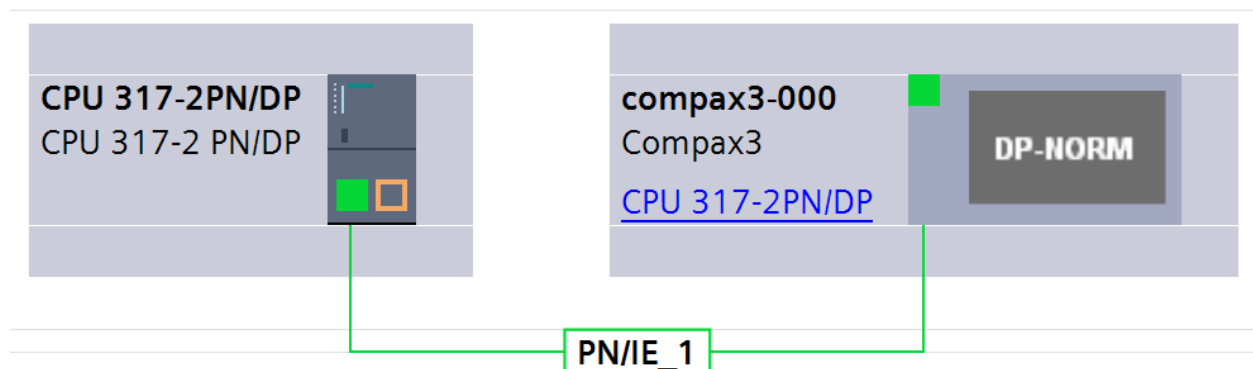




CONTROL TECHNOLOGY FROM PARKER

C3 I20 I32 T11 ControlManager FB45



AN_0029
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

1.	DEVICE SUPPLEMENT.....	4
1.1.	With the option.....	4
1.2.	And the Master plc.....	4
2.	PURPOSE OF THE BLOCK	4
2.1.	Overview.....	4
2.2.	Restrictions and application.....	4
2.3.	History of modifications.....	4
3.	ADJUSTMENTS.....	5
3.1.	Compax3 configuration.....	5
3.2.	Compax3 Hardware	Fehler! Textmarke nicht definiert.
3.3.	3.3 SIMATIC - HW Config	6
3.4.	Application interface of "C3ControlManager"	8
3.5.	settings for external Master	11
4.	APPLICATION EXAMPLE	11
4.1.	Overview of the connection:	11
4.2.	cyclic channel (PZD).....	12
4.3.	Acyclic channel (PKW)	12
4.4.	Application Positioning.....	12
4.5.	Parameter channel.....	14

1. Device supplement

1.1. With the option

- F10 F11 F12
- I20 I32
- T11

1.2. And the Master plc

- SIMATIC S7-300 SIMATIC S7-400
- with integrated PROFIBUS DP Master (z. B. CPU315-2DP) or with integrated PROFINET IO Master (z. B. CPU315-2DP / CPU317-2PN)

2. Purpose of the Block

2.1. Overview

Absolute	Symbol	Comment	Vers	Date	Device	Application
FB45	C3ControlManager	C3 I20 T11 ControlManager	0.6	2014-10-27	C3 I20 T11	Positioning, absolute / relative

2.2. Restrictions and application

This block simplifies the control of a C3 T11 (with PROFIdrive profile) with the S7. The block needs the commands and set values. The block distributes messages and actual values from the drive. The channels PZD and PKW are used in both directions.

This block is only usable with PPO13 others are strictly forbidden.

