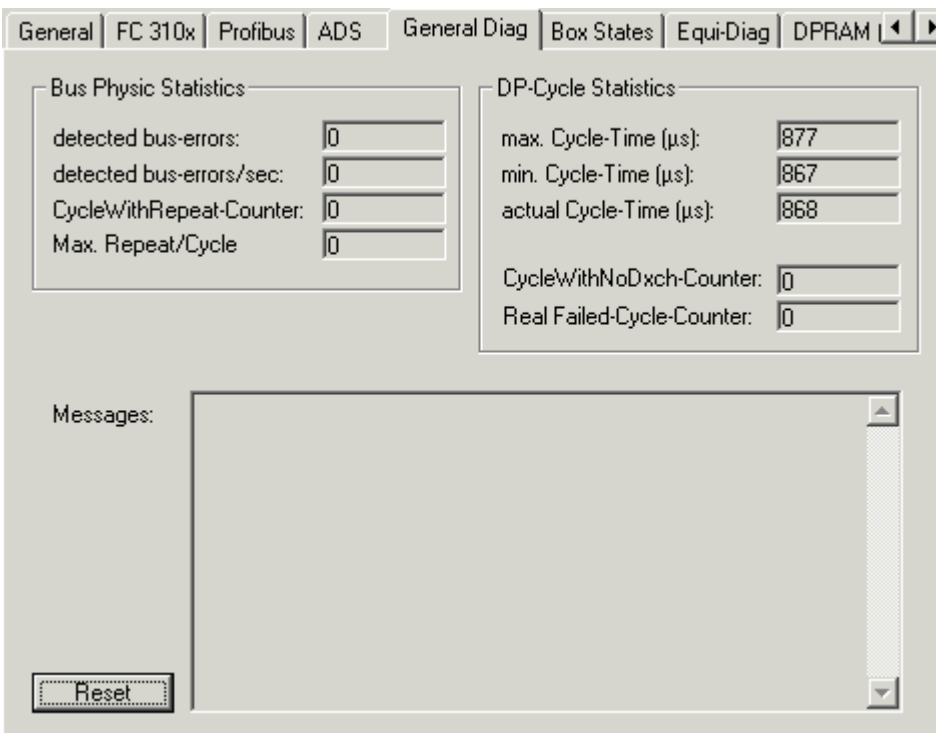


Abb. 23: TwinCAT 2.8 - Karteireiter General Diag

Hier werden Busverkabelungsprobleme und DP-Zykluszeiten angezeigt:

detected bus-errors: Hier wird die Anzahl der festgestellten Busfehler angezeigt, wenn dieser Zähler ungleich 0 ist, sollte die Verkabelung geprüft werden (falls keine PROFIBUS-Stecker abgezogen oder gesteckt wurden (in der Regel gibt es beim Abziehen oder Stecken von PROFIBUS-Steckern auch kurze Busstörungen)).

CycleWithRepeatCounter: Hier wird die Anzahl der PROFIBUS-Zyklen angezeigt, in denen mindestens einmal ein Telegramm wiederholt wurde. Wiederholungen sind auch ein Anzeichen dafür, dass die Busphysik nicht völlig okay ist.

max. Repeat/Cycle: Hier die maximale Anzahl von Wiederholungen innerhalb eines Zyklus angezeigt.

min./max./actual Cycle-Time: Hier werden minimale, maximale und aktuelle DP-Zykluszeit angezeigt, wobei nur Zyklen berücksichtigt werden, in denen alle Slaves im Datenaustausch waren und keine Wiederholungen auftraten.

CycleWithNoDxch-Counter: zählt hoch, wenn nicht alle Slaves im Datenaustausch sind (d.h. einen DpState ungleich 0 haben).

Real Failed-Cycle-Counter: zählt hoch, wenn der DP-Zyklus nicht fertig wurde, bevor der nächste Task-Zyklus gestartet ist und alle Slaves im Datenaustausch sind (d.h. einen DpState gleich 0 haben).

3.8.1.5 Karteireiter Box States

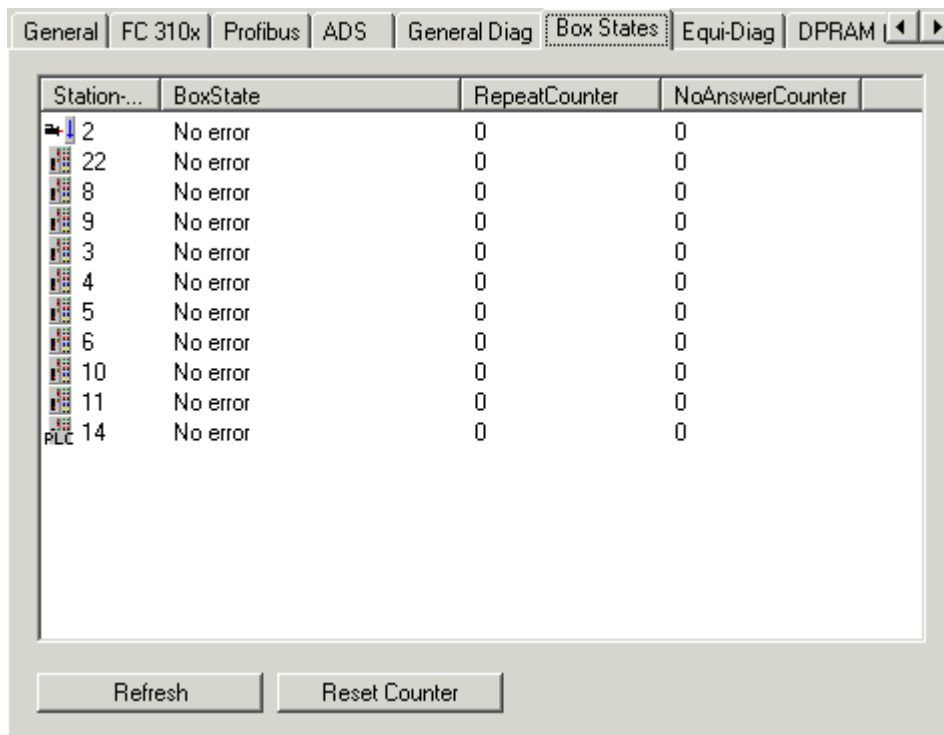


Abb. 24: TwinCAT 2.8 - Karteireiter Box States

Hier wird eine Übersicht aller aktuellen Box-States, der Repeat-Counter (zählt bei jeder Telegrammwiederholung zu dem Slave hoch) und der NoAnswer-Counter (zählt jedes Mal hoch, wenn der Slave nicht geantwortet hat) angezeigt.

3.8.1.6 Karteireiter EquiDiag

Es gibt den Karteireiter "EquiDiag", auf dem Online verschiedene Equidistant-Überwachungsparameter angezeigt werden:

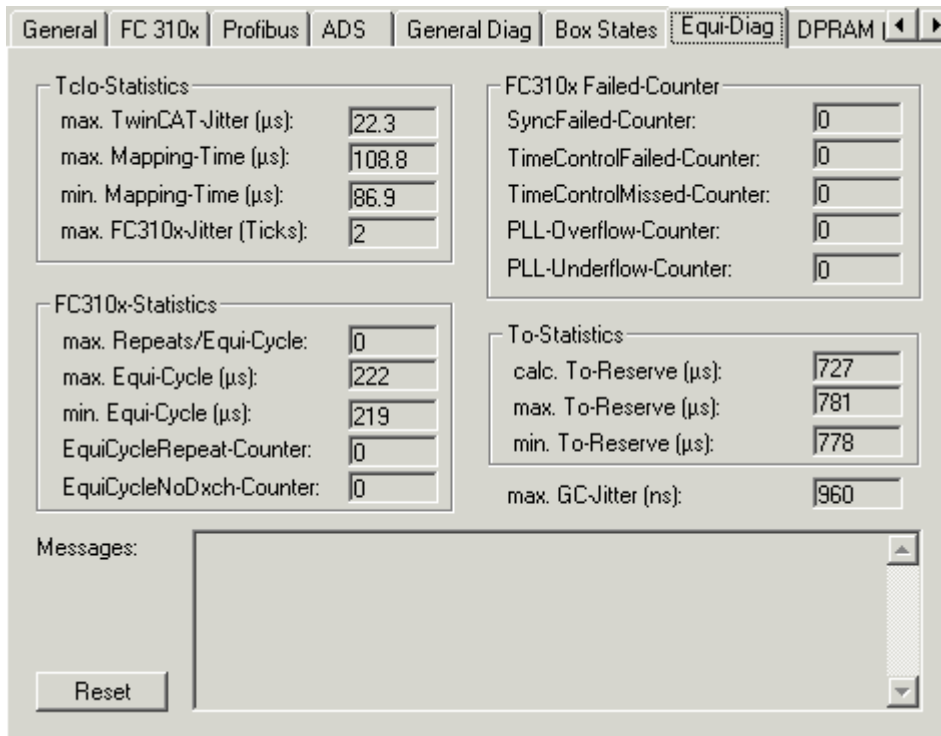


Abb. 25: TwinCAT 2.8 - Karteireiter EquiDiag

Tc10-Statistics

max. TwinCAT-Jitter: Hier wird der maximale TwinCAT-Jitter angezeigt, die Messung wird zurückgesetzt, sobald der Karteireiter nicht mehr aktiv ist.

min./max. Mapping-Time bzw. NC-Access-Time: Hier die minimale bzw. maximale NC-Access-Time angezeigt, wobei diese neben der Mapping-Zeit noch die Task-Laufzeit enthält (falls "IO am Taskanfang" bei der höchstprioriten mit der FC310x verknüpften Task (in der Regel die NC-Task) gesetzt ist).

Die NC-Access-Time sollte größer als die Addition dieser beiden gemessenen Zeiten (max. TwinCAT-Jitter plus max. Mapping-Time bzw. (NC-)Task-Laufzeit) mit einem Sicherheitsaufschlag (ca. 10%) eingestellt werden.

max. FCxxxx-Jitter: Ist nur bei Sync-Mode = "Disabled" von Bedeutung, wenn der Wert größer als 5 wird, jittert die Echtzeit zu stark, es sollte dann ein leistungsfähigerer PC genommen werden. Falls die zugehörige (NC-)Task nicht die höchste Priorität hat, kann der max. FCxxxx-Jitter auch größer werden, das sollte vermieden werden.

FC310x-Statistics

Da die PROFIBUS MC-Slaves immer zuerst an das FC310x-Device angefügt werden sollten, werden diese auch immer vor den DP-Slaves im DP-Zyklus angesprochen, dieser Teil des DP-Zyklus wird im Folgenden als Equi-Cycle bezeichnet. Wenn der Equi-Cycle größer als die To-Zeit der MC-Slaves wird, bekommen die letzten MC-Slaves am FC310x-Device in der Regel einen Synchronisationsfehler (Fehler 597 oder 598 bei Simodrive 611U).

max. Repeats/Equi-Cycle: Hier wird angezeigt, wie viele Data_Exchange-Telegramme während eines Equi-Cycles maximal wiederholt wurden, dadurch verlängert sich der Equi-Cycle, normalerweise sollten keine Wiederholungen auftreten (es sein denn, dass Busstecker gezogen oder ein MC-Slave ausgeschaltet wurde).

min./max. Equi-Cycle: Hier wird die minimale bzw. die maximale Equi-Cycle-Time angezeigt.

Equi-Cycle-Repeat-Counter: Hier wird angezeigt, wie oft im Equi-Cycle Telegramm-Wiederholungen auftraten.

Equi-Cycle-NoDxch-Counter: Hier wird angezeigt, wie oft im Equi-Cycle nicht alle MC-Slaves im Datenaustausch waren.

FC310x-Failed-Counter

Sync-Failed-Counter: Dieser Zähler zählt hoch, wenn TwinCAT-Task und DP-Zyklus nicht miteinander synchronisiert sind. Das kann beim Hochlauf des TwinCAT Systems passieren, danach sollte dieser Zähler nicht hochzählen. Falls die zugehörige (NC-)Task nicht die höchste Priorität hat, kann dieser Zähler auch hochzählen, das sollte vermieden werden.

Time-Control-Failed-Counter: Dieser Zähler zählt hoch, wenn zum Zeitpunkt des DP-Zyklus-Starts der PROFIBUS nicht frei war. Ursache dafür können Busstörungen, nicht vorhandenen Teilnehmer, ein zweiter Master oder eine zu kleine Safety-Time sein.

PLL-Overflow-/Underflow-Counter: Dieser Zähler ist nur beim Sync-Mode "Disabled" relevant und zählt hoch, falls die TwinCAT-Task, auf die sich der DP-Zyklus synchronisiert, zu stark jittert (kann z. B. passieren, falls der DP-Zyklus nicht mit der höchstpriorären Task synchronisiert ist). Falls die zugehörige (NC-)Task nicht die höchste Priorität hat, kann dieser Zähler auch hochzählen, das sollte vermieden werden.

