



# CONTROL TECHNOLOGY FROM PARKER

## C3 I20 I32 T11 ControlManager FB45



UR	
1	
2	CPU 315-2 DP
X2	PROFIBUS DP
3	
4	
5	
6	
7	
8	
9	
10	
11	

PROFIBUS DP:



UR	
1	
2	CPU 317-2PN/DP
X1	DP (MPI)
X2	PN-IO
X2 P1	Port 1
X2 P2	Port 2
3	
4	
5	
6	
7	
8	
9	
10	
11	

PROFIBUS DP:

Ethernet: PROFINET-IO



AN0029  
C3 ControlManager FB45v6 SIMATIC 5.5

## Warranty Disclaimer

While efforts were made to verify the accuracy of the information contained in this documentation, Parker expressly disclaims all warranties with regard to this application note, including, but not limited to, the implied warranties of merchantability and fitness of a particular purpose. Parker does not warrant, guarantee, or make any representation regarding the use or the results of the use of this application note in terms of correctness, accuracy, or reliability. The contents of this application note are subject to change without notice. Parker will publish updates and revisions of this document as needed. The documents supersede all previous versions.

## Limitation of Liability

You agree that Parker shall not be liable to you under this agreement for any damages, including without limitation any lost profits, or any consequential, incidental, or punitive damages arising out of the use or inability to use this application note and related documents, or for any claim by another party. You agree and hold Parker harmless for all claims and damages from any third party as a result of their use or inability to use any product that you develop based on this application note and the products and/or services documented herein.

Parker Hannifin Manufacturing Germany GmbH & Co KG  
Electromechanical Automation  
Business Development & Applications  
Copyright © 2016

All Rights Reserved

## Table of Contents

<b>1. GERÄTE-ZUORDNUNG .....</b>	<b>4</b>
1.1. Geräteauswahl .....	4
1.2. übergeordneten Steuerung.....	4
<b>2. ZWECK DES BAUSTEINS .....</b>	<b>4</b>
2.1. Übersicht .....	4
2.2. Einschränkungen und Verwendung .....	4
2.3. Änderungshistorie .....	4
<b>3. EINSTELLUNGEN .....</b>	<b>5</b>
3.1. Compax3 Konfiguration .....	5
3.2. Compax3 Hardware .....	5
3.3. SIMATIC - HW Konfig .....	5
3.4. Anwenderschnittstelle des „C3ControlManager“ .....	7
3.5. Einstellungen für externen Master.....	10
<b>4. APPLIKATIONS-BEISPIEL.....</b>	<b>11</b>
4.1. Verbindungsübersicht: .....	11
4.2. Zyklischer Kanal (PZD).....	11
4.3. Azyklischer Kanal (PKW) .....	11
4.4. Applikation Positionieren.....	12
4.5. Parameterkanal .....	13

## 1. Geräte-Zuordnung

### 1.1. Geräteauswahl

- |   |     |     |     |
|---|-----|-----|-----|
| ▪ | F10 | F11 | F12 |
| ▪ | I20 | I32 |     |
| ▪ | T11 |     |     |

### 1.2. Übergeordnete Steuerung

- SIMATIC S7-300 SIMATIC S7-400
- mit integriertem PROFIBUS DP Master (z. B. CPU315-2DP) oder integriertem PROFINET IO Master (z. B. CPU315-2DP / CPU317-2PN)

## 2. Zweck des Bausteins

### 2.1. Übersicht

Absolut	Symbol	Kommentar	Vers	Datum	Gerät	Anwendung
FB45	C3ControlManager	C3 I20 T11 ControlManager	0.6	27.10.2014	C3 I20 T11	Positionieren, absolut / relativ

### 2.2. Einschränkungen und Verwendung

Der Baustein vereinfacht Steuerung des C3 T11 (mit PROFIdrive Profil) durch die S7. Dabei werden die Befehle und Sollwerte an den Baustein übergeben, dieser liefert Meldungen und Istwerte vom C3 zurück. Sowohl PZD als auch PKW wird in beiden Richtungen genutzt.

**Der Baustein ist für PPO13 ausgelegt; andere Typen sind nicht zulässig.**

### 2.3. Änderungshistorie

- V0.1 26.04.2004 21:31:13
- V0.2 19.12.2006 15:37:48
  - Beim Wechsel von Positionieren auf Handbetrieb ist kein Stop mehr nötig (STW1.4 = false und STW1.5 = false).
  - Wenn <bStartPositioning> nicht ausgeführt werden kann, wird gesichert, wieso es nicht ausgelöst wird. <bStartPositioning> (In\_Out Bit) wird in jedem Fall zurückgesetzt).
  - Relativ positionieren ist auch ohne Referenzlauf möglich.
  - Korrektur der Timer für Laufzeitüberwachung (Watchdog) diese liefen auch weiter, wenn die Bewegung nicht ausgeführt werden konnte (z. B. bei nicht bestromt).
- V0.3 28.06.2007 16:24:21
 

Flankenauswertung korrigiert für positionieren:

  - <bStartPositioning>: geht nur wenn vorherige Positionierung beendet
  - <bChangeSetImmediate>: kann laufende Bewegung unterbrechen
- V0.4 10.06.2009 08:45:45
  - <bPosRunning> ablöschen beim entstromen während der Fahrt, sonst muss bevor wiedereinschalten möglich ist bis <bPosErr> = True gewartet Werden!