2.3. History of modifications

1. V0.1 2004-04-26 09:31:13 PM
2. V0.2 2006-12-19 03:37:48 PM
 - At change from positioning to manual (JOG) is no longer necessary to choose Stop first (STW1.4 = false and STW1.5 = false).
 - If <bStartPositioning> is not possible to be done, the block will save the reason. <bStartPositioning> (InOut Bit) is resettled in any case
 - Relative positioning is now possible without having done homing first.
 - Correction of Timer for Watchdog this was not stopped during a movement was not possible (e.g. „not energized“).
3. V0.3 2007-06-28 04:24:21 PM
 - Rising edge deception was corrected at positing
 - <bStartPositioning>: is only possible if the actual movement has react the target
 - <bChangeSetImmediate>: can interrupt actual move (new target without stop)
3. V0.4 2009-06-10 08:45:45 AM
 - Delete <bPosRunning> during disabling power stage. Otherwise it would necessary to wait until <bPosErr> = True before a new order is accepted!

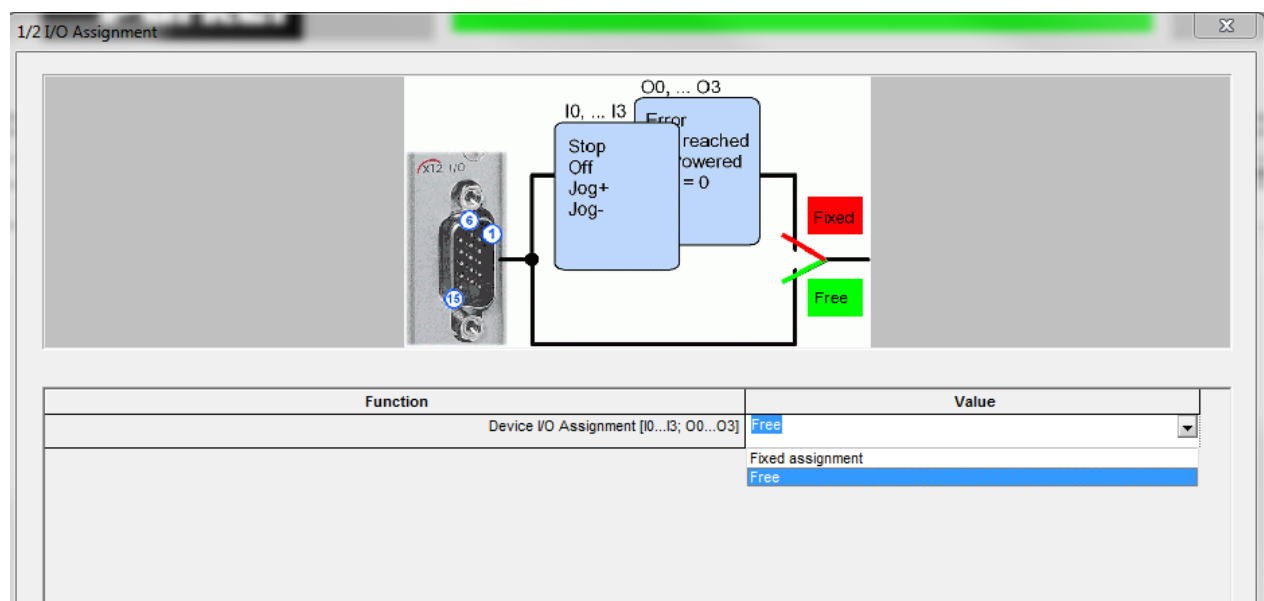
3. Adjustments

3.1. Compax3 configuration

With C3ServoManager few following adjustments:

	Profinet
ProfiNet Betriebsart	Positioning
	PLC ----> Compax3
PAD 1	Control word 1 [1100.3] p1 w1
PAD 2	Commanded position value XSOLL_A [1100.6] p 27 w2
PAD 3	Commanded speed D [1100.7] p111 w2
PAD 4	Commanded acceleration B [1111.3] p114 w2
	Compax3 ----> PLC
PED 1	Status word 1 [1000.3] p w1
PED 2	Actual position value XIST_A [680.5] p28 w2
PED 3	Actual speed value NIST_B [681.9] p8 w2
Acyclic channel	Parameter channel [4 words]
Profinet error reaction (0x8130)	2 - Stop, drive disabled
PPO type	Parameter channel [4 words] I/O 7 words (PPO13)