To-Statistics

Mit der To-Zeit wird bei jedem MC-Slave festgelegt, wann, bezogen auf den DP-Zyklusstart, dieser die vom Master empfangenen Outputs übernehmen soll. Die MC-Slaves können miteinander synchronisiert werden, wenn bei allen MC-Slaves der gleiche Wert für To eingestellt wird. Dieser muss aber mindestens so groß wie die Equi-Cycle-Time plus eine Sicherheitsreserve von ca. 200 µs sein. Die To-Zeit wird bei allen MC-Slaves mit dem Button "Calc. Equi-Times" (s.o.) berechnet.

calc. To-Reserve: Hier steht die kalkulierte To-Reserve (To-Zeit - Equi-Cycle-Time)

min./max. To-Reserve: Hier wird die min. bzw. max. To-Reserve gemessen.

max. GC-Jitter (ab TwinCAT 2.8)

Hier wird der maximale Jitter des DP-Zyklus gemessen (GC für Global-Control-Telegramm, das immer am Anfang eines Zyklus gesendet wird). Beim Hochlauf kann der Jitter etwas größer sein, im eingeschwungenen Zustand sollte er 1 µs (bei Sync-Mode "Sync-Master") bzw. 2 µs (bei Sync-Mode "Disabled") nicht überschreiten.

3.8.2 TwinCAT 2.9

3.8.2.1 Karteireiter FC310x

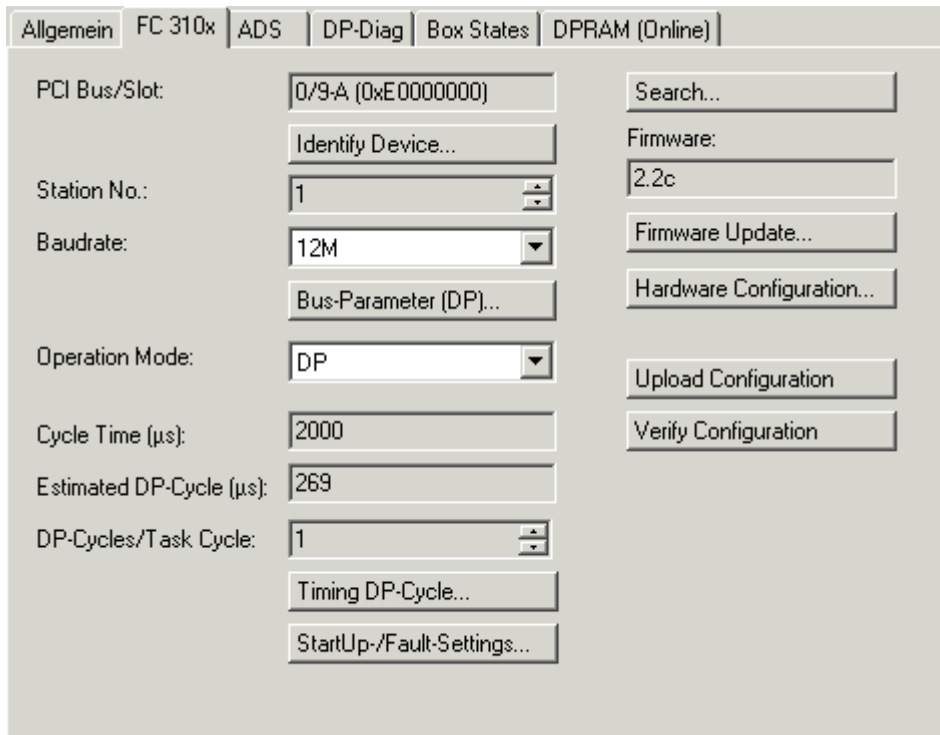


Abb. 26: TwinCAT 2.9 - Karteireiter FC310x

PCI Slot/Irq: Zeigt an in welchem logischen PCI-Slot die Karte gefunden wurde.

Search...: Hierüber werden alle gesteckten FC310x-Kanäle gesucht, und es kann der gewünschte ausgewählt werden. Bei einer FC3102 erscheinen beide Kanäle A und B, die sich logisch wie zwei FC3101-Karten verhalten.

Identify Device...: Hier kann ein LED-Code auf dem entsprechenden FC310x-Kanal ausgegeben werden.

Stations-No.: Jeder Profibusteilnehmer benötigt eine eindeutige Stations-Nr. - auch der Master.

Baudrate: Hier wird die Profibus Baudrate eingestellt.

Bus-Parameter (DP)...: Hier wird der [Bus-Parameter-Dialog](#) [► 50] angewählt.

Operation Mode: Bei allen drei Betriebsarten übernimmt die höchstprioräre Task, die mit dem entsprechendem Gerät verknüpft ist, die Ansteuerung des PROFIBUS-Zyklus und ist somit mit dem DP-Zyklus synchronisiert (s. Kapitel [Synchronisierung](#) [► 12]). Wenn diese Task gestoppt wird oder auf einen Breakpoint läuft, schaltet die FC310x in den CLEAR-Mode (Ausgänge gehen bei den Slaves auf 0 oder auf sichere Werte) (s. Kapitel [Fehlerreaktionen](#) [► 17]). Alle anderen Tasks werden asynchron über entsprechende Puffer bedient. Wenn eine dieser Tasks gestoppt wird oder auf einen Breakpoint läuft, kommt in der Regel im System Manager die Meldung, dass der Watchdog des entsprechenden asynchronen Mappings zugeschlagen hat und die entsprechenden Ausgänge werden auf 0 gesetzt. Bei allen Betriebsarten kann je Slave eine Poll-Rate eingestellt werden (im Karteireiter [Features](#) [► 60] der Box). Die Reihenfolge der Slaves im PROFIBUS-Zyklus entspricht der Reihenfolge, in der sie in dem Baum unter dem FC310x-Device hängen. Die Betriebsart "DP" ist für den Standard-DP-Betrieb, die Betriebsarten "DP/MC (Equidistant)" und "Equidistant (no GC)" sind im Kapitel [PROFIBUS-MC](#) [► 26] beschrieben.

Cycle-Time: Hier wird die Zykluszeit der zugehörigen höchstpriorären Task angezeigt.

Estimated Cycle: Hier wird die voraussichtliche PROFIBUS-Zykluszeit angezeigt.

DP-Cycles/Task-Cycle: Hier können mehrere DP-Zyklen bei einem Task-Zyklus eingestellt werden, um möglichst neue Eingänge zu bekommen (s. Kapitel [Slave-Priorisierung/mehrere DP-Zyklen](#) [► 14]).

Timing DP-Cycle...: Hier wird das Timing des DP-Zyklus angezeigt, das ist insbesondere bei der Verwendung der [Slave-Priorisierung](#) [► 14] von Vorteil.

StartUp-/Fault-Settings...: Hier wird der [Fault-Settings-Dialog](#) [► 52] angewählt.

Firmware: Hier wird die aktuelle Firmware-Version der FC310x angezeigt.

Firmware Update...: Hierüber kann die Firmware der FC310x-Karte aktualisiert werden.

Hardware Configuration...: Hier kann die Hardware-Versionsnummer der FC310x angezeigt werden

Upload Configuration...: Hiermit wird der PROFIBUS gescannt und alle gefunden Geräte werden dem FC310x-Device hinzugefügt (bei TwinCAT 2.8 darf keine Box angefügt sein, ab TwinCAT 2.9 kann auch bei angefügten Boxen gescannt werden, die FC310x übernimmt dann die neue Konfiguration, zeigt aber Änderungen an). Bei Beckhoff-Boxen wird die Konfiguration genau ausgelesen, bei Fremdgeräten wird versucht, die zugehörige GSD-Datei zu finden.

Verify Configuration...: Hiermit wird der PROFIBUS gescannt und mit den aktuell angefügten Boxen verglichen, Änderungen werden angezeigt (ab TwinCAT 2.9).

3.8.2.2 Dialog Bus-Parameter

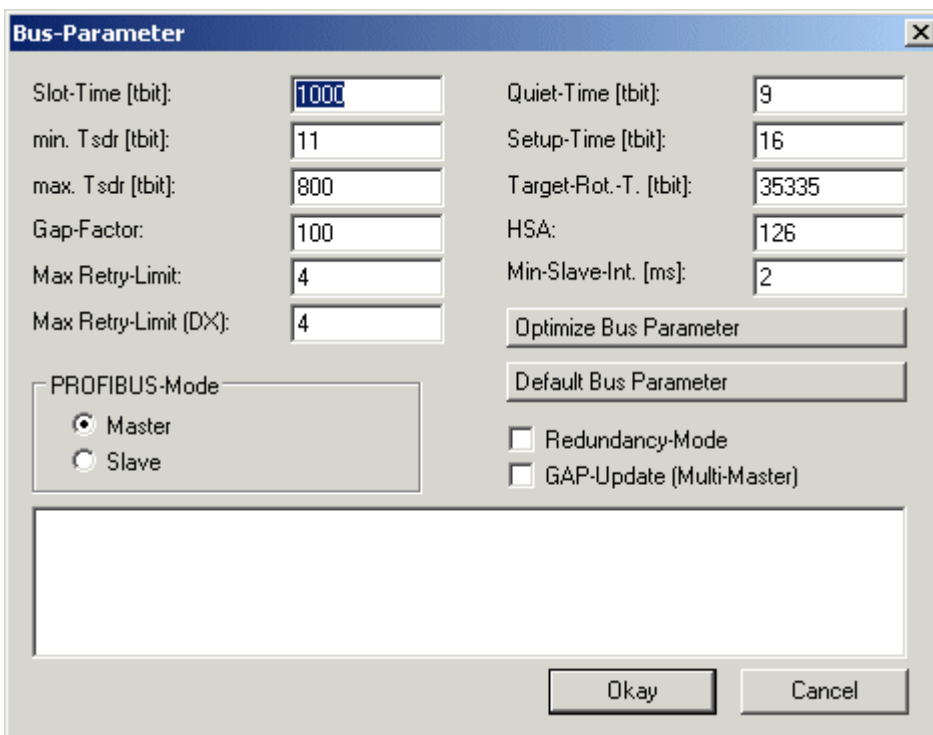


Abb. 27: TwinCAT 2.9 - Dialog Bus-Parameter

Slot-Time: Die Slot-Time gibt an, wie lange der DP-Master auf eine Antwort des DP-Slaves wartet, bevor er eine Wiederholung oder das nächste Telegramm sendet.

min. Tsdr: Die min. Tsdr gibt an, wie lange der DP-Slave mindestens mit einer Antwort wartet. Diese Zeit wird bei allen DP-Slaves während des DP-Start-Ups eingestellt (Wertebereich 11-255 Bit-Zeiten). Die min. Tsdr muss kleiner als die max. Tsdr sein.

max. Tsdr: Die max. Tsdr gibt an, wie lange der DP-Slave höchstens mit einer Antwort warten darf. Diese Zeit wird entsprechend der GSD-Datei-Einträge der DP-Slaves eingestellt. Die max. Tsdr muss kleiner als die Slot-Time sein.

GAP-Factor: Der GAP-Factor bestimmt, wie häufig das GAP-Update durchgeführt wird (falls es aktiviert ist), der Abstand zwischen zwei GAP-Update-Zyklen ist **Gap-Factor * Target-Rot.-T.**

Max-Retry-Limit: Das Max-Retry-Limit gibt an, wie oft ein Telegramm zu wiederholen ist, wenn der adressierte Teilnehmer nicht antwortet. Hier sollte minimal ein Wert von 1 eingestellt werden, damit bei azyklischen Telegrammen im Fehlerfall mindestens einmal wiederholt wird (s. Kapitel [Fehlerreaktionen](#) [[▶ 17](#)]).

Max-Retry-Limit (DX): Da das Data_Exchange-Telegramm zyklisch wiederholt wird, könnte man für die Wiederholung des Data_Exchange-Telegrammes hier den Wert 0 eintragen, um im Equidistant-Betrieb den Zyklus auch bei einem nicht antwortenden Teilnehmer einigermaßen konstant zu halten. In diesem Fall sollte aber sinnigerweise auf dem Karteireiter [Features](#) [[▶ 60](#)] der Box eingestellt werden, dass ein Nicht-Antworten des Slaves nicht zum Verlassen des DATA-EXCH führt. Dass ein Teilnehmer nicht geantwortet hat, kann man immer am [DpState](#) [[▶ 26](#)] erkennen, der dann für einen Zyklus auf ungleich 0 wäre (s. Kapitel [Fehlerreaktionen](#) [[▶ 17](#)]).

HSA: Die HSA bestimmt die höchste aktive Adresse, bis zu der das GAP-Update durchgeführt wird (falls es aktiviert ist).

Min. Slave-Int.: Das MinSlaveIntervall gibt an mit welcher minimalen Zykluszeit die DP-StartUp-Telegramme an die DP-Slaves gesendet werden (wird aus den Einstellungen in der GSD-Datei bestimmt).

PROFIBUS-Mode: Hier kann zwischen [Master](#) [[▶ 10](#)]-Funktionalität (Defaulteinstellung) und [Slave](#) [[▶ 65](#)] unterschieden werden.