### 3. Einstellungen

#### 3.1. Compax3 Konfiguration

Mit dem C3ServoManager werden folgende Einstellungen eingegeben:  
im Bereich:

```
\\ Kommunikation konfigurieren \ Profibus - Telegramm
[PROFIBUS – Betriebsart]
    Positionieren
```

```
[PLC -> Compax3]
    STW 1
    Lagesollwert XSOLL_A
    Soll-Verfahrdrehzahl D (32 Bit)
    Soll-Beschleunigung B (32 Bit)
```

```
[Compax3 -> PLC]
    ZSW 1
    Lageistwert XIST_A
    Drehzahlwert NIST_B
```

```
[Betriebsart-Einstellungen]
    Bedarfsdaten-Kanal / Parameterkanal
    „PKW ” muss gewählt werden
```

Fehlerreaktion bei Busausfall ist beliebig.

```
\\ I20T11 konfigurieren \ E/A - Belegung
Auswahl:
- frei
- feste Belegung
```

Es ist zu empfehlen „frei“ zu wählen. Bei „feste Belegung“ werden Binär-Verknüpfungen zwischen Steuerwort und digitalen Eingängen (E0 ... E3) erstellt, d.h. ohne die Anschlüsse an X12 ist das C3 nicht steuerbar.

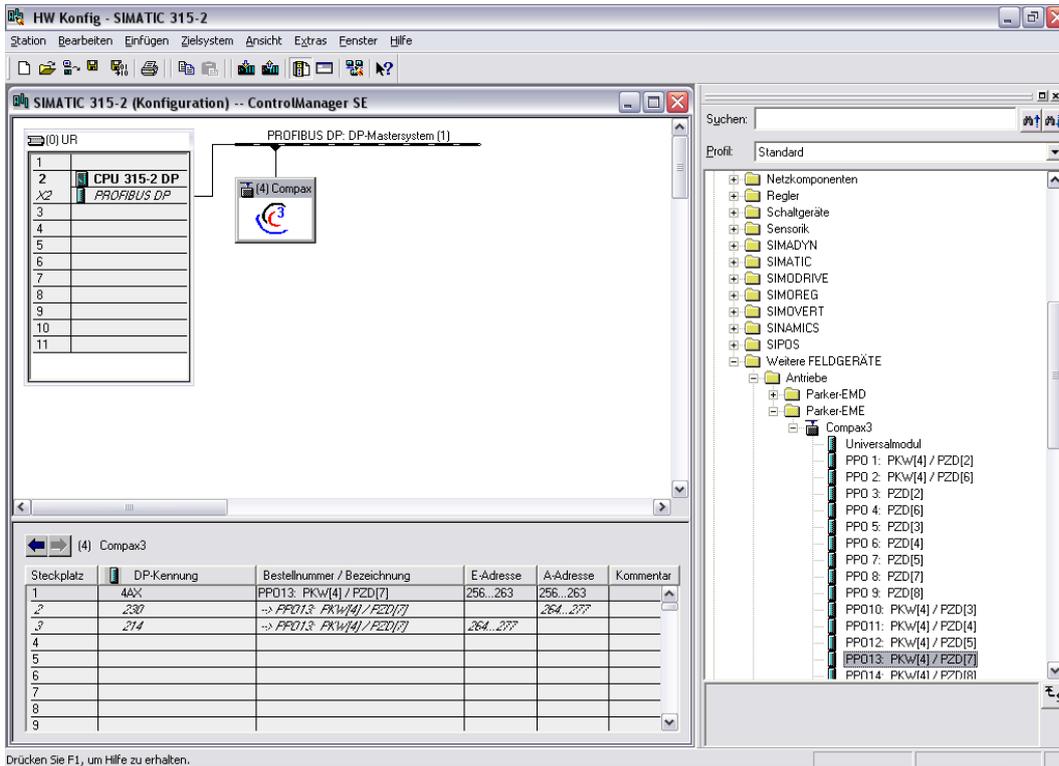
#### 3.2. Compax3 Hardware

DIP-Schalter: Busadresse

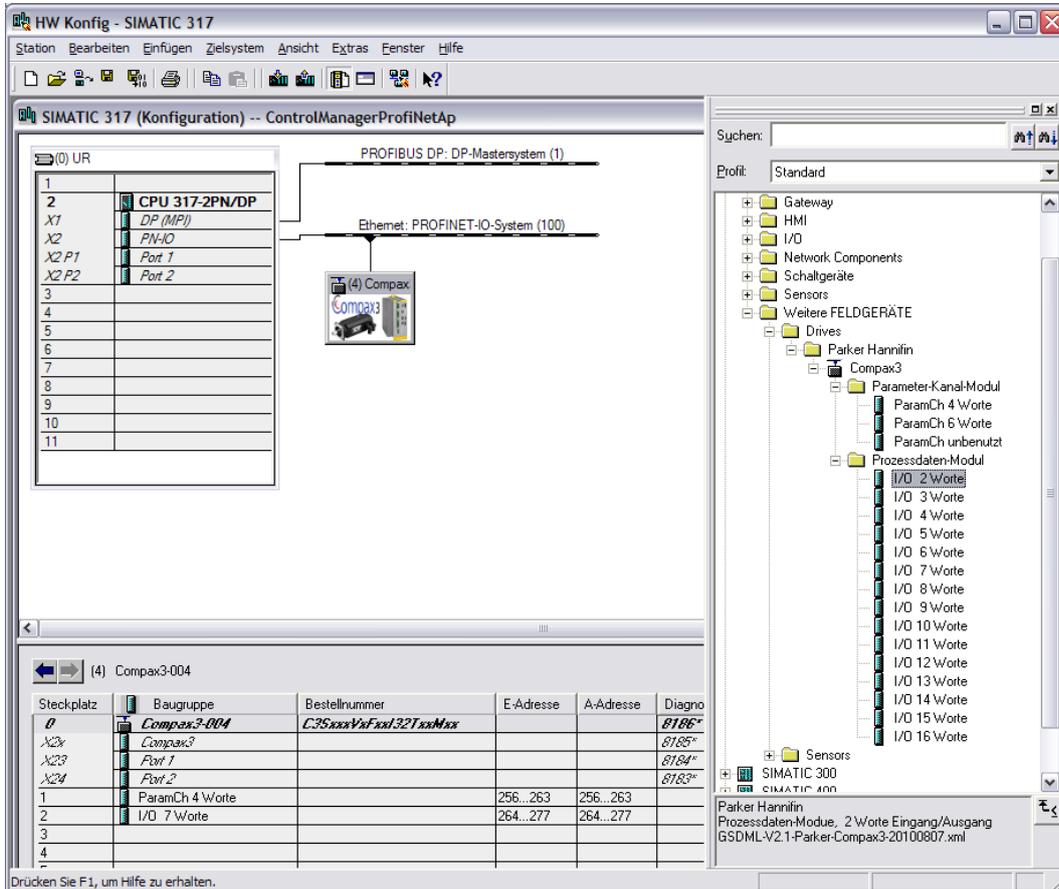
Busanschluß-Stecker: „ON / OFF“ Bus-Abschluss-Widerstand