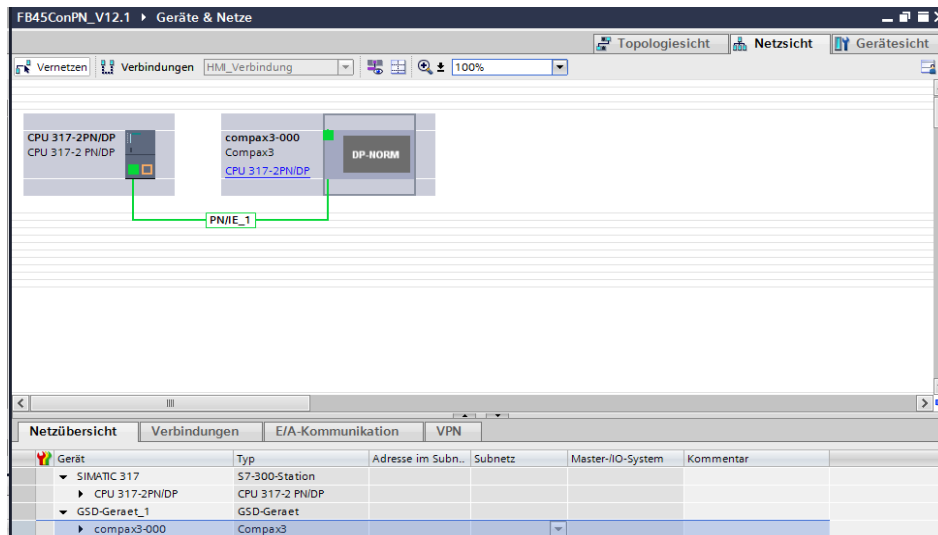
I20T11 drive configuration \ I/O – assignment



It is recommended to select „free“. With „Fixed assignment“ are used some Binary connections between control word and digital inputs (E0 ... E3), that cases if there is no plug at X12 the C3 is not possible to be controlled.