Redundancy-Mode: Hier kann beim DP-Master der Redundancy-Mode eingestellt werden, in dem Fall hört er nur auf dem Bus mit (s. Kapitel [Master-Redundanz](#) [[▶ 39](#)]).

GAP-Update: Das GAP-Update, mit dem alle Stationen bis zur HSA ab und zu angefragt werden, ob sie vorhanden sind, kann ein- oder ausgeschaltet werden. Das GAP-Update ist eigentlich nur bei Multi-Master-Betrieb von Bedeutung, bei Single-Master-Betrieb hat es zur Folge, dass der Jitter des PROFIBUS-Zyklus größer ist, daher ist es defaultmäßig ausgeschaltet.

Optimize Bus-Parameter: Hiermit werden die optimierten Bus-Parameter eingestellt.

Default Bus-Parameter: Hiermit werden die Default-Bus-Parameter eingestellt.

3.8.2.3 Dialog Fault-Settings

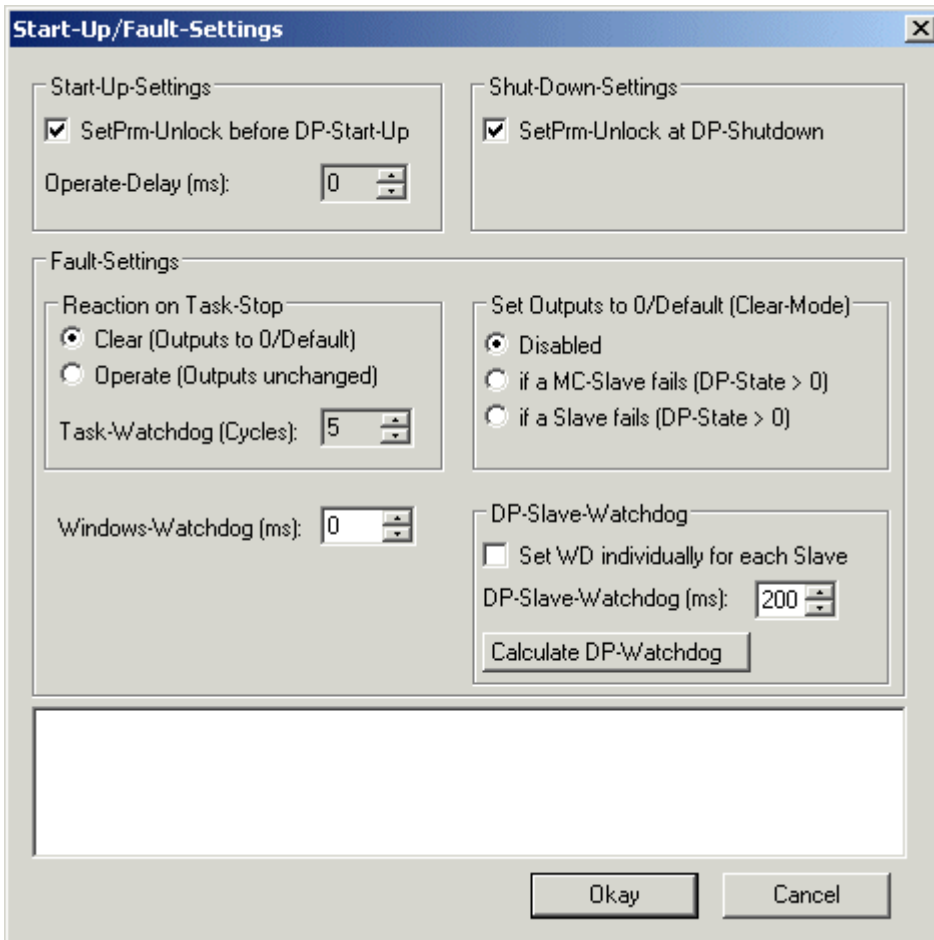


Abb. 28: TwinCAT 2.9 - Dialog Fault-Settings

SetPrm-Unlock before DP-Start-Up: Normalerweise führt der DP-Master beim DP-Start-Up einen Verbindungsabbau der zyklischen Verbindung durch, damit der DP-Slave immer mitbekommt, dass der DP-Master neu gestartet wurde. Im Redundancy-Mode kann es aber gewünscht sein, dass der DP-Slave das gerade nicht mitbekommt, da das Umschalten vom Primary- auf den Redundancy-Master rückwirkungsfrei für den DP-Slave erfolgen soll (s. Kapitel [Master-Redundanz](#) [► 39]).

SetPrm-Unlock at DP-Shutdown: Normalerweise führt der DP-Master beim DP-Shutdown einen Verbindungsabbau der zyklischen Verbindung durch, damit der DP-Slave immer mitbekommt, dass der DP-Master gestoppt wurde. Im Redundancy-Mode kann es aber gewünscht sein, dass der DP-Slave das gerade nicht mitbekommt, da das Umschalten vom Primary- auf den Redundancy-Master rückwirkungsfrei für den DP-Slave erfolgen soll (s. Kapitel [Master-Redundanz](#) [► 39]).

Operate-Delay: Der DP-Master wechselt automatisch bei Beachtung des Auto-Clear-Modes in Operate-Zustand, wenn die Task gestartet wurde. Der Übergang von Clear nach Operate kann mit der Operate-Delay-Zeit verzögert werden. Im Zustand Clear werden alle Ausgänge auf 0 (DP-Slave unterstützt kein Fail_Safe) bzw. auf den Fail_Safe-Wert (DP-Slave unterstützt Fail_Safe) gesetzt, im Zustand Operate haben die Ausgänge den von der Task vorgegebenen Wert.

Reaction on Task-STOP: Hier kann eingestellt werden, ob der DP-Master beim PLC-STOP/-Breakpoint die Ausgänge auf 0 setzen und unverändert lassen soll (s. Kapitel [Fehlerreaktionen](#) [► 17]).

Task-Watchdog: Der DP-Master wechselt automatisch in den Clear-Mode (Ausgänge der Slaves werden auf 0 bzw. Fail_Safe-Werte gesetzt), wenn er keinen Interrupt mehr von der ihm zugehörigen Task bekommt (z. B. PLC-Breakpoint oder -Absturz). Hier kann angegeben werden, wie viele fehlende Task-Zyklen toleriert werden, bevor der Master in den Clear-Mode schaltet. Diese Einstellung ist unabhängig von der Einstellung des **Clear-Modes**.

Clear-Mode: Hier kann eingestellt werden, dass der Master in den Zustand "Clear" wechselt bzw. bleibt, solange mindestens ein MC-Slave (Einstellung "Only MC-Slaves") bzw. irgendein Slave (Einstellung "All Slaves") nicht korrekt antwortet (einen `DpState` [► 26] ungleich 0 hat) (s. Kapitel Fehlerreaktionen [► 17]).

Windows-Watchdog: Hier kann ein Watchdog aktiviert werden, der dazu führt, dass bei einem Absturz des PCs die FC310x in den STOP-Zustand geht und den Datenaustausch mit allen projektierten Slaves beenden würde (s. Fehlerreaktionen [► 17]). Die Zeit ist wichtig im **Redundanz-Mode** [► 39] beim Primary-Master.

Set WD individually for each Slave: Hier angewählt werden, ob der WD individuell für jeden Slave eingestellt werden soll (auf dem Karteireiter **Profibus** [► 59] der Box)

DP-Watchdog-Time: Wenn die Check-Box "Set WD individually for each Slave" nicht angeklickt ist, kann hier der DP-Watchdog auf einen einheitlichen Wert für alle Slaves eingestellt werden.

Calculate DP-Slave Watchdog Time: Hier wird die DP-Watchdog-Time aller DP-Slaves auf einen vernünftigen Wert gestellt.

3.8.2.4 Karteireiter MC

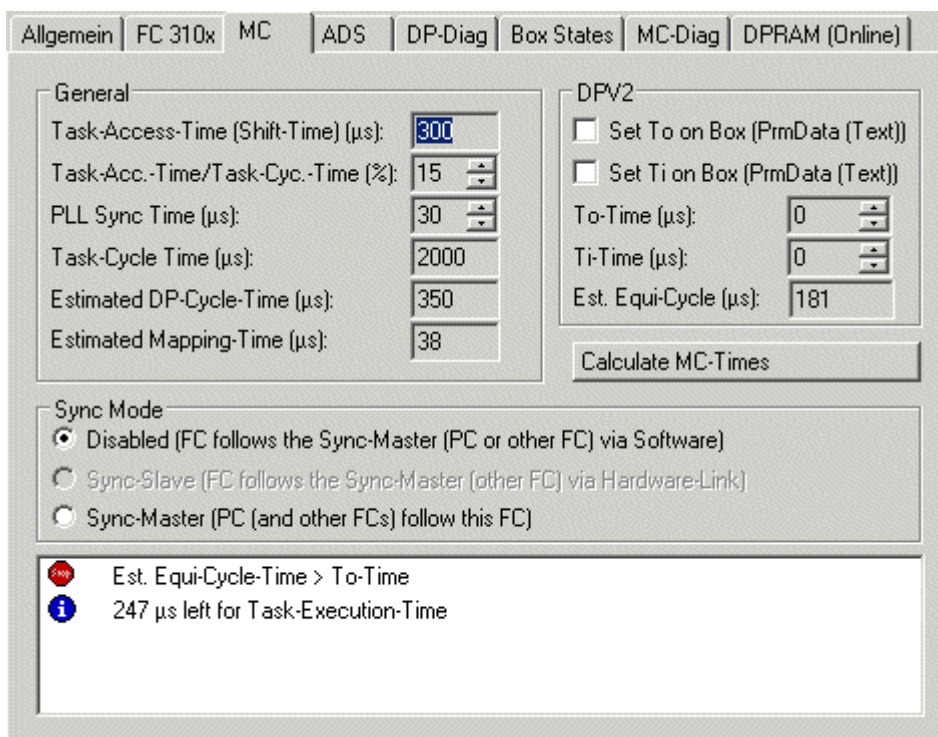


Abb. 29: TwinCAT 2.9 - Karteireiter MC

Task-Access-Time (Shift-Time): Diese Zeit muss größer als der maximale TwinCAT-Jitter plus die maximale Mapping-Zeit sein (s. Kapitel PROFIBUS MC [► 26])

Task-Access-Time/Task Cycle-Time: Dieses Verhältnis ist sinnvoll, wenn bei der Benutzung des Buttons **Calculate MC-Times** vermieden werden soll, dass die Task-Access-Time manuell angepasst werden soll.

PLL-Sync-Time: nur für den **Sync-Mode Disabled** von Bedeutung, stellt das PLL-Fenster auf der FC310x ein

Task Cycle Time: Hier wird die Zykluszeit der zugehörigen höchstpriorären Task angezeigt

Estimated DP-Cycle Time: Hier wird die voraussichtliche PROFIBUS-Zykluszeit angezeigt.

Estimated Mapping Time: Hier wird die voraussichtliche Mapping-Zeit angezeigt.

Set To on Box: Die To-Time kann für jeden Slave individuell auf dem Karteireiter **Prm Data(Text)** [► 63] der Box eingestellt werden

Set Ti on Box: Die Ti-Time kann für jeden Slave individuell auf dem Karteireiter Prm Data(Text) [▶ 63] der Box eingestellt werden

To-Time: Wenn die Check-Box **Set To on Box** nicht aktiviert ist, kann die To-Time für alle Slaves einheitlich eingestellt werden.

Ti-Time: Wenn die Check-Box **Set Ti on Box** nicht aktiviert ist, kann die Ti-Time für alle Slaves einheitlich eingestellt werden

Estimated Equi-Cycle Time: Hier wird der DPV2-Teil der voraussichtlichen PROFIBUS-Zykluszeit angezeigt.

Calculate MC-Times: Mit diesem Button werden alle DPV2-Zeiten automatisch berechnet

Sync-Mode: Der Sync-Mode entscheidet, ob die FC dem PC folgt (Disabled) oder ob der PC der FC folgt (Master), um den TwinCAT- mit dem PROFIBUS-Zyklus zu synchronisieren

3.8.2.5 Karteireiter ADS

Die FC310x ist ein ADS-Device mit einer eigenen Net-ID, die hier verändert werden kann. Alle ADS-Dienste (Diagnose, azyklische Kommunikation), die an die FC310x gehen, müssen diese Net-ID adressieren.