#### 3.3. SIMATIC - HW Konfig

Analog zum PPO-Typ, der in der C3 Konfiguration angezeigt wird (hier PPO13) ist in *SIMATIC - HW Konfig* der entsprechende Typ auszuwählen:



pic 1: SIMATIC - HW Konfig



pic 2: SIMATIC - HW Konfig

Die Startadresse des PKW (hier 256) in Instanz-Variable <nLaddr> (DB45.DBW6) eintragen.

### 3.4. Anwenderschnittstelle des „C3ControlManager“

#### 3.4.1. Schematische Darstellung der Ein- und Ausgänge

#### Bereiche des FB45 / DB45

in	Eingang	out	Ausgang	in_out	Ein-/Ausgang	stat	Speicher
----	---------	-----	---------	--------	--------------	------	----------

#### DB45

	EN	ENO	
DBX0.0	bEnable	bEnabled	DBX2.0
DBX0.1	bAbsoluteRelative	bDriveErr	DBX2.1
DBX0.2	bPositionResetMode	bPosRunning	DBX2.2
DBX0.3	bHold	bInPosition	DBX2.3
DBX0.4	bStop	bPosErr	DBX2.4
DBX0.5	bFaultReset	bHomingRunning	DBX2.5
DBX0.6	bJogP	bHomingAttained	DBX2.6
DBX0.7	bJogN	bHomingErr	DBX2.7
DBX1.0	bExDataTransfer	bCommErr	DBX3.0
DBX4.0	bStartPositioning		
DBX4.1	bChangeSetImmediate	stRd.iPositionValue	DBD46
DBX4.2	bStartHoming	stRd.iVelocityValue	DBD50
DBW6	nLaddr	stRd.nActualError	DBW54
DBD8	iPosition		
DBD12	iVelocity		
DBD16	iAcceleration		
DBD20	iDeceleration		
DBD24	iInPosWindowAbs		
DBW28	nCmd	bTransErr	DBX3.1
DBW30	nPNUIndex		
DBW32	nPNUSubindex		
DBD34	iParameterValue	iParameterValue	DBD34
DBD38	TonTimer1		
DBD42	TonTimer2		
DBX56.0 Word 4	stC3PKWInDint.nPKE	stC3PKWOutDint.nPKE	DBX64.0 Word 4
DBX72.0 Word 7	stC3PZDIn.nStatus	stC3PZDOut.nControl	DBX86.0 Word 7

#### 3.4.2. Deklaration der Ein- und Ausgänge

Parameter	Deklaration	Datentyp	Beschreibung
bAbsoluteRelative	IN	BOOL	=0 absolute, =1 relative Bewegung
bEnable	IN	BOOL	=1 Bestromen =0 Entstromen über AUS3-Rampe (Not-Stopp)
bExDataTransfer	IN	BOOL	=0 interne DP Schnittstelle mit SFC14/15 (interner Master in S7 CPU) =1 externe DP Schnittstelle mit FC2/FC1 (externer Master CP 342-5)
bFaultReset	IN	BOOL	positive Flanke quittiert, danach ist erneut das <bEnable> zu aktivieren (bedingt durch die Flankenbewertung, muß dieses zunächst auf 0 gestellt werden)
bHold	IN	BOOL	=1 Zwischenhalt (Verfahrsatz bleibt erhalten), =0 weiterfahren
bJogN	IN	BOOL	Tippen negativ: Handbetrieb - Fahren innerhalb der Endgrenzen - so lange wie der Eingang betätigt wird!
bJogP	IN	BOOL	Tippen positiv: Handbetrieb - Fahren innerhalb der Endgrenzen - so lange wie der Eingang betätigt wird!
bPositionResetMode	IN	BOOL	=0 Normal-, =1 Rücksetzbetrieb gewählt (im C3Mgr \I20T11 konfigurieren \Bezugssystem wurde Rücksetzstrecke Zähler, Nenner gewählt)
bStop	IN	BOOL	=1 Stopp (Fahrauftrag verwerfen)
bChangeSetImmediate	IN_OUT	BOOL	positive Flanke übernimmt neues Positionierbewegung, <bChangeSetImmediate> wird vom FB selbst zurückgesetzt Dieser Befehl wird durch bPosRunning quittiert. Ein neuer Befehl wird auch angenommen,