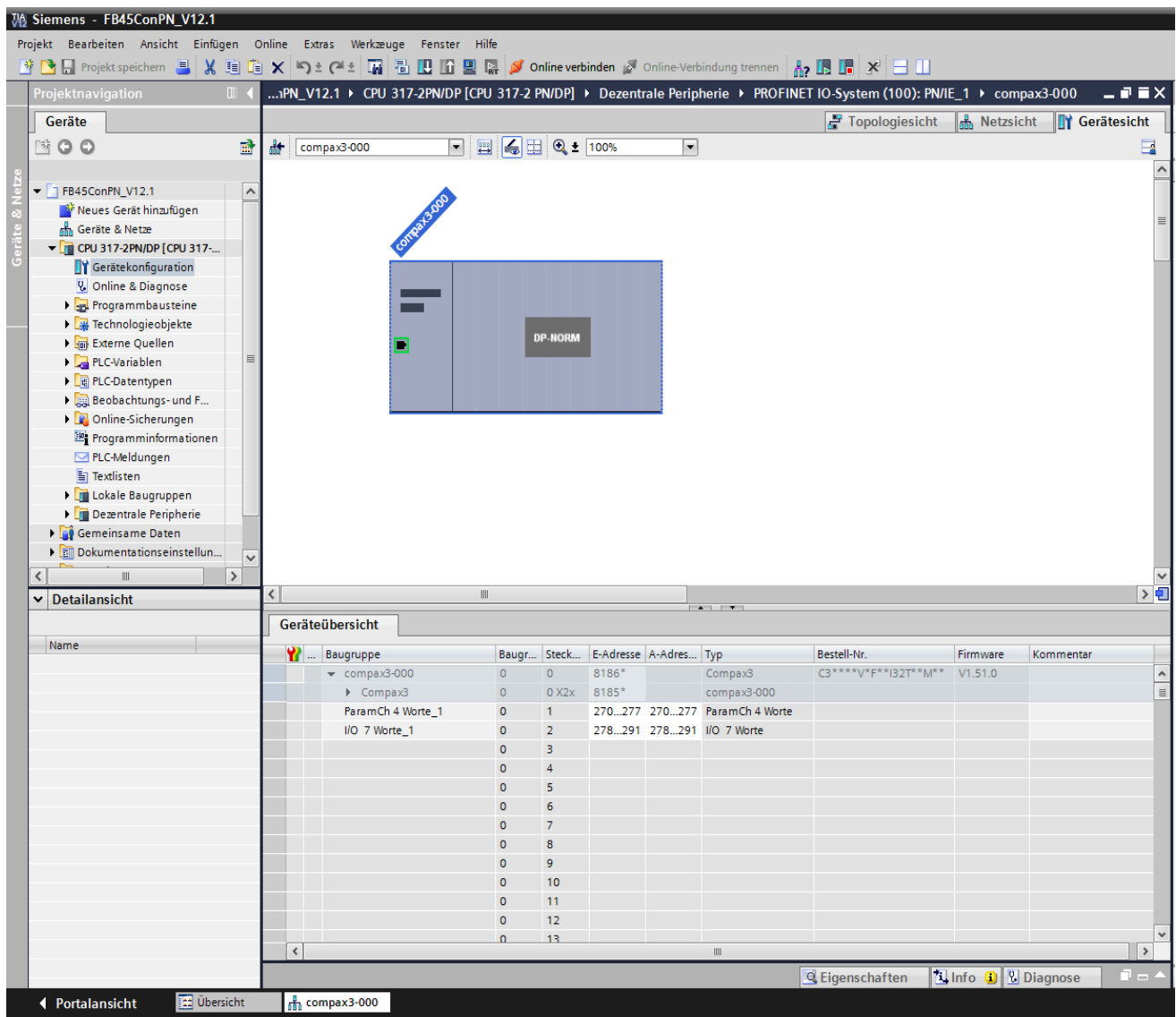
3.2. SIMATIC - HW Config

Add Compax3 from Hardware Catalog



Assign IP-address and device name

The screenshot shows the 'compax3-000 [Module]' Properties window. The 'General' tab is active, showing the 'PROFINET interface [X1]' configuration. The 'Ethernet addresses' section shows the interface connected to Subnet: PN/IE_1. The 'IP protocol' section shows 'Use IP protocol' checked, with 'Set IP address in the project' selected, resulting in IP address 192.168.100.2 and Subnet mask 255.255.255.0. The 'PROFINET' section shows 'Generate PROFINET device name automatically' checked, resulting in device name compax3-000 and device number 1.



pic 1: SIMATIC - HW Config

Edit the Start address of PKW (here 270) in Instance Variable <nLaddr> (DB45.DBW6).

3.3. Application interface of "C3ControlManager"

3.3.1. Schematic drawing for in- and output

Areas of FB45 / DB45

in input	out output	In_out In- and Output	stat Static Memory
-----------------	-------------------	------------------------------	---------------------------