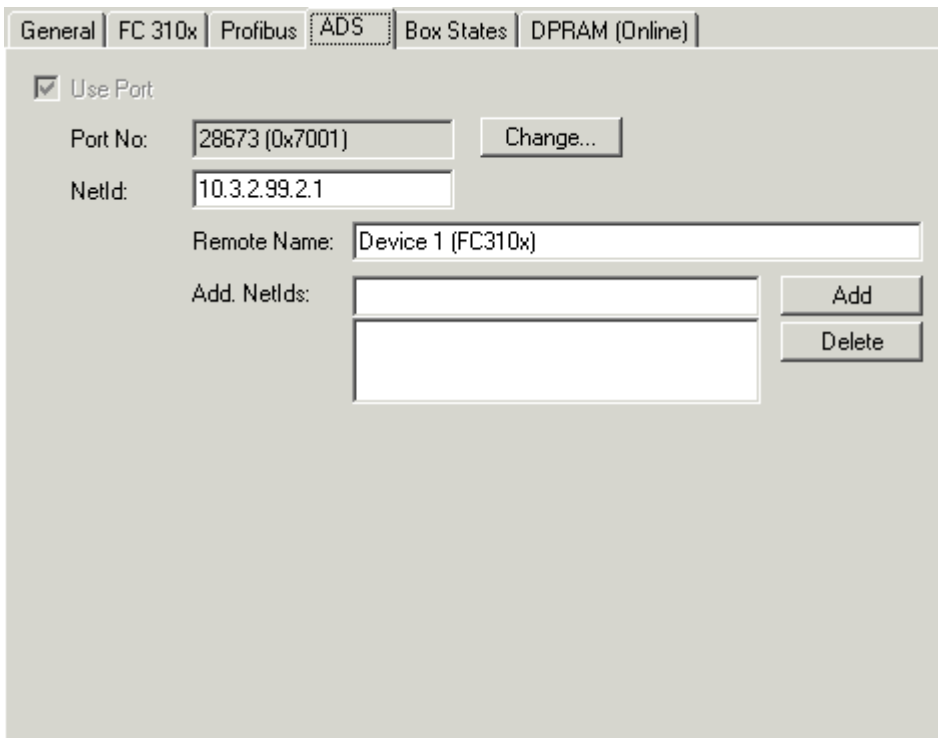


Abb. 30: TwinCAT 2.9 - Karteireiter ADS

3.8.2.6 Karteireiter DP-Diag

Allgemein	FC 310x	ADS	DP-Diag	Box States	DPRAM (Online)																																
<table border="1"> <thead> <tr> <th colspan="2">Bus Physic Statistics</th> <th colspan="2">DP-Cycle Statistics</th> </tr> </thead> <tbody> <tr> <td>detected bus-errors:</td> <td><input type="text"/></td> <td>max. Cycle-Time (µs):</td> <td><input type="text" value="361"/></td> </tr> <tr> <td>detected bus-errors/sec:</td> <td><input type="text"/></td> <td>min. Cycle-Time (µs):</td> <td><input type="text" value="187"/></td> </tr> <tr> <td>Last detected bus error:</td> <td><input type="text"/></td> <td>actual Cycle-Time (µs):</td> <td><input type="text" value="203"/></td> </tr> <tr> <td>CycleWithRepeat-Counter:</td> <td><input type="text" value="0"/></td> <td>CycleWithNoDxch-Counter:</td> <td><input type="text" value="169"/></td> </tr> <tr> <td>Max. Repeat/Cycle</td> <td><input type="text" value="0"/></td> <td>Last cycle with no Dxch:</td> <td><input type="text" value="18:27:05"/></td> </tr> <tr> <td>Last Repeat:</td> <td><input type="text"/></td> <td>Failed-Cycle-Counter:</td> <td><input type="text" value="0"/></td> </tr> <tr> <td></td> <td></td> <td>Last Failed-Cycle:</td> <td><input type="text"/></td> </tr> </tbody> </table>						Bus Physic Statistics		DP-Cycle Statistics		detected bus-errors:	<input type="text"/>	max. Cycle-Time (µs):	<input type="text" value="361"/>	detected bus-errors/sec:	<input type="text"/>	min. Cycle-Time (µs):	<input type="text" value="187"/>	Last detected bus error:	<input type="text"/>	actual Cycle-Time (µs):	<input type="text" value="203"/>	CycleWithRepeat-Counter:	<input type="text" value="0"/>	CycleWithNoDxch-Counter:	<input type="text" value="169"/>	Max. Repeat/Cycle	<input type="text" value="0"/>	Last cycle with no Dxch:	<input type="text" value="18:27:05"/>	Last Repeat:	<input type="text"/>	Failed-Cycle-Counter:	<input type="text" value="0"/>			Last Failed-Cycle:	<input type="text"/>
Bus Physic Statistics		DP-Cycle Statistics																																			
detected bus-errors:	<input type="text"/>	max. Cycle-Time (µs):	<input type="text" value="361"/>																																		
detected bus-errors/sec:	<input type="text"/>	min. Cycle-Time (µs):	<input type="text" value="187"/>																																		
Last detected bus error:	<input type="text"/>	actual Cycle-Time (µs):	<input type="text" value="203"/>																																		
CycleWithRepeat-Counter:	<input type="text" value="0"/>	CycleWithNoDxch-Counter:	<input type="text" value="169"/>																																		
Max. Repeat/Cycle	<input type="text" value="0"/>	Last cycle with no Dxch:	<input type="text" value="18:27:05"/>																																		
Last Repeat:	<input type="text"/>	Failed-Cycle-Counter:	<input type="text" value="0"/>																																		
		Last Failed-Cycle:	<input type="text"/>																																		
<input type="button" value="Reset"/>																																					

Abb. 31: TwinCAT 2.9 - Karteireiter DP-Diag

Hier werden Busverkabelungsprobleme und DP-Zykluszeiten angezeigt:

detected bus-errors: Hier wird die Anzahl der festgestellten Busfehler angezeigt, wenn dieser Zähler ungleich 0 ist, sollte die Verkabelung geprüft werden (falls keine PROFIBUS-Stecker abgezogen oder gesteckt wurden (in der Regel gibt es beim Abziehen oder Stecken von PROFIBUS-Steckern auch kurze Busstörungen)).

CycleWithRepeatCounter: Hier wird die Anzahl der PROFIBUS-Zyklen angezeigt, in denen mindestens einmal ein Telegramm wiederholt wurde. Wiederholungen sind auch ein Anzeichen dafür, dass die Busphysik nicht völlig okay ist.

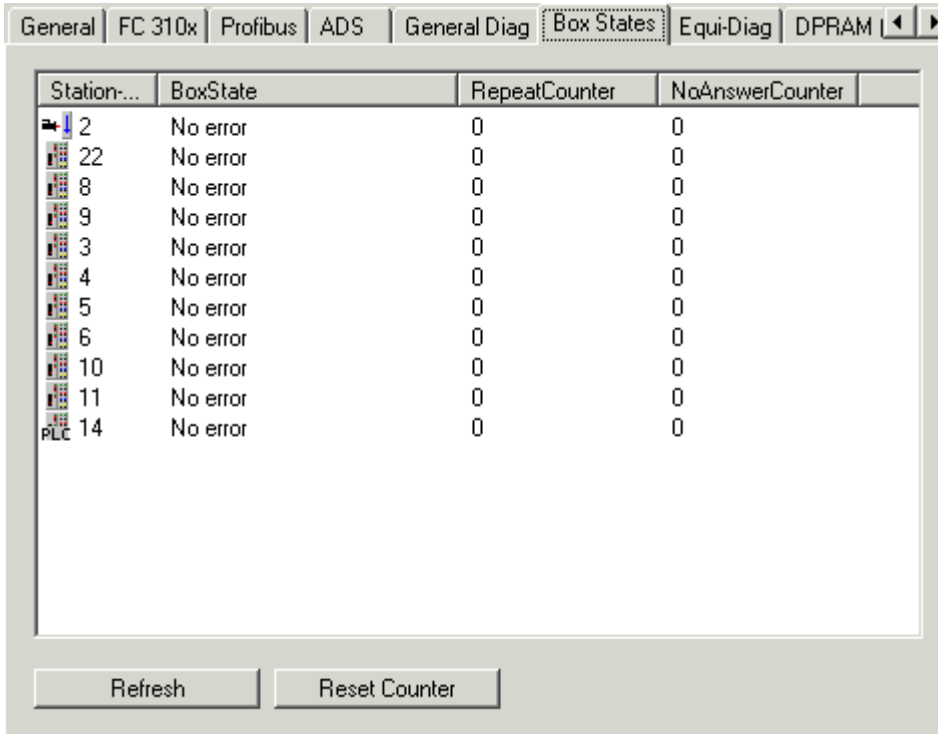
max. Repeat/Cycle: Hier die maximale Anzahl von Wiederholungen innerhalb eines Zyklus angezeigt.

min./max./actual Cycle-Time: Hier werden minimale, maximale und aktuelle DP-Zykluszeit angezeigt, wobei nur Zyklen berücksichtigt werden, in denen alle Slaves im Datenaustausch waren und keine Wiederholungen auftraten.

CycleWithNoDxch-Counter: zählt hoch, wenn nicht alle Slaves im Datenaustausch sind (d.h. einen DpState ungleich 0 haben)

Real Failed-Cycle-Counter: zählt hoch, wenn der DP-Zyklus nicht fertig wurde, bevor der nächste Task-Zyklus gestartet ist und alle Slaves im Datenaustausch sind (d.h. einen DpState gleich 0 haben).

3.8.2.7 Karteireiter Box States



Station...	BoxState	RepeatCounter	NoAnswerCounter
2	No error	0	0
22	No error	0	0
8	No error	0	0
9	No error	0	0
3	No error	0	0
4	No error	0	0
5	No error	0	0
6	No error	0	0
10	No error	0	0
11	No error	0	0
PLC 14	No error	0	0

Abb. 32: TwinCAT 2.9 - Karteireiter Box States

Hier wird eine Übersicht aller aktuellen Box-States, der Repeat-Counter (zählt bei jeder Telegrammwiederholung zu dem Slave hoch) und der NoAnswer-Counter (zählt jedes Mal hoch, wenn der Slave nicht geantwortet hat) angezeigt.

3.8.2.8 Karteireiter MC-Diag

Es gibt den Karteireiter "EquiDiag", auf dem Online verschiedene Equidistant-Überwachungsparameter angezeigt werden:

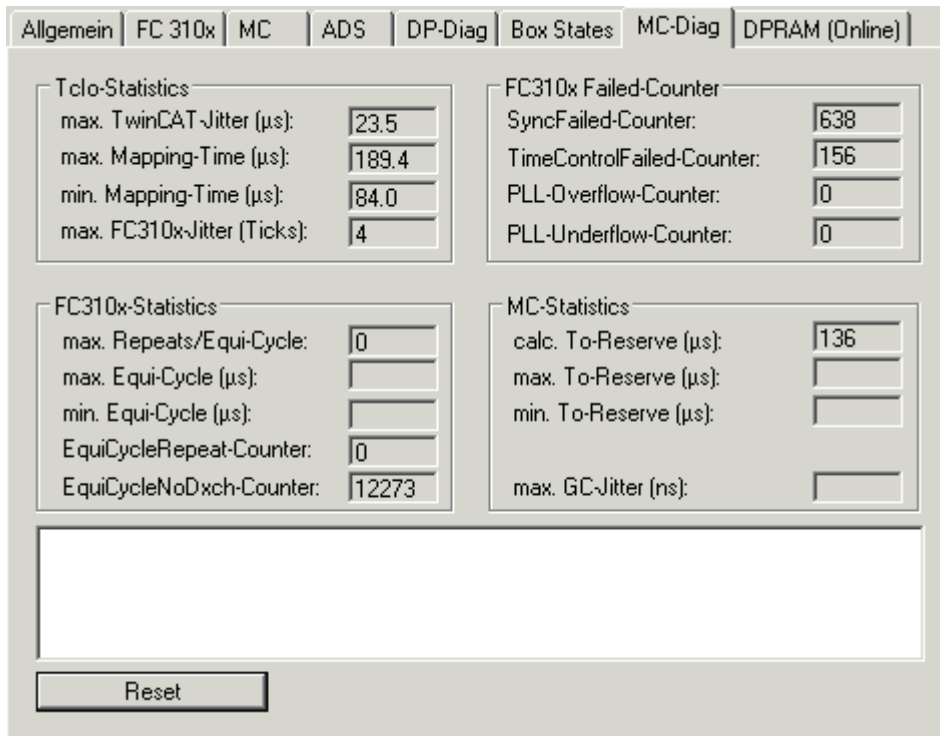


Abb. 33: TwinCAT 2.9 - Karteireiter MC-Diag

Tclo-Statistics

max. TwinCAT-Jitter: Hier wird der maximale TwinCAT-Jitter angezeigt, die Messung wird zurückgesetzt, sobald der Karteireiter nicht mehr aktiv ist.

min./max. Mapping-Time bzw. NC-Access-Time: Hier die minimale bzw. maximale NC-Access-Time angezeigt, wobei diese neben der Mapping-Zeit noch die Task-Laufzeit enthält (falls "IO am Taskanfang" bei der höchstprioriten mit der FC310x verknüpften Task (in der Regel die NC-Task) gesetzt ist).

Die NC-Access-Time sollte größer als die Addition dieser beiden gemessenen Zeiten (max. TwinCAT-Jitter plus max. Mapping-Time bzw. (NC-)Task-Laufzeit) mit einem Sicherheitsaufschlag (ca. 10 %) eingestellt werden.

max. FCxxxx-Jitter: Ist nur bei Sync-Mode = "Disabled" von Bedeutung, wenn der Wert größer als 5 wird, jittert die Echtzeit zu stark, es sollte dann ein leistungsfähigerer PC genommen werden. Falls die zugehörige (NC-)Task nicht die höchste Priorität hat, kann der max. FCxxxx-Jitter auch größer werden, das sollte vermieden werden.