Parameter	Deklaration	Datentyp	Beschreibung
bStartHoming	IN_OUT	BOOL	wenn die vorherige Bewegung nicht beendet wurde (<bInPosition> =true). positive Flanke startet Referenzierung, negative Flanke stoppt Referenzierung, <bStartHoming> sollte erst mit <bHomingAttained> zurückgesetzt werden
bStartPositioning	IN_OUT	BOOL	positive Flanke startet Positionierung, <bStartPositioning> wird vom FB selbst zurückgesetzt. Dieser Befehl wird durch bPosRunning quittiert. Ein neuer Befehl wird nur angenommen, wenn die vorherige Bewegung beendet wurde (<bInPosition> =1).
bCommErr	OUT	BOOL	=1 Kommunikationsfehler mit c3 (Fehler aus SFC14/ SFC15) (alle sonstigen Anzeigen sind eventuell ungültig)
bDriveErr	OUT	BOOL	=1 Fehler (Gerät/Motor)
bEnabled	OUT	BOOL	=1 Achse bestromt =0 Achse nicht bestromt
bHomingAttained	OUT	BOOL	=1 Referenz o.k.
bHomingErr	OUT	BOOL	=1 Überwachung Laufzeit für Referenz -fahrt (ggf. TonTimer2 korrigieren)
bHomingRunning	OUT	BOOL	=1 Referenzfahrt aktiv
bInPosition	OUT	BOOL	=1 Achse in Zielposition
bPosErr	OUT	BOOL	=1 Überwachung Laufzeit für Positionier -auftrag (ggf. TonTimer1 korrigieren)
bPosRunning	OUT	BOOL	=1 Fahrauftrag ist aktiv
bTransErr	OUT	BOOL	=1 Format-, Kommandofehler bei Übertragung zu C3
iAcceleration	STATIC	DINT	Beschleunigung im U32-Format (Ganzzahl)
iDeceleration	STATIC	DINT	Verzögerung im U32-Format (Ganzzahl) iDeceleration wird bei Änderung auf dem PKW-Kanal übertragen
iInPosWindowAbs	STATIC	DINT	Positionsfenster im C4_3-Format (3 dezimale Nachkommastellen), zusätzliche Überwachung der Meldung „InPosition“ vom C3. Der aktuelle Lage-Istwert wird über dieses Positionierfenster mit dem Sollwert verglichen. Das ist nur bei Absolut-Positionierung (<bAbsoluteRelative> = false) und nicht aktiviertem Rücksetzbetrieb (<bPositionResetMode> = false) möglich (z. B. Standardwert = 1000 entspricht 1 u ; u = Einheit aus C3 Mgr Inkr, mm, Grad, Inch).
iParameterValue	STATIC	DINT	ProfiDrive Parameter Transfer: Wert (Quelle und Ziel) - Schreib-/Lesewert des Parameters, bei INT-/WORD-Format rechtsbündig
iPosition	STATIC	DINT	Zielposition/Distanz im C4_3-Format (3 dezimale Nachkommastellen)
iVelocity	STATIC	DINT	Geschwindigkeit im C4_3-Format (3 dezimale Nachkommastellen)
stRd.iPositionValue	STATIC	DINT	Istposition im C4_3-Format (3 dezimale Nachkommastellen)
stRd.iVelocityValue	STATIC	DINT	Istgeschwindigkeit im C4_3-Format (3 dezimale Nachkommastellen)
nCmd	STATIC	INT	ProfiDrive-Parameter Transfer: Befehl: 1 lesen 2 schreibe WORD 3 Schreibe DWORD
nPNUIndex	STATIC	INT	PROFIdrive-Parameter Transfer: PNU-Index
nPNUSubindex	STATIC	INT	ProfiDrive-Parameter Transfer: PNU-Subindex (keine Erhöhung des Wertes bei DPV0 notwendig)
TonTimer1	STATIC	TIME	Zeitwert für Überwachung der Positionierung
TonTimer2	STATIC	TIME	Zeitwert für Überwachung der Referenzierung
nLaddr	STATIC	WORD	= 0100 <sub>hex</sub> (=256 <sub>dez</sub> ) Startadresse C3-Slave aus der SIMATIC HW Konfig, notwendig bei <bExdataTransfer> = 0
stRd.nActualError	STATIC	WORD	Aktueller Fehler (siehe C3-Dokumentation) im WORD - Format =1 kein Fehler!
stC3PKWInDint.nPKE	STATIC	Word 4	Lokales PKW - Eingangsfeld für externen CP
stC3PKWOutDint.nPKE	STATIC	Word 4	Lokales PKW - Ausgangsfeld für externen CP
stC3PZDIn.nStatus	STATIC	Word 7	Lokales PZD - Eingangsfeld für externen CP
stC3PZDOut.nControl	STATIC	Word 7	Lokales PZD - Ausgangsfeld für externen CP

### 3.4.3. Vorgehensweise

#### 3.4.3.1. Einstellung der statischen Operanden am Baustein

##### 1. <nLaddr>

- Parameter aus *HW Konfig*: die erste Startadresse des C3-Slaves am PROFIBUS.
- Der Baustein errechnet intern die Felder für Ein- und Ausgangsbereich, aus den Längen die in PPO13 festgelegt wurden.
- Die Adressen für Ein- und Ausgangsbereich müssen die gleichen Werte haben es dürfen keine Lücken zwischen PZD und PKW sein.
- Die Reihenfolge muß so sein, dass zuerst der PKW- und danach der PZD Bereich gewählt wird.

- Beispiel:

	E-Adresse	A-Adresse
PKW	256 .. 263	256 .. 263
PZD	264 .. 277	264 .. 277

- Hier muß der Wert „256“ an `<nLaddr>` übergeben werden.
2. `<BPositionResetMode>`
    - muß gesetzt werden, sofern in der C3-Konfiguration eine Rücksetzstrecke definiert wurde.
  3. `<bExDataTransfer>`
    - Low: Einstellung für (in CPU) integrierte Profibus-Master-Schnittstelle
    - High: Einstellung für externen Master
    - Siehe „Einstellungen für externen Master“
  4. `<iInPosWindowAbs>`
    - Überwachungsfenster für die Meldung `bInPosition`
  5. `<TonTimer1>`
    - Zeitwert für Überwachung der Positionierung, sofern diese Zeit nicht ausreicht die Positionierung auszuführen, wird die Meldung `bPosError` gesetzt.
  6. `<TonTimer2>`
    - Zeitwert für Überwachung der Referenzierung, sofern diese Zeit nicht ausreicht, um die Positionierung auszuführen, wird die Meldung `<bHomingErr>` gesetzt.