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

3.3.2. Declaration of In- and Output

Parameter	Declaration	Data type	description
bAbsoluteRelative	IN	BOOL	=0 absolute, =1 relative movement
bEnable	IN	BOOL	=1 energise =0 disenergise with AUS3 - Ramp (Not-Stop)
bExDataTransfer	IN	BOOL	=0 internal DP interface with SFC14/15 (internal Master in S7 CPU) =1 external DP interface with FC2/FC1 (external Master CP 342-5)
bFaultReset	IN	BOOL	Acknowledge with rising edge, after that it is necessary to activate energise <bEnable> (caused by the edge detection it is needed to be set to 0 first)
bHold	IN	BOOL	=1 Temporary stop (the movement function is still available), =0 continue
bJogN	IN	BOOL	manual negative: JOG – movement within positioning end limits as long as true
bJogP	IN	BOOL	manual positive: JOG – movement within positioning end limits as long as true
bPositionResetMode	IN	BOOL	=0 Normal-, =1 Reset mode selected (in C3Mgr \ I20T11 drive Configuration \ units... positioning reset distance and positioning reset distance denominator is different from 0)
bStop	IN	BOOL	=1 Stop (movement function cancelled)
bChangeSetImmediate	IN_OUT	BOOL	With the Rising edge, a new position profile is activated; <bChangeSetImmediate> is rested from the block itself. This command is acknowledged from the block with <bPosRunning>. A new command is

Parameter	Declaration	Data type	description
bStartHoming	IN_OUT	BOOL	also accepted if the actual movement is not finished (<bInPosition> =1). rising edge starts referencing movement, if permitted, neg. edge stops referencing movement, bStartHoming may only be reseted with <bHomingAttained>.
bStartPositioning	IN_OUT	BOOL	With the Rising edge a new position profile is activated, <bChangeSetImmediate> is reseted from the block itself. This command is acknowledged from with <bPosRunning>. A new command is not accepted if the actual movement is not finished (<bInPosition> =1).
bCommErr	OUT	BOOL	=1 Communication failure wit C3 (Failure from SFC14 / SFC15) (all other messages are invalid)
bDriveErr	OUT	BOOL	=1 failure from C3 (device / Motor)
bEnabled	OUT	BOOL	=1 Axis energised =0 Axis nit energised
bHomingAttained	OUT	BOOL	=1 Reference ok.
bHomingErr	OUT	BOOL	=1 watchdog/ timeout for Reference -run (occasionally TonTimer2 correct)
bHomingRunning	OUT	BOOL	=1 Reference run active
bInPosition	OUT	BOOL	=1 Axis in target position
bPosErr	OUT	BOOL	=1 watchdog timeout for Position -order (occasionally. TonTimer1 correct)
bPosRunning	OUT	BOOL	=1 position or active
bTransErr	OUT	BOOL	=1 Format-, commando failure at transfer from / to C3
iAcceleration	STATIC	DINT	acceleration in U32 -Format (integer)
iDeceleration	STATIC	DINT	deceleration in U32 -Format (integer) <iDeceleration> is transmitted throw the PKW channel in the case of a change.
iInPosWindowAbs	STATIC	DINT	Position window in C4_3 Format (3 decimal places in two word integer), additional monitoring of <bInPosition> with absolute positioning. the actual position value is compared with the position set value. That is only possible with absolute positioning (<bAbsoluteRelative> = false) and position Reset mode (<bPositionResetMode> = false) is not activated (e.g. default value = 1000 equal 1 u; u = dimension at C3 Mgr Increments, mm, Degrees, Inch).
iParameterValue	STATIC	DINT	PROFIdrive Parameter Transfer value (source and destination) - write-/read value of the Parameters, with INT-/WORD -Format only one word used.
iPosition	STATIC	DINT	target position / Distance in C4_3-Format (3 decimal places)
iVelocity	STATIC	DINT	speed in C4_3-Format (3 decimal places)
stRd.iPositionValue	STATIC	DINT	actual position in C4_3-Format (3 decimal places)
stRd.iVelocityValue	STATIC	DINT	actual speed in C4_3-Format (3 decimal places)
nCmd	STATIC	INT	ProfiDrive-Parameter Transfer: command: 1 read 2 write WORD 3 write DWORD
nPNUIndex	STATIC	INT	ProfiDrive-Parameter Transfer: PNU-Index
nPNUSubindex	STATIC	INT	ProfiDrive-Parameter Transfer: PNU - Sub index (incremented by one for DPV0 done)
TonTimer1	STATIC	TIME	time value for timeout of Positioning
TonTimer2	STATIC	TIME	time value for timeout Reference run
nLaddr	STATIC	WORD	= 0100 _{hex} (=256 _{dez}) Start address C3-Slave out of SIMATIC HW Config, if <bExDataTransfer> = true here it is required: 0000hex (=0dez).
stRd.nActualError	STATIC	WORD	Actual Failure (see C3 - Manual)) in WORD - Format =1 no Error!
stC3PKWInDint.nPKE	STATIC	Word 4	Local PKW - input area for external CP
stC3PKWOutDint.nPKE	STATIC	Word 4	Local PKW - output area for external CP
stC3PZDIn.nStatus	STATIC	Word 7	Local PZD - input area for external CP
stC3PZDOut.nControl	STATIC	Word 7	Local PZD - output area for external CP