FC310x-Statistics

Da die PROFIBUS MC-Slaves immer zuerst an das FC310x-Device angefügt werden sollten, werden diese auch immer vor den DP-Slaves im DP-Zyklus angesprochen, dieser Teil des DP-Zyklus wird im Folgenden als Equi-Cycle bezeichnet. Wenn der Equi-Cycle größer als die To-Zeit der MC-Slaves wird, bekommen die letzten MC-Slaves am FC310x-Device in der Regel einen Synchronisationsfehler (Fehler 597 oder 598 bei Simodrive 611U).

max. Repeats/Equi-Cycle: Hier wird angezeigt, wie viele Data_Exchange-Telegramme während eines Equi-Cycles maximal wiederholt wurden, dadurch verlängert sich der Equi-Cycle, normalerweise sollten keine Wiederholungen auftreten (es sein denn, dass Busstecker gezogen oder ein MC-Slave ausgeschaltet wurde).

min./max. Equi-Cycle: Hier wird die minimale bzw. die maximale Equi-Cycle-Time angezeigt.

Equi-Cycle-Repeat-Counter: Hier wird angezeigt, wie oft im Equi-Cycle Telegramm-Wiederholungen auftraten.

Equi-Cycle-NoDxch-Counter: Hier wird angezeigt, wie oft im Equi-Cycle nicht alle MC-Slaves im Datenaustausch waren.

FC310x-Failed-Counter

Sync-Failed-Counter: Dieser Zähler zählt hoch, wenn TwinCAT-Task und DP-Zyklus nicht miteinander synchronisiert sind. Das kann beim Hochlauf des TwinCAT Systems passieren, danach sollte dieser Zähler nicht hochzählen. Falls die zugehörige (NC-)Task nicht die höchste Priorität hat, kann dieser Zähler auch hochzählen, das sollte vermieden werden.

Time-Control-Failed-Counter: Dieser Zähler zählt hoch, wenn zum Zeitpunkt des DP-Zyklus-Starts der PROFIBUS nicht frei war. Ursache dafür können Busstörungen, nicht vorhandenen Teilnehmer, ein zweiter Master oder eine zu kleine Safety-Time sein.

PLL-Overflow-/Underflow-Counter: Dieser Zähler ist nur beim Sync-Mode "Disabled" relevant und zählt hoch, falls die TwinCAT-Task, auf die sich der DP-Zyklus synchronisiert, zu stark jittert (kann z. B. passieren, falls der DP-Zyklus nicht mit der höchstpriorären Task synchronisiert ist). Falls die zugehörige (NC-)Task nicht die höchste Priorität hat, kann dieser Zähler auch hochzählen, das sollte vermieden werden.

To-Statistics

Mit der To-Zeit wird bei jedem MC-Slave festgelegt, wann, bezogen auf den DP-Zyklusstart, dieser die vom Master empfangenen Outputs übernehmen soll. Die MC-Slaves können miteinander synchronisiert werden, wenn bei allen MC-Slaves der gleiche Wert für To eingestellt wird. Dieser muss aber mindestens so groß wie die Equi-Cycle-Time plus eine Sicherheitsreserve von ca. 200 µs sein. Die To-Zeit wird bei allen MC-Slaves mit dem Button "Calc. Equi-Times" (s.o.) berechnet.

calc. To-Reserve: Hier steht die kalkulierte To-Reserve (To-Zeit - Equi-Cycle-Time).

min./max. To-Reserve: Hier wird die min. bzw. max. To-Reserve gemessen.

max. GC-Jitter (ab TwinCAT 2.8)

Hier wird der maximale Jitter des DP-Zyklus gemessen (GC für Global-Control-Telegramm, das immer am Anfang eines Zyklus gesendet wird). Beim Hochlauf kann der Jitter etwas größer sein, im eingeschwungenen Zustand sollte er 1 µs (bei Sync-Mode "Sync-Master") bzw. 2 µs (bei Sync-Mode "Disabled") nicht überschreiten.

3.9 Karteireiter Box

3.9.1 Karteireiter Profibus

Abb. 34: Karteireiter Profibus

Stations-No.: Hier ist die auf dem jeweiligen Slave eingestellte PROFIBUS-Stationsadresse einzustellen. Es gibt Slaves, bei denen die Stationsadresse nicht hardwaremäßig eingestellt werden kann, sondern nur über den SetSlaveAddress-Dienst. In dem Fall ist der Button "Set..." zu drücken, woraufhin ein Dialog aufgeht, mit dem das Senden eines SetSlaveAddress-Telegramms ausgelöst werden kann.

Watchdog: Schaltet den DP-Watchdog ein. Wenn der Slave bei eingeschaltetem Watchdog für die Dauer der Watchdog-Zeit kein DP-Telegramm empfängt, verlässt er automatisch den Datenaustausch. Die minimal einzustellende Watchdog-Zeit hängt von der DP-Zykluszeit ab und sollte größer als der nach der folgenden Formel berechnete Wert sein: **Estimated-Cycle-Time * 10**

Für besonders kritische Ausgänge kann bei DP-Slaves, die eine Watchdog-Base-Time von 1 ms unterstützen (alle Beckhoff-Slaves mit Ausnahme von BK3000 und BK3100 sowie alle Fremd-Geräte in deren GSD-Datei der Eintrag "WD_Base_1ms_supp = 1" steht) ein DP-Watchdog herunter bis zu 2 ms eingestellt werden, allerdings sollte die DP-Watchdog-Time mindestens doppelt so groß wie das Maximum aus **Cycle-Time** und **Estimated Cycle-Time** sein (s. Karteireiter [FC310x](#) [▶ 41] des Masters).

Ident Nr: Hier wird die Ident-Nummer aus der GSD-Datei angezeigt.

PrmData: Erlaubt die Profibus-spezifischen Parameterdaten zu editieren, die Größe der aktuellen Parameterdaten wird ebenfalls angezeigt. Die PrmData können aber in der Regel textuell (-> PrmData (Text)) oder bei Beckhoff-DP-Slaves teilweise über den Karteireiter "Beckhoff" eingestellt werden

CfgData: Die aktuellen Konfigurationsdaten (ergeben sich aus den angefügten Modulen bzw. Klemmen) sowie deren Länge werden angezeigt.

Sync/Freeze: In der Betriebsart DP/MC (Equidistant) des Masters können Slaves mit [Sync und Freeze](#) [▶ 16] betrieben werden.

DPV1 Class 2: Bei der FC310x kann eine DPV1-Klasse 2-Verbindung zu einem DPV1-Slave aktiviert werden. Das ist z. B. sinnvoll, wenn der DP-Slave mit einem anderen Master im Datenaustausch ist, aber trotzdem von TwinCAT azyklisch angesprochen werden soll. Mit dem Parameter Timeout wird die Verbindungsüberwachungszeit der Klasse 2 Verbindung eingestellt (s. Kapitel [DPV1](#) [▶ 32]).

DP Class 2: Wenn der DP-Slave mit einem anderen Master im Datenaustausch ist, aber trotzdem von TwinCAT azyklisch angesprochen oder die DP-Inputs und Outputs zyklisch gelesen werden sollen, ist unter DP-Class 2 "No Cyclic Connection" oder "ReadOnly" auszuwählen. Wenn "ReadOnly" ausgewählt ist, sind wie bei der normalen zyklischen Verbindung die Module auszuwählen, die aber im TwinCAT System alle mit Input-Variablen erscheinen, egal ob es sich dabei um Input- oder um Output-Module handelt (ReadOnly erst ab Firmware-Version 3.00).

ResetSlave: Mit diesem Button kann bei gestartetem TwinCAT der zyklische Datenaustausch zu dem DP-Slave abgebaut und gleich wieder aufgebaut werden (entspricht einem IO-Reset aber nur für den einen Slave).

3.9.2 Karteireiter Features

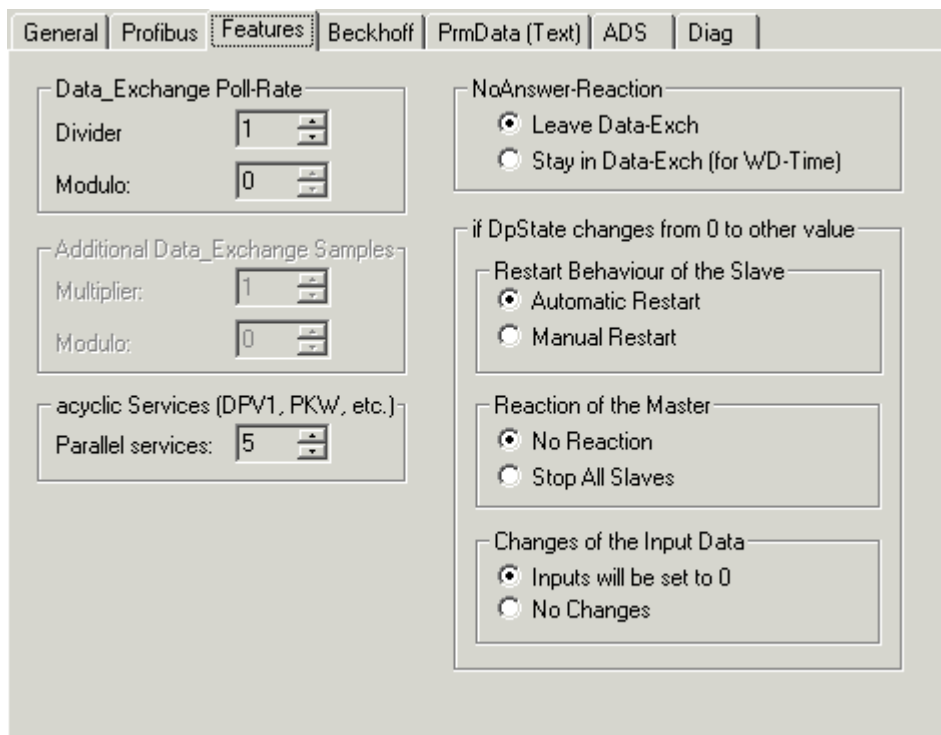


Abb. 35: Karteireiter Features

Data_Exchange Poll-Rate: Je Slave kann eine unterschiedliche Polling-Rate (Divider) eingestellt werden, Divider 1 bedeutet, dass der Slave jeden Zyklus gepollt wird, Divider 2 bedeutet jeden 2. Zyklus, usw. Mit dem Modulo können Slaves mit Divider größer 1 auf unterschiedliche Zyklen aufgeteilt werden, um die maximale Zykluszeit zu verkürzen (Divider 2 und Modulo 0 bedeutet, dass der Slave in jedem geraden Zyklus gepollt wird, Divider 2 und Modulo 1 bedeutet, dass der Slave in jedem ungeraden Zyklus gepollt wird) (s. Kapitel [Slave-Priorisierung/mehrere DP-Zyklen](#) [▶ 14]).

Additional Data_Exchange Samples: Es besteht die Möglichkeit, innerhalb eines Task-Zyklus mehrere DP-Zyklen zu fahren. Optional kann jeder Slave dann noch mit unterschiedlichen Outputdaten je DP-Zyklus versorgt und die Inputdaten eines jeden DP-Zyklus an die Steuerung zu übergeben. In diesem Fall gibt es je DP-Zyklus einen eigenen Variablensatz (s. Kapitel [Slave-Priorisierung/mehrere DP-Zyklen](#) [▶ 14]).

NoAnswer-Reaction: Je Slave kann angegeben werden, ob dieser weiterhin in Data-Exch bleiben soll, auch wenn er nicht oder fehlerhaft antwortet. In diesem Fall (Stay in Data-Exch) wird der Datenaustausch erst verlassen, wenn der Slave innerhalb der Ansprechüberwachungszeit nie korrekt geantwortet hat (falls der **Watchdog** (s. Karteireiter [Profibus](#) [▶ 59] der Box) aktiviert ist, andernfalls wird der Datenaustausch erst beendet, wenn der Slave 65535 mal nicht korrekt geantwortet hat) (s. Kapitel [Fehlerreaktionen](#) [▶ 17]).

Restart-Behaviour: Je Slave kann angegeben werden, ob dieser nach einem Verlassen von Data-Exch automatisch wieder hochlaufen oder im Zustand Wait-Prm bleiben soll (s. Kapitel [Fehlerreaktionen](#) [▶ 17]).

Reaction of the Master: Je Slave kann angegeben werden, ob dessen Verlassen von Data-Exch zum Stopp des PROFIBUS-Zyklus führen soll (alle Slaves verlassen den Datenaustausch und gehen nach Wait-Prm, Wiederanlauf nur durch IO-Reset oder Restart des TwinCAT Systems) (s. Kapitel [Fehlerreaktionen](#) [[▶ 17](#)]).

Changes of the Input Data: Je Slave kann angegeben werden, ob beim Verlassen von Data-Exch (DpState wird ungleich 0) dessen Inputdaten auf 0 gesetzt oder unverändert bleiben sollen (s. Kapitel [Fehlerreaktionen](#) [[▶ 17](#)]).

acyclic Services: Die Anzahl der parallelen ADS-Dienste zu einer Box kann hier eingestellt werden.