### 3.4.3.2. Einstellung der dynamischen Operanden am Baustein

1. einschalten
  - `<bEnable>`: setzen, der Baustein meldet `<bEnabled>`
  - `<bStartHoming>` setzen, der Baustein meldet `<bHomingRunning>`. Nach Beendigung der Referenzierung meldet der Baustein `<bHomingAttained>`.
  - Nun kann `<bStartHoming>` wieder zurückgesetzt werden.
  - Mit Erreichen der Referenzposition wird die Meldung `<bInPosition>` gesetzt.
  - Bei einigen Maschinen-Nullmodi (z.B. 35) und hoher Geschwindigkeit kann `<bHomingRunning>` so kurz gesetzt sein, dass es nicht gesehen werden kann.
  - Sofern ein Motor mit Absolutwertgeber (SinCos©) verwendet wird, genügt die einmalige Aktivierung der MN-Suchfahrt: `<bHomingAttained>` bleibt nach aus- und wieder Einschalten erhalten. **Achtung:** durch die Aktivierung der MN-Suchfahrt aus der C3-Optimierung wird `<bHomingAttained>` nicht gesetzt.
2. Positionieren
  - Zunächst Parameter für Positionen einstellen:
    - `<bAbsoluteRelative>`
    - `<iPosition>`
    - `<iVelocity>`
    - `<iAcceleration>`
    - `<iDeceleration>`
  - danach die Positionierung durch Aktivierung von `<bStartPositioning>` setzen,
    - `<bStartPositioning>` wird durch den Baustein selbständig zurückgesetzt
    - Der Baustein meldet `<bPosRunning>`.
    - Bei Erreichen der Zielposition wird die Meldung `<bInPosition>` gesetzt.
    - eine neue Positionierung kann erst nach Beendigung der vorherigen gestartet werden.
  - Für dynamische Positionierung kann der Eingang `<bChangeSetImmediate>` verwendet werden. Dieser ist wie `<bStartPositioning>` zu verwenden. Jedoch wird eine neue Positionier-Aufgabe auch übernommen, sofern die vorherige Bewegung nicht beendet wurde.

### 3.4.3.3. sonstige Operanden am Baustein

- `<bFaultReset>` dient zum quittieren von Laufzeitfehlern und C3-Fehlern.
- `<bStop>` stoppt einen Positionierauftrag mit der steigenden Flanke.
- `<bHold>` unterbricht einen Positionierauftrag, solange das Bit gesetzt ist, der Auftrag wird fortgesetzt, sobald das Bit zurückgesetzt ist.
- `<bJogN>` Handbetrieb in negativer Richtung, solange das Bit gesetzt ist.
- `<bJogP>` Handbetrieb in positiver Richtung, solange das Bit gesetzt ist.

#### 3.4.3.4. Meldungen und Anzeigen

- `<bCommErr>` Verbindung zum C3 nicht möglich
- `<bDriveErr>` C3 ist im Fehlerzustand
- `<stRd.nActualError>` aktuelle Fehlernummer des C3 (Erläuterungen siehe C3 Hilfe)
- `<stRd.iPositionValue>` aktuelle Istposition
- `<stRd.iVelocityValue>` aktuelle Istgeschwindigkeit

#### 3.4.3.5. Parameter lesen und ändern

- Parameter können gelesen oder geschrieben werden. Parameter sind in der Tabelle „Objekte für den Parameterkanal“ aufgeführt (siehe C3 Hilfe). Darin ist die Zuordnung zwischen Objektnummern (C3) und PNU (PROFIdrive-Profile) enthalten.
- Der Parameter wird durch `<nPNUIndex>` und `<nPNUSubindex>` bestimmte
- Der Wert ist in `<iParameterValue>`
- `<nCmd>` dient zum Starten des Transfers.
  - 1 steht für Lesen
  - 2 für schreiben von Wortparametern (16 Bit)
  - 3 für schreiben von Doppelwort-Parametern (32 Bit)
 Ob der Parameter 16 Bit oder 32 Bit umfasst ist in der Tabelle „Objekte für den Parameter -Kanal“ aus der Spalte „Busformat“ ersichtlich.
- `<bTransErr>` meldet einen Fehler bei der Datenübertragung

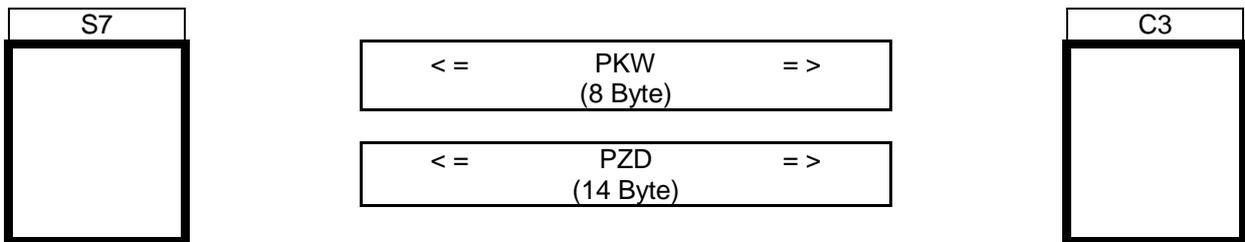
### 3.5. Einstellungen für externen Master