3.3.3. sequence of process data

3.3.3.1. settings of the static Operands at the Block

1. <nLaddr>

- Parameter from *HW Config*: the first Start address of the C3 -Slave at the PROFIBUS.
- The Block calculates itself the areas for in- and output, with the Lengths of PPO13.
- the Addresses for in- and output need to have equal values. There must not be holes between PZD and PKW.
- There must to be first PKW- and second PZD.

- Example

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

- forward here the value „256“ to *<nLaddr>*.

2. *<bPositionResetMode>*
 - needs to set if there's reset a "position reset distance" defined in C3-configuration
3. *<bExDataTransfer>*
 - Low: settings for CPU with integrated PROFIBUS DP Master
 - High: settings for CPU with external PROFIBUS DP Master
4. *<iInPosWindowAbs>*
 - additional control window for the message *<bInPosition>*
5. *<TonTimer1>*
 - Time value for watchdog positioning, if this time is too short there is shown the error message *<bPosError>*.
6. *<TonTimer2>*
 - Time value for watchdog homing, if this time is too short there is shown the error message *<bHomingErr>*.

3.3.3.2. Settings of dynamic Operands at the Block

1. Switch on
 - Set *<bEnable>*: the block notifies *<bEnabled>*
 - Set *<bStartHoming>*, the block notifies *<bHomingRunning>*. If the homing finished, the block notifies *<bHomingAttained>*.
 - Now reset *<bStartHoming>*.
 - At reaching home position, the message *<bInPosition>* is set.
 - With some home modes (e.g. MN-M 35) and high velocity *<bHomingRunning>* is set so short, that is not visible.
 - If there is a motor with absolute position feedback (SinCos©), it is only one time needed to activate homing. *<bHomingAttained>* remains at True even if the drive is switch off and on again. Attention: with activating the homing from C3–Optimisation *<bHomingAttained>* is not set.
2. Positioning
 - Set Parameters for Positioning
 - *<bAbsoluteRelative>*
 - *<iPosition>*
 - *<iVelocity>*
 - *<iAcceleration>*
 - *<iDeceleration>*
 - start the positioning with activating : *<bStartPositioning>*
 - *<bStartPositioning>* is reseted by the block itself
 - the block notifies *<bPosRunning>*
 - With reaching the target position, the message *<bInPosition>* is set.
 - A new target position is only possible after *<bInPosition>* was set.
 - For dynamic positioning the input *<bChangeSetImmediate>* is usable
 - It has the same conditions as *<bStartPositioning>* but here is a new command possible during a movement.

3.3.3.3. other Operands at Block

- *<bFaultReset>* acknowledgement of failures of Function block (watchdog) or drive (C3).
- *<bStop>* stops a positioning with rising edge
- *<bHold>* interrupt of Positioning command, as long as Bit it is *true*; Positioning will be finished when the bit is *false* again.
- *<bJogN>* manual mode negative direction, as long as Bit it is *true*.
- *<bJogP>* manual mode positive direction, as long as Bit it is *true*.