3.9.3 Karteireiter Beckhoff

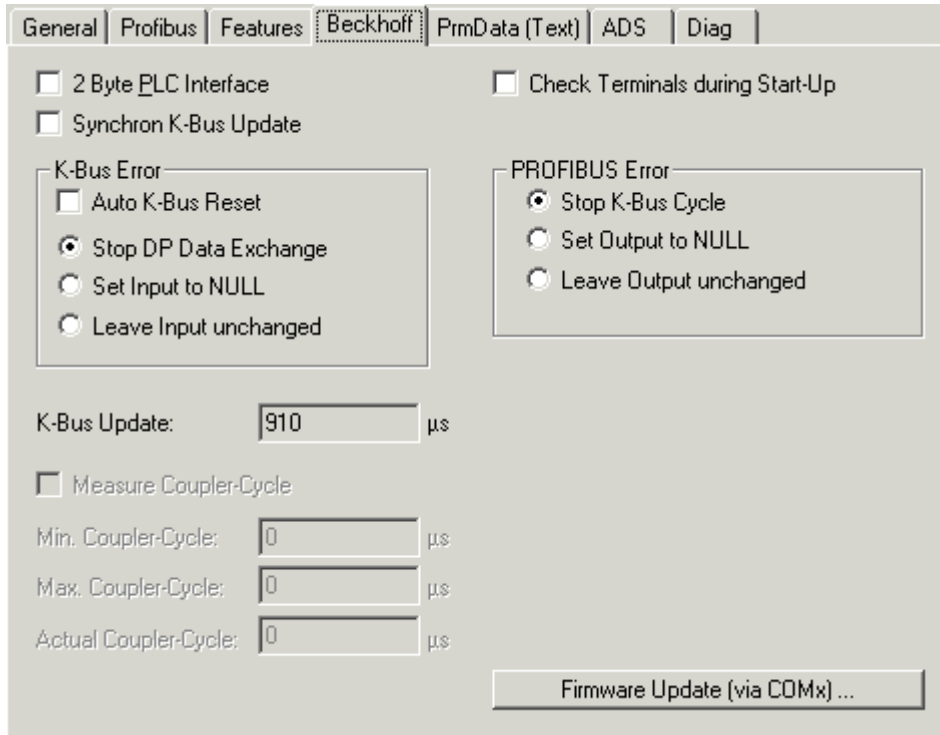


Abb. 36: Karteireiter Beckhoff

FirmwareUpdate: Mit diesem Button kann die Firmware eines Beckhoff-DP-Slaves per KS2000-Kabel über die serielle Schnittstelle aktualisiert werden

2 Byte SPS Interface: Schaltet das 2-Byte SPS Interface des Beckhoff-DP-Slaves ein.

Synchron K-Bus-Update: Hier ist bei Buskopplern die erwartete interne Zykluszeit (K-Bus-Zyklus + Übergabe der DP-Puffer angegeben).

Check Terminals during Start-Up: Wenn diese Check-Box aktiviert wird, wird die Tabelle 9 per DPV1-Write an den Koppler übertragen und der Koppler geht nur in den Datenaustausch (DpState = 0), wenn die Einträge übereinstimmen. Dadurch ist eine genauere Prüfung der Klemmen beim Hochlauf möglich, als es mit den PROFIBUS-CfgData getan wird.

K-Bus Error: Hier kann die Reaktion auf einen K-Bus-Fehler angegeben werden (automatischer oder manueller K-Bus Reset, Reaktion in den Inputdaten im Koppler)

PROFIBUS Error: Hier kann die Reaktion auf einen PROFIBUS-Fehler angegeben werden (Reaktion in den Outputdaten im Koppler)

Measure Coupler-Cycle: Hier kann die Zykluszeit auf dem Koppler (DP + K-Bus) gemessen werden, die K-Bus Update-Zeit ist manchmal etwas ungenau für synchronisierte Vorgänge.

3.9.4 Karteireiter ProcessData

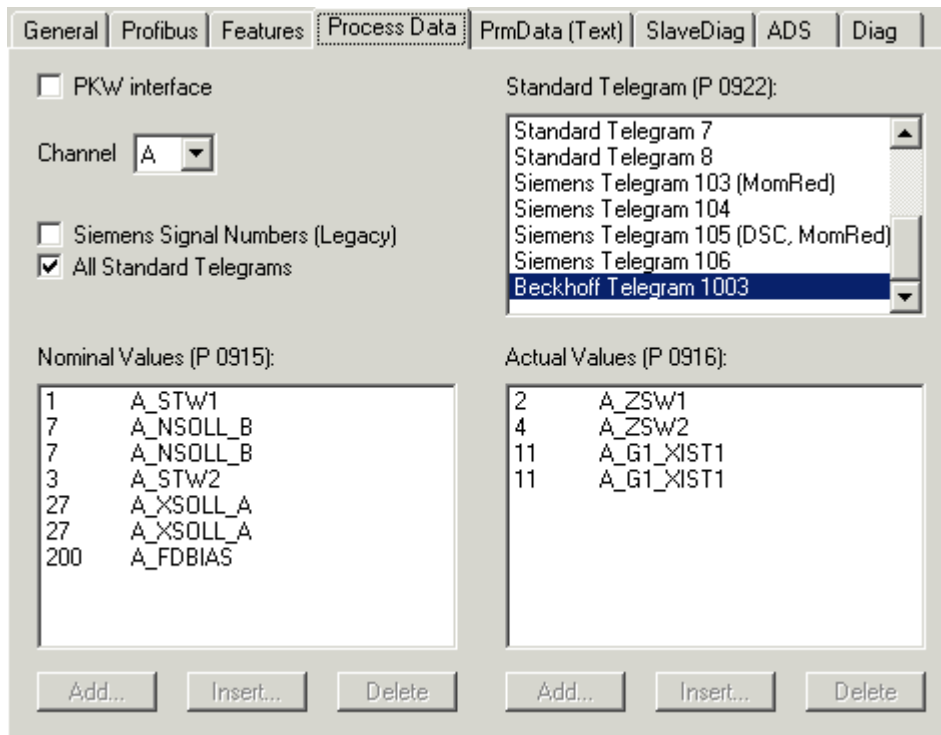


Abb. 37: Karteireiter ProcessData

Unter Siemens AG bzw. Profidrive MC können die Boxen Profidrive MC und Profidrive MC (2-fach) ausgewählt werden. Standardmäßig bekommt die Box die Einstellungen für einen Simodrive 611U mit Standardtelegramm 3. Für andere PROFIBUS MC-Slaves ist die Ident-Nummer auf dem Karteireiter [Profibus](#) [► 59] des Slaves entsprechend zu ändern. Der gewünschte Telegrammtyp kann auf dem Karteireiter **ProcessData** eingestellt werden und muss in der Regel mit dem Parameter 922 des PROFIBUS MC-Slaves übereinstimmen, der über ein herstellerspezifisches Konfigurationstool (SimoCon U bei Simodrive 611U) eingestellt werden kann.

Auf diesem Karteireiter gibt es auch eine Check-Box "PKW-Interface", die das PKW-Interface aktiviert. darüber können online die Parameter der Box auf dem Karteireiter Online angezeigt werden (bisher funktioniert das nur mit dem Simodrive 611U, da ein Parameterfile erforderlich ist, dass der SimoCon U erzeugt). Auf jeden Fall ist es möglich, die Parameter über das PKW-Interface per ADS zu lesen und schreiben (s. Kapitel [PKW-Protokoll](#) [► 36]).

3.9.5 Karteireiter PrmData (Text)



Abb. 38: Karteireiter PrmData (Text)

Durch Anklicken einer Zeile kann der aktuelle Wert verändert werden. Die Beschreibung der jeweiligen Einstellungen ist der Dokumentation des entsprechenden Herstellers zu entnehmen.

3.9.6 Karteireiter Diag

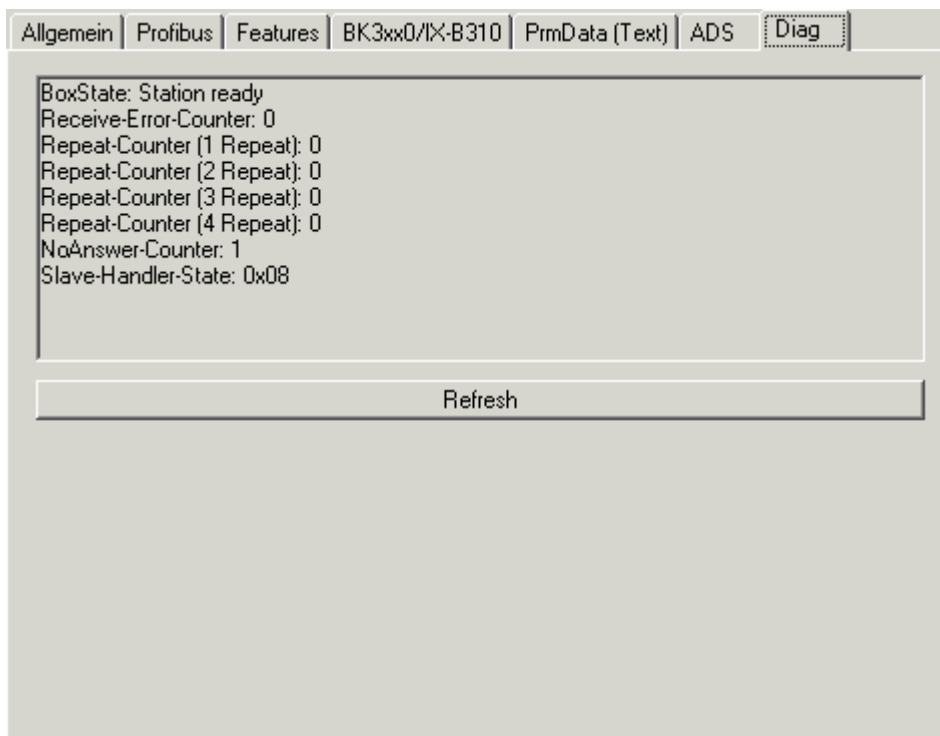


Abb. 39: Karteireiter Diag

Hier können die folgenden Informationen angezeigt werden:

BoxState: Hier wird der aktuelle [DpState \[► 26\]](#) angezeigt.

Receive-Error-Counter: Anzahl der gestörten Telegramme von dem Slave.

Repeat-Counter: Anzahl der notwendigen Wiederholungen aufgrund keiner oder einer gestörten Antwort des Slaves.

NoAnswer-Counter: Anzahl der Telegramme, auf die der Slave nicht geantwortet hat.

letzter DPV1-Fehler: Error-Decode, Error-Class, Error-Code und Error-Code 2 (s. Beschreibung [DPV1-Fehlercodes \[► 73\]](#)).

Bei Beckhoff DP-Slaves werden noch weitere Diagnoseinformationen angezeigt.

4 FC310x als Slave

4.1 Slave

Als Slave werden die Protokolle PROFIBUS DP und PROFIBUS DPV1 unterstützt.

PROFIBUS DP

Um den Slave für den zyklischen DP-Betrieb zu konfigurieren, ist im TwinCAT System Manager wie folgt vorzugehen:

DP-Slave projektieren

Zuerst ist ein E/A-Gerät "PROFIBUS Slave FC310x, PCI" zu projektieren (mit rechter Maustaste auf "E/A-Geräte", dann "Gerät anfügen" auswählen). Dabei werden das Device und eine Box angefügt (dazu muss die GSD-Datei "TCDPSLAV.GSD" im Verzeichnis "TwinCAT\Io\PROFIBUS" stehen):

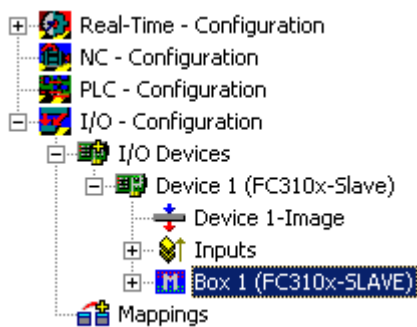


Abb. 40: FC310x als Slave angefügt

Der entsprechende Kanal ist auf dem Karteireiter "**FC310x**" (TwinCAT 2.8 [► 41] bzw. TwinCAT 2.9 [► 49]) des Devices zu suchen ("Search"-Button), die Stationsadresse anzupassen und ggf. die Baudrate anzupassen (ist standardmäßig auf 12 Mbit/s eingestellt).