1. Der Baustein ist für S7-CPU's mit integriertem PROFIBUS DP - Master geeignet, ohne diesen ist der Betrieb des Bausteins nicht möglich.
2. sofern diese Schnittstelle für andere Verbindungen genutzt wird, und die Verbindung zum COMPAX3 durch einen externen Kommunikationsprozessor (CP342-5) erfolgen soll, kann der Baustein unter Beachtung folgender Hinweise verwendet werden:
  - Bei Verwendung eines CP 342-5 müssen mit FC1 / FC2 (DP\_SEND / DP\_RECV, SIMATIC Standardbibliothek) die Daten übertragen werden.
  - An DP\_SEND wird das globale Ausgangsfeld als Parameter angegeben, an DP\_RECV wird das globale Eingangsfeld als Parameter angegeben.
  - Achtung: Diese globalen Felder (DB InOut) enthalten sämtliche am Bus angeschlossenen DP - Slaves.
  - Der dem betreffenden COMPAX3 zugeordneter Bereich (lokales Eingangsfeld / lokales Ausgangsfeld), muß manuell an die Stelle im DB45 übertragen werden (z. B. durch SFC20 BLKMOV).
  - Ermitteln des lokalen Eingangsfelds / des lokalen Ausgangsfelds anhand der *SIMATIC - HW Konfig.*
3. Lokales Eingangsfeld: `<StC3PKWInDint.nPKE>`
4. Lokales Ausgangsfeld: `<StC3PKWOutDint.nPKE>`

## 4. Applikations-Beispiel

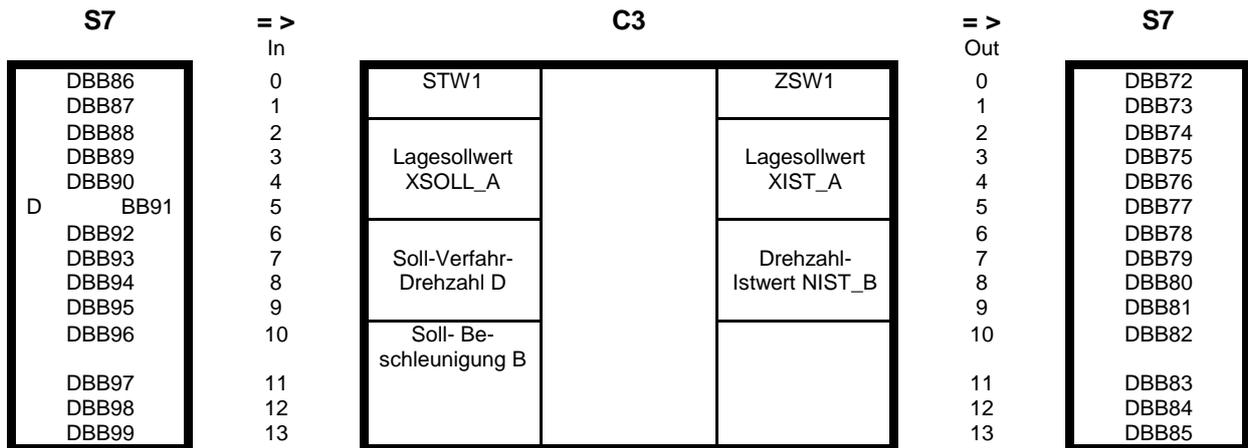
### 4.1. Verbindungsübersicht:

Verbindung einer SIMATIC S7 300 als PROFIBUS DP Master/ProfiNET IO Master und einem C3 I20/I32 T11 als PROFIBUS DP/ ProfiNET IO Slave.



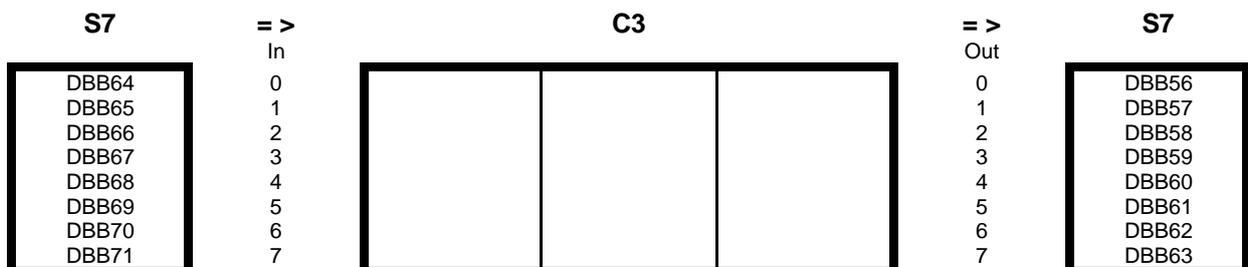
### 4.2. Zyklischer Kanal (PZD)

Die Ein- und Ausgangsparameter sind im C3 im Bereich *Kommunikation* einzustellen. Grundsätzlich werden nur Variablen im Word- oder Doppelwort-Format übertragen. Die Einstellung der Variablen ist für den FB zwingend so vorgeschrieben:



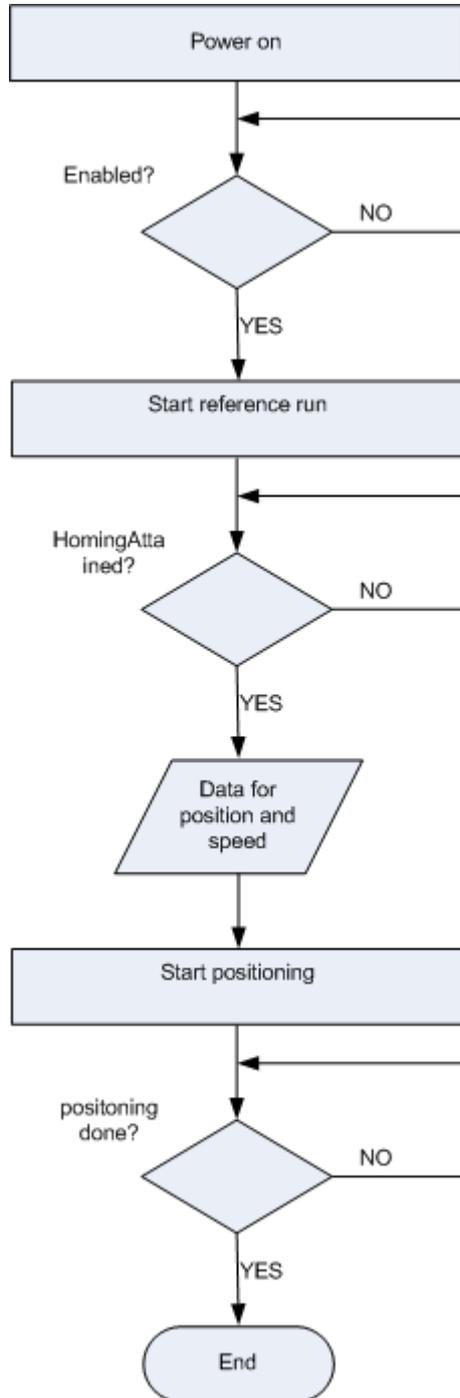
### 4.3. Azyklischer Kanal (PKW)

Über die 8 Byte PKW-Schnittstelle können Variablen azyklisch an das C3 übertragen werden. Über diese Schnittstellen kann der Anwender Parameter ändern oder lesen. Der Baustein nutzt diese Schnittstelle zur Übertragung von „Verzögerung“ (deceleration – Bremsrampe) und Fehlernummer.



#### 4.4. Applikation Positionieren

Im S7 Programm wird der FB45 aufgerufen und mit den geforderten Parametern für eine Positionierung versorgt. Der Ablauf ist in folgendem Bild dargestellt:



pic 3

Die Schrittkette ist im FC400. Durch setzen des Merkers

M 0.4 "start"

Auf TRUE wird diese gestartet.

Mit dem Merker

M 0.5 "reset"

kann die Schrittkette zurückgesetzt werden.

## 4.5. Parameterkanal

Über die SPS (HMI) soll gelegentlich der Wert für den Momenten-Istwert [683.1] gelesen werden, sowie die Steifigkeit [2100.2] geändert wird. Die Eingabe wird mit der SIMATIC Variablen-tabelle erläutert.

### 4.5.1. Zugriffsbereich

		FB45	
DBW28	NCmc		BTransErr
DBW30	NPNUIndex		
DBW32	NPNUSubindex		
DBD34	ParameterValue	ParameterValue	
			DBX3.1
			DBD34

### 4.5.2. Vorgehensweise zum Lesen des Momenten-Istwerts

1. Momenten-Istwert in der Tabelle „Objektübersicht nach Objektnamen sortiert“

Objekt-Adresse	Objektname	Objekttyp
681.5	Status Ist-Geschwindigkeit ungefiltert	C3.StatusSpeed_Actual
680.5	Status Ist-Position	C3.StatusPosition_Actual
688.2	Status Ist-Strom effektiv (momentenbildend)	C3.StatusCurrent_Actual
683.3	Status Langzeit-Motorauslastung	C3.StatusDevice_ActualMotorLoad
684.2	Status Motortemperatur	C3.StatusTemperature_Motor
681.6	Status Regeldifferenz Geschwindigkeit	C3.StatusSpeed_Error
680.6	Status Schleppfehler	C3.StatusPosition_FollowingError
682.4	Status Soll-Beschleunigung	C3.StatusAccel_DemandValue
681.10	Status Soll-Geschwindigkeit Regler-Eingang	C3.StatusSpeed_DemandSpeedController
681.4	Status Soll-Geschwindigkeit Sollwertgeber	C3.StatusSpeed_DemandValue
680.4	Status Soll-Position	C3.StatusPosition_DemandValue
680.12	Status Soll-Position ohne Absolutbezug	C3.StatusPosition_DemandController
688.1	Status Soll-Strom effektiv (momentenbildend)	C3.StatusCurrent_Reference
<b>683.1</b>	<b>Status Stromistwert</b>	<b>C3.StatusDevice_ActualCurrent</b>
682.7	Status Vorsteuerung Beschleunigung	C3.StatusAccel_FeedForwardAccel
681.11	Status Vorsteuerung Geschwindigkeit	C3.StatusSpeed_FeedForwardSpeed
685.2	Status Zwischenkreisspannung	C3.StatusVoltage_BusVoltage
2100.2	Steifigkeit (Drehzahlregler)	C3.ControllerTuning_Stiffness
1100.4	Steuerwort 2	C3Plus.DeviceControl_Controlword_2
1100.3	Steuerwort STW	C3Plus.DeviceControl_Controlword_1
2010.4	Stromvorsteuerung	C3.FeedForward_Current

pic 4

2. Objektbeschreibung aufschlagen:

#### Status Stromistwert (Objekt)

<b>Profibus-Nr.</b>	<b>PNU:112</b>	<b>Objektnr.</b>	<b>683.1</b>
<b>Objektname</b>	<b>C3.StatusDevice_ActualCurrent</b>		
<b>Maßeinheit</b>	%	<b>Zugriff:</b>	read only
<b>Busformat:</b>	E2_6	<b>gültig nach:</b>	-
<b>Minimalwert</b>	0 n/a	<b>Maximalwert</b>	-- n/a
<b>Anmerkung:</b>	Stromistwert (Momentenistwert) Bezug ist der Motornennstrom		

pic 5

3. hieraus ergeben sich die Daten, die eingetragen werden müssen:

nPNUIndex <112>  
 nPNUSubindex <0>

4. Diese beiden Werte eintragen und „Steuerwerte aktivieren“ (siehe “Variable”).

5. Anschließend das Kommando für „Lesen“ (1) eintragen!

nCmd <1>

6. Nun nochmals „ Steuerwerte aktivieren“

	Operand	Symbol	Symbolkommentar	Anzeige	Statuswert	Steuerwert
1		// interface for parameter transfer				
2						
3		// Stat for input				
4	DB45.DBW 28	"IDB ControlManager SE".nCmd	=0 do nothing, self-resetting	DEZ	0	
5	DB45.DBW 30	"IDB ControlManager SE".nPNUIndex	PNU index	DEZ	112	
6	DB45.DBW 32	"IDB ControlManager SE".nPNUSubindex	PNU subindex	DEZ	0	
7						
8		// Stat for Input Output				
9	DB45.DBD 34	"IDB ControlManager SE".iParameterValue	parameter value	DEZ	L#1199	
10						
11		// Stat for Output				
12	DB45.DBX 3.1	"IDB ControlManager SE".bTransErr	=1 error due to parameter transmitting	BOOL	true	
13						

pic 6

hier wurde in

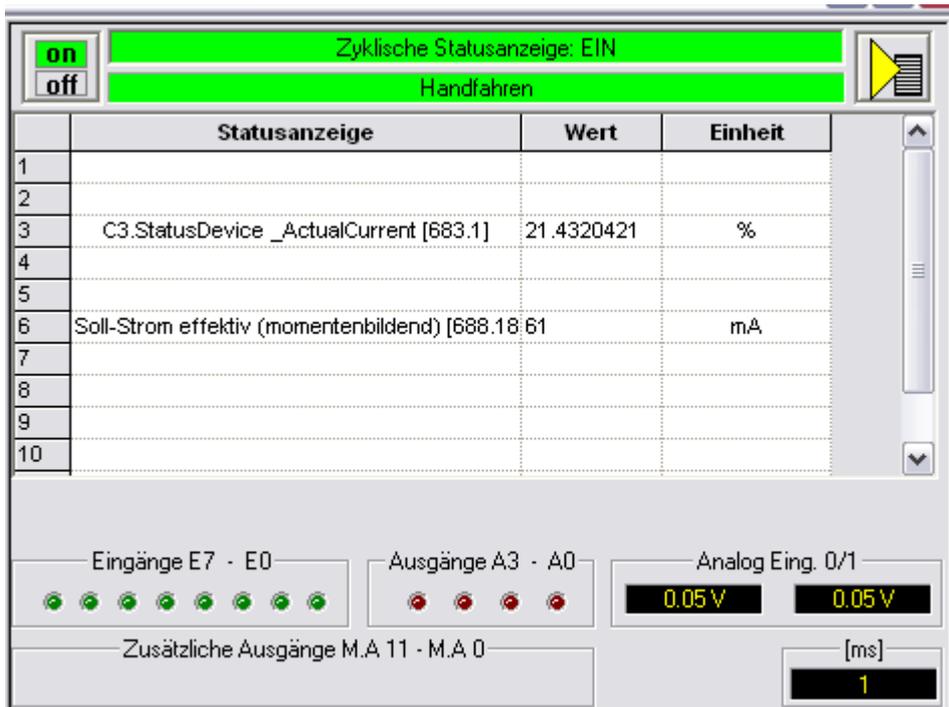
iParameterValue <1199>

eingetragen.

7. Das Busformat ist **E2\_6** (siehe pic 4) was bedeutet, dass der Wert durch 64 geteilt werden muß. Die Anzeige ist in Prozent bezogen auf den Nennstrom.

$$1199 / 64 = \underline{18,7}$$

8. zur Kontrolle vergleichen wir mit dem Status in der Optimierung:



pic 7

#### 4.5.3. Vorgehensweise zum Schreiben der Steifigkeit

##### 1. Suche der Parameter

Name: Steifigkeit (Drehzahlregler C3.ControllerTuning\_Stiffness)  
 Objektnummer: 2100.2  
 Busformat: U16  
 Maßeinheit: %  
 nPNUIndex: <402>  
 nPNUSubindex: <2>

##### 2. Übertragen der Werte mittels Variablen-tabelle

nPNUIndex: <402>  
 nPNUSubindex: <2>

der Default-Wert für die Steifigkeit ist 100 % eine Erhöhung um 10 % ergibt den Wert „110“:

iParameterValue: <110>

„Steuerwerte aktivieren“ (siehe „Variable“).

##### 3. Aktivieren des Kommandos:

Da das Busformat „U16“ ein Wort-Format ist, ist als Kommando zum Schreiben „2“ erforderlich (bei Doppel-Wort-Formaten ist „3“ einzutragen).

nCmd: <1>  
 „Steuerwerte aktivieren“ (siehe „Variable“).

##### 4. Dieses Objekt ist damit zwar übertragen, muß nun aber durch den Befehl "Objekte gültig setzen" aktiviert werden. (siehe *gültig* nach *VP*).

5. Dazu muß nun das Objekt "Objekte gültig setzen" (VP Objekt 210.10 C3.*ValidateParameter\_Global*) geschrieben werden. Die Vorgehensweise ist die gleiche wie bei der Steifigkeit.
6. Die geänderten Werte sind nun zwar gültig, aber nicht nullspannungssicher gespeichert. Zum Speichern auf das *Flash* dient der Befehl *Write Flash* (WF Objekt 20.1 "C3.ObjectDir\_Objekts-->FLASH").

**Anmerkung:**

VP und WF nutzen Systemressourcen, die die interne Kommunikation verlangsamen, dies kann u. U. zu Fehlermeldungen wie „Zykluszeit-Überschreitung“ führen. Es ist zu empfehlen, diese Befehle im Prozessbetrieb nur sparsam zu verwenden (nur wenn unbedingt erforderlich, eventuell alle Parameter übertragen und dann VP aktivieren).