Module anfügen

Entsprechend der zyklisch zu übertragenen Daten sind Module an die Box anzufügen (mit rechter Maustaste auf die Box, dann "Module anfügen" auswählen):

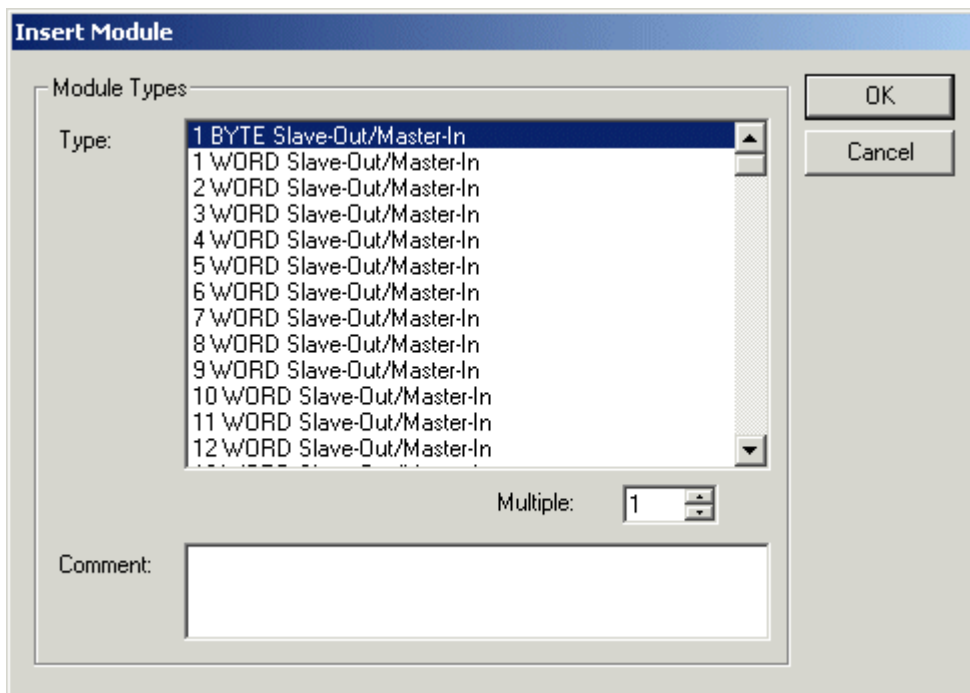


Abb. 41: Module anfügen

Konfiguration des Masters

Zur Konfiguration des Masters ist die GSD-Datei "TCDP SLAV.GSD" aus dem Verzeichnis "TwinCAT\Io\PROFIBUS" zu nehmen. Die Module müssen im Master-Konfigurator **in der gleichen Reihenfolge** angefügt werden, wie bei der Konfiguration des Slaves im System Manager.

Fehlerreaktionen

In der Default-Einstellung werden die Eingänge des DP-Slaves auf 0 gesetzt, wenn der DP-Slave nicht im Datenaustausch ist. Auf dem Karteireiter "Features" der Box kann diese Input-Fehlerreaktion **Changes of the Input Data** auf "No Changes" gestellt werden:

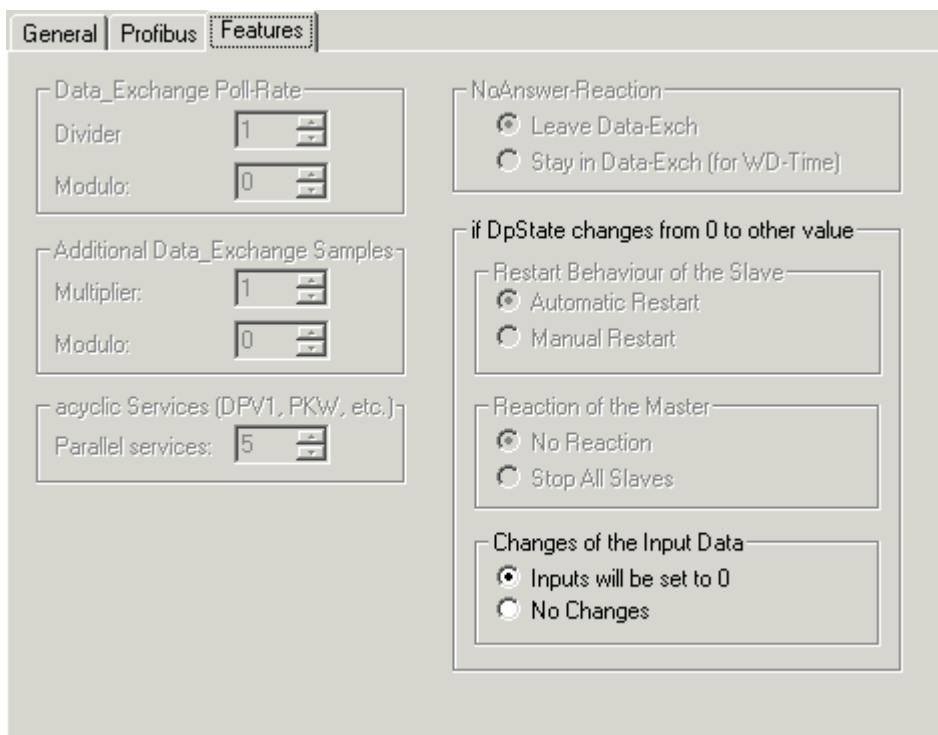


Abb. 42: Features

PROFIBUS DPV1

Der DP-Slave unterstützt eine DPV1-MSAC_C1-Server-Verbindung, die zusammen mit der zyklischen Verbindung aufgebaut wird. Diese kann benutzt werden, um neben den zyklischen Daten noch größere Mengen azyklische Daten übertragen zu können. Ein vom Master empfangenes DPV1-Read-Telegramm wird als ADS-Read-Indication an die PLC gemeldet, ein vom Master empfangenes DPV1-Write-Telegramm wird als ADS-Write-Indication an die PLC gemeldet. Das PLC-Programm ist dann für die Read bzw. Write-Antwort verantwortlich, dazu sind die Funktionen ADS-Read-Response bzw. ADS-Write-Response aufzurufen.

MSAC-C1 Read

Eine DPV1-MSAC_C1-Read-Indication wird wie folgt auf eine ADS-Read-Indication abgebildet:

ADS-Read-Indication-Parameter	Bedeutung
Source-Net-ID (NETID)	Net-ID des Slaves (s. Karteireiter ADS [▶ 45] des Devices)
Source-Port (PORT)	0x200
Invoke-ID (INVOKEID)	eindeutige Nummer, muss bei Response wieder auftauchen
IndexGroup (IDXGRP)	Slot-Number (DPV1-Parameter)
IndexOffset (IDXOFFS)	Index (DPV1-Parameter)
Length (LENGTH)	Länge der auszulesenden Daten

Eine ADS-Read-Response wird wie folgt auf eine DPV1-MSAC_C1-Read-Response abgebildet:

ADS-Read-Response-Parameter	Bedeutung
Destination-Net-ID (NETID)	Net-ID des Slaves (s. Karteireiter ADS [▶ 45] des Devices)
Destination-Port (PORT)	0x200
Invoke-ID (INVOKEID)	eindeutige Nummer, wie bei Indication
Result (RESULT)	Result des Read: 0 = kein Fehler, sonst: Bit 0-15 = Standard-ADS-Fehlercodes, Bit 16-23 = Error_Code_1, Bit 24-31 = Error_Code_2, s. Beschreibung DPV1-Fehlercodes [▶ 73]
Length (LENGTH)	Länge der gelesenen Daten
Data (DATAADDR)	gelesene Daten

MSAC-C1 Write

Eine DPV1-MSAC_C1-Write-Indication wird wie folgt auf eine ADS-Write-Indication abgebildet:

ADS-Write-Indication-Parameter	Bedeutung
Source-Net-ID (NETID)	Net-ID des Slaves (s. Karteireiter ADS [▶ 45] des Devices)
Source-Port (PORT)	0x200
Invoke-ID (INVOKEID)	eindeutige Nummer, muss bei Response wieder auftauchen
IndexGroup (IDXGRP)	Slot-Number (DPV1-Parameter)
IndexOffset (IDXOFFS)	Index (DPV1-Parameter)
Length (LENGTH)	Länge der zu schreibenden Daten
Data (DATAADDR)	zu schreibende Daten

Eine ADS-Read-Response wird wie folgt auf eine DPV1-MSAC_C1-Read-Response abgebildet:

ADS-Read-Response-Parameter	Bedeutung
Destination-Net-ID (NETID)	Net-ID des Slaves (s. Karteireiter ADS [► 45] des Devices)
Destination-Port (PORT)	0x200
Invoke-ID (INVOKEID)	eindeutige Nummer, wie bei Indication
Result (RESULT)	Result des Read: 0 = kein Fehler, sonst: Bit 0-15 = Standard-ADS-Fehlercode, Bit 16-23 = Error_Code_1, Bit 24-31 = Error_Code_2, s. Beschreibung DPV1-Fehlercodes [► 73]
Length (LENGTH)	Länge der gelesenen Daten

ADS-Interface (ab TwinCAT 2.9)

In TwinCAT-Systemen kann die Kommunikation auch über ADS erfolgen. Von der Funktionalität entspricht das einer ADS-Verbindung zwischen zwei PCs über Ethernet, nur dass stattdessen die Übertragung über PROFIBUS erfolgt, mit der Ausnahme, dass der Requester, der den ADS-Auftrag anstößt, immer der DP-Master-PC ist. Ein FC310x-DP-Master ist dann mit einem FC310x-DP-Slave verbunden.

Beim DP-Master ist dann auf dem Karteireiter ADS der Box das **ADS-Interface** zu aktivieren und die **Net-ID** des DP-Slave-PCs einzutragen:



Abb. 43: Aktivieren des ADS-Interfaces auf dem DP-Master

Beim DP-Slave ist auf dem Karteireiter ADS des Devices die Net-ID des DP-Master-PCs unter **Add. NetIds** einzutragen:

The screenshot shows a software configuration window with a tabbed interface. The active tab is 'ADS'. The window contains the following fields and controls:

- Use Port
- Port No: 28673 (0x7001) [Change...]
- NetId: 10.3.2.99.2.1
- Remote Name: Device 1 (FC310x-Slave)
- Add. NetIds: [Empty field] [Add]
- 10.3.2.98.1.1 [Delete]

Abb. 44: Karteireiter ADS auf dem DP-Slave

5 Anhang

5.1 Diagnosedaten - DiagData

Es folgt eine Beschreibung der DP-Diagnosedaten

Offset	Bedeutung
0x00.0	StationNonExistent: Slave beim letzten Telegramm nicht geantwortet
0x00.1	StationNotReady: Slave verarbeitet noch das Set_Prm bzw. Chk_Cfg-Telegramm
0x00.2	CfgFault: Slave meldet einen Konfigurationsfehler
0x00.3	ExtDiag: Extended DiagData sind vorhanden und gültig
0x00.4	NotSupported: Slave unterstützt ein Feature nicht, das mit Set_Prm oder Global_Control gefordert wurde
0x00.5	InvalidSlaveResponse: Slave antwortet nicht DP-konform
0x00.6	PrmFault: Slave meldet einen Parametrierfehler
0x00.7	MasterLock: Slave ist im Datenaustausch mit einem anderen Master
0x01.0	PrmReq: Slave muß neu parametrier und konfiguriert werden
0x01.1	StatDiag: Slave meldet statische Diagnose bzw. Applikation des DPV1-Slave noch nicht bereit für den Datenaustausch
0x01.2	PROFIBUS-DP-Slave
0x01.3	WdOn: DP-Watchdog ist eingeschaltet
0x01.4	FreezeMode: DP-Slave ist im Freeze-Mode
0x01.5	SyncMode: DP-Slave ist im Sync-Mode
0x01.6	reserviert
0x01.7	Deactivated: DP-Slave wurde deaktiviert
0x02.0	reserviert
0x02.1	reserviert
0x02.2	reserviert
0x02.3	reserviert
0x02.4	reserviert
0x02.5	reserviert
0x02.6	reserviert
0x02.7	ExtDiagOverflow: zu viele Extended DiagData vorhanden
0x03	MasterAdd: Stationsadresse des Masters, der mit dem Slave Datenaustausch macht
0x04,0x05	IdentNumber
ab 0x06	Extended DiagData

Extended DiagData

Bei den Extended DiagData wird zwischen Kennungsdiagnose, Kanaldiagnose und Herstellerspezifischer Diagnose unterschieden, wobei das erste Byte jeweils den Typ der Diagnose und die Länge der dazugehörigen Daten anzeigt. In den Extended DiagData können auch mehrere Diagnosetypen nacheinander folgen.

Header-Byte

Bit	Bedeutung
0-5	Länge der zugehörigen Diagnosedaten inklusive Header-Byte
6-7	0 = Herstellerspezifische Diagnose (DPV1 wird nicht unterstützt) bzw. DPV1-Diagnose (DPV1 wird unterstützt (DPV1_Enable = 1) in zugehöriger GSD-Datei)
	Moduldiagnose
	Kanaldiagnose
	Revision-Number

Herstellerspezifische Diagnose

Der Aufbau der Herstellerspezifischen Diagnose ist der Dokumentation des DP-Slaves zu entnehmen.

DPV1-Diagnose

Bei DP-Slaves, die auch DPV1 unterstützen wird statt der Herstellerspezifischen Diagnose die DPV1-Diagnose gesendet, bei der zwischen Status-Messages und Alarmen unterschieden wird:

Byte	Bedeutung
0	Header-Byte (Bit 6,7 = 0, Bit 0-5 = 4..63)
1	Bit 0-6: Alarm-Typ Bit 7: immer 0
2	Slot-Number (0-254)
3	Bit 0-1: Alarm-Specifier Bit 2: Additional-Acknowledge Bit 3-7: Sequence-Number
4-63	herstellerspezifisch (s. Dokumentation des DP-Slaves)

Alarm-Typ

Wert	Bedeutung
0	reserviert
1	Diagnose-Alarm
2	Prozess-Alarm
3	Ziehen-Alarm
4	Stecken-Alarm
5	Status-Alarm
6	Update-Alarm
7-31	reserviert
20-126	herstellerspezifisch (s. Dokumentation des DP-Slaves)
127	reserviert

Moduldiagnose

In der Moduldiagnose gibt es je DP-Modul ein Bit, das anzeigt, ob bei dem entsprechenden DP-Modul eine Diagnose vorliegt.

Byte	Bedeutung
0	Header-Byte (Bit 6,7 = 1, Bit 0-5 = 2..32)
1	Bit 0: 1. DP-Modul hat Diagnose
	Bit 1: 2. DP-Modul hat Diagnose
	...
	Bit 7: 8. DP-Modul hat Diagnose
...	...
31	Bit 0: 241. DP-Modul hat Diagnose
	Bit 1: 242. DP-Modul hat Diagnose
	Bit 2: 243. DP-Modul hat Diagnose
	Bit 3: 244. DP-Modul hat Diagnose (maximal 244 DP-Module möglich)

Kanaldiagnose

Mit der Kanaldiagnose wird die Diagnoseursache eines DP-Moduls genauer beschrieben.

Byte	Bedeutung
0	Header-Byte = 0x83 (3 Bytes inklusive Header, Bit 6,7 = 2)
1	Bit 0-5: Kanal-Nummer
	Bit 6-7: 0 = Reserved, 1 = Input, 2 = Output, 3 = In-/Output
2	Bit 0-4: Error-Typ
	Bit 5-7: Kanal-Typ

Error-Typ

Wert	Bedeutung
0	reserviert
1	Kurzschluss
2	Unterspannung
3	Überspannung
4	Überlast
5	Übertemperatur
6	Leitungsbruch
7	oberer Grenzwert überschritten
8	unterer Grenzwert unterschritten
9	Fehler
10-15	reserviert
16-31	herstellerspezifisch (s. Dokumentation des DP-Slaves)

Kanal-Typ

Wert	Bedeutung
0	beliebiger Typ
1	Bit
2	2 Bit
3	4 Bit
4	Byte
5	Wort
6	2 Worte
7	reserviert

Revision-Number

Der Aufbau der Revision-Number ist der Dokumentation des DP-Slaves zu entnehmen.

5.1.1 DPV1-Fehlercodes

Bei einem fehlerhaften DPV1-Zugriff antwortet der Slaves mit 4 Bytes Daten (alle nicht beschriebenen Werte sind nicht in der DPV1-Norm beschrieben und daher dem Handbuch des Slaves zu entnehmen):

Byte 0	DPV1-Dienst
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 (Bit 4-7)	Error-Code (Bit 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
0x0C	0x09	Access, Invalid Type
	0x00	Resource, Read Constrain Conflict
	0x01	Resource, Write Constrain Conflict
	0x02	Resource, Busy
	0x03	Resource, Unavailable

Byte 3	Error_Code_2

5.2 Konfigurationsdaten - CfgData

Die CfgData beschreiben die Struktur und die Länge der Ein- und Ausgangsdaten, die per Data_Exchange zyklisch ausgetauscht werden sollen. Es folgt eine Beschreibung der DP-Konfigurationsdatenbytes

Bit 4-7	Bedeutung
0000B	Modul ohne Daten, Bit 0-3 geben an, wie viel Bytes herstellerspezifische Daten noch folgen
0001B	Eingänge vom Typ Byte ohne Konsistenz, in Bit 0-3 steht die Länge der Eingangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Byte, Bit 0-3 = 1111B entspricht Länge 16 Bytes)
0010B	Ausgänge vom Typ Byte ohne Konsistenz, in Bit 0-3 steht die Länge der Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Byte, Bit 0-3 = 1111B entspricht Länge 16 Bytes)
0011B	Ein- und Ausgänge vom Typ Byte ohne Konsistenz, in Bit 0-3 steht die Länge der Eingangs- bzw. Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Byte, Bit 0-3 = 1111B entspricht Länge 16 Bytes)
0100B	spezielles Kennungsformat Eingänge, es folgt ein Byte, das die zugehörigen Eingangsdaten beschreibt (s.u.), Bit 0-3 geben an, wie viel Bytes herstellerspezifische Daten danach noch folgen
0101B	Eingänge vom Typ Word ohne Konsistenz, in Bit 0-3 steht die Länge der Eingangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Word, Bit 0-3 = 1111B entspricht Länge 16 Words)
0110B	Ausgänge vom Typ Word ohne Konsistenz, in Bit 0-3 steht die Länge der Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Word, Bit 0-3 = 1111B entspricht Länge 16 Words)
0111B	Ein- und Ausgänge vom Typ Word ohne Konsistenz, in Bit 0-3 steht die Länge der Eingangs- bzw. Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Word, Bit 0-3 = 1111B entspricht Länge 16 Words)
1000B	spezielles Kennungsformat Ausgänge, es folgt ein Byte, das die zugehörigen Ausgangsdaten beschreibt (s.u.), Bit 0-3 geben an, wie viel Bytes herstellerspezifische Daten danach noch folgen
1001B	Eingänge vom Typ Byte mit Konsistenz, in Bit 0-3 steht die Länge der Eingangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Byte, Bit 0-3 = 1111B entspricht Länge 16 Bytes)
1010B	Ausgänge vom Typ Byte mit Konsistenz, in Bit 0-3 steht die Länge der Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Byte, Bit 0-3 = 1111B entspricht Länge 16 Bytes)
1011B	Ein- und Ausgänge vom Typ Byte mit Konsistenz, in Bit 0-3 steht die Länge der Eingangs- bzw. Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Byte, Bit 0-3 = 1111B entspricht Länge 16 Bytes)
1100B	spezielles Kennungsformat Ein- und Ausgänge, es folgt erst ein Byte, das die zugehörigen Ausgangsdaten, dann ein Byte, das die zugehörigen Eingangsdaten beschreibt (s.u.), Bit 0-3 geben an, wie viel Bytes herstellerspezifische Daten danach noch folgen
1101B	Eingänge vom Typ Word mit Konsistenz, in Bit 0-3 steht die Länge der Eingangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Word, Bit 0-3 = 1111B entspricht Länge 16 Words)
1110B	Ausgänge vom Typ Word mit Konsistenz, in Bit 0-3 steht die Länge der Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Word, Bit 0-3 = 1111B entspricht Länge 16 Words)
1111B	Ein- und Ausgänge vom Typ Word mit Konsistenz, in Bit 0-3 steht die Länge der Eingangs- bzw. Ausgangsdaten minus 1 (d.h. Bit 0-3 = 0000B entspricht Länge 1 Word, Bit 0-3 = 1111B entspricht Länge 16 Words)

Falls das erste Byte vom Typ "spezielles Kennungsformat" ist hat das zweite bzw. dritte Byte die folgende Bedeutung:

Bit 6-7	Bedeutung
00B	Typ Byte ohne Konsistenz, in Bit 0-5 steht die Länge der Daten minus 1 (d.h. Bit 0-5 = 000000B entspricht Länge 1 Byte, Bit 0-5 = 111111B entspricht Länge 64 Bytes)
01B	Typ Word ohne Konsistenz, in Bit 0-5 steht die Länge der Daten minus 1 (d.h. Bit 0-5 = 000000B entspricht Länge 1 Word, Bit 0-5 = 111111B entspricht Länge 64 Words)
10B	Typ Byte mit Konsistenz, in Bit 0-5 steht die Länge der Daten minus 1 (d.h. Bit 0-5 = 000000B entspricht Länge 1 Byte, Bit 0-5 = 111111B entspricht Länge 64 Bytes)
11B	Typ Word mit Konsistenz, in Bit 0-5 steht die Länge der Daten minus 1 (d.h. Bit 0-5 = 000000B entspricht Länge 1 Word, Bit 0-5 = 111111B entspricht Länge 64 Words)

5.3 Support und Service

Beckhoff und seine weltweiten Partnerfirmen bieten einen umfassenden Support und Service, der eine schnelle und kompetente Unterstützung bei allen Fragen zu Beckhoff Produkten und Systemlösungen zur Verfügung stellt.

Beckhoff Support

Der Support bietet Ihnen einen umfangreichen technischen Support, der Sie nicht nur bei dem Einsatz einzelner Beckhoff Produkte, sondern auch bei weiteren umfassenden Dienstleistungen unterstützt:

- Support
- Planung, Programmierung und Inbetriebnahme komplexer Automatisierungssysteme
- umfangreiches Schulungsprogramm für Beckhoff Systemkomponenten

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

Beckhoff Service

Das Beckhoff Service-Center unterstützt Sie rund um den After-Sales-Service:

- Vor-Ort-Service
- Reparaturservice
- Ersatzteilservice
- Hotline-Service

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

Weitere Support- und Serviceadressen finden Sie auf unseren Internetseiten unter <http://www.beckhoff.de>.

Beckhoff Firmenzentrale

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Deutschland

Telefon: +49(0)5246/963-0
Fax: +49(0)5246/963-198
E-Mail: info@beckhoff.com

Die Adressen der weltweiten Beckhoff Niederlassungen und Vertretungen entnehmen Sie bitte unseren Internetseiten:

<http://www.beckhoff.de>

Dort finden Sie auch weitere Dokumentationen zu Beckhoff Komponenten.

Abbildungsverzeichnis

Abb. 1	Anfügen eines DP-Slaves.....	11
Abb. 2	I/O am Task-Anfang und keine Überschreitung der Echtzeit-Ressourcen	12
Abb. 3	I/O am Task-Anfang und Überschreitung der Echtzeit-Ressourcen.....	13
Abb. 4	I/O nicht am Task-Anfang und keine Überschreitung der Echtzeit-Ressourcen.....	13
Abb. 5	I/O nicht am Task-Anfang und Überschreitung der Echtzeit-Ressourcen	13
Abb. 6	Aufteilung der DP-Slaves auf mehrere DP-Zyklen	14
Abb. 7	Mehrere DP-Zyklen in einem Task-Zyklus.....	15
Abb. 8	Mehrere Data-Samples innerhalb eines Task-Zyklus.....	15
Abb. 9	Sync/Freeze-Funktionalität	16
Abb. 10	Normaler DP-Zyklus	17
Abb. 11	Erstmalig gestörter DP-Zyklus	17
Abb. 12	Folgende DP-Zyklen	17
Abb. 13	Normaler DP-Zyklus bei Stay in Data-Exch (for WD-Time).....	18
Abb. 14	Erster gestörter und folgende DP-Zyklen bei Stay in Data-Exch (for WD-Time).....	18
Abb. 15	Diagnosevariablen des PROFIBUS-masters.....	21
Abb. 16	Slave-Diagnose - DP-State.....	24
Abb. 17	PC ist Sync-Master	27
Abb. 18	FC310x ist Sync-Master	28
Abb. 19	Counter und StartRedundancyMasterFlag	40
Abb. 20	TwinCAT 2.8 - Karteireiter FC310	41
Abb. 21	TwinCAT 2.8 - Karteireiter Profibus	43
Abb. 22	TwinCAT 2.8 - Karteireiter ADS.....	45
Abb. 23	TwinCAT 2.8 - Karteireiter General Diag.....	45
Abb. 24	TwinCAT 2.8 - Karteireiter Box States.....	46
Abb. 25	TwinCAT 2.8 - Karteireiter EquiDiag.....	47
Abb. 26	TwinCAT 2.9 - Karteireiter FC310x.....	49
Abb. 27	TwinCAT 2.9 - Dialog Bus-Parameter	50
Abb. 28	TwinCAT 2.9 - Dialog Fault-Settings	52
Abb. 29	TwinCAT 2.9 - Karteireiter MC.....	53
Abb. 30	TwinCAT 2.9 - Karteireiter ADS.....	54
Abb. 31	TwinCAT 2.9 - Karteireiter DP-Diag.....	55
Abb. 32	TwinCAT 2.9 - Karteireiter Box States.....	56
Abb. 33	TwinCAT 2.9 - Karteireiter MC-Diag	57
Abb. 34	Karteireiter Profibus	59
Abb. 35	Karteireiter Features	60
Abb. 36	Karteireiter Beckhoff	61
Abb. 37	Karteireiter ProcessData.....	62
Abb. 38	Karteireiter PrmData (Text).....	63
Abb. 39	Karteireiter Diag.....	63
Abb. 40	FC310x als Slave angefügt.....	65
Abb. 41	Module anfügen	66
Abb. 42	Features.....	66
Abb. 43	Aktivieren des ADS-Interfaces auf dem DP-Master.....	68
Abb. 44	Karteireiter ADS auf dem DP-Slave.....	69