3.3.3.4. Messages and display

- `<bCommErr>` communication with C3 not possible
- `<bDriveErr>` C3 is in failure status
- `<stRd.nActualError>` actual Failure number of C3 (see C3 Help)
- `<stRd.iPositionValue>` actual position
- `<stRd.iVelocityValue>` actual velocity

3.3.3.5. read and write Parameters

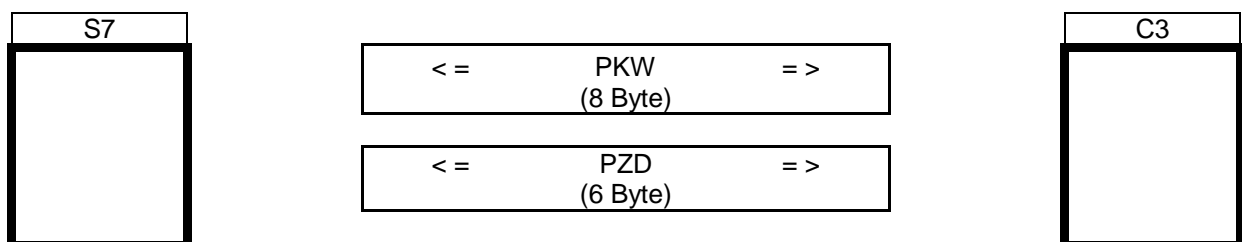
- there are Parameters to be read or changed over bus. These are listed in the table „Objects for the Parameter channel" (see C3 Help). Here is the correlating between Objects (C3) and PNU (PROFIdrive-Profile).
- the Parameters are selected with `<nPNUIIndex>` and `<nPNUSubindex>`
- the value is at `<iParameterValue>`
- `nCmd` is the command for the transfer
 - 1 command for read
 - 2 command for write Word parameter (16 Bit)
 - 3 command for write double word Parameter (32 Bit)
- if you have a Parameter with 16 Bit or 32 Bit is shown in the table „Objects for the Parameter channel": look at column "Bus format".
- `<bTransErr>` shows if there is failure with the data transfer.

3.4. settings for external Master

1. The function block is only possible with S7 PLC with integrated PROFIBUS DP Master: Without the block is not useable.
2. If this interface is used for another purpose and the connection to C3 should be realized with an external communication processor (CP342-5) the function block is possible to be used with following settings:
 - For running the CP 342-5 there are to functions: FC1 / FC2 (DP_SEND / DP_RECV, out of SIMATIC Standard library).
 - Put at DP_SEND the global output t area, and at DP_RECV the global input area.
 - Attention! This global areas include the data's of all bus slaves.
 - The local field of C3 must be transferred to the field at DB45 (e.g. via SFC20 BLKMOV).
 - The local input and output field could be identified with SIMATIC - HW Config.
3. local input area: `<StC3PKWInDint.nPKE>`
4. local output area: `<StC3PKWOutDint.nPKE>`

4. Application example**4.1. Overview of the connection:**

Connection between one plc SIMATIC S7 300 as PROFIBUS DP Master and one drive C3 I20 T11 as PROFIBUS DP Slave.



4.2. cyclic channel (PZD)

The In- and output parameters are selected in the C3 servo manager wizard (folder communication). All tags are either word or double word format. The settings are instructed in the following order to assure the FB is working.

S7	= >	C3	= >	S7
	In		Out	
DBB86	0	STW1	0	DBB72
DBB87	1		1	DBB73
DBB88	2	Position set point	2	DBB74
		value A		
DBB89	3	XSOLL_A	3	DBB75
DBB90	4		4	DBB76
DBB91	5		5	DBB77
DBB92	6	Commanded motion speed	6	DBB78
DBB93	7	NSOLL_D	7	DBB79
DBB94	8		8	DBB80
DBB95	9		9	DBB81
DBB96	10	Set point	10	DBB82
DBB97	11	acceleration B	11	DBB83
DBB98	12		12	DBB84
DBB99	13		13	DBB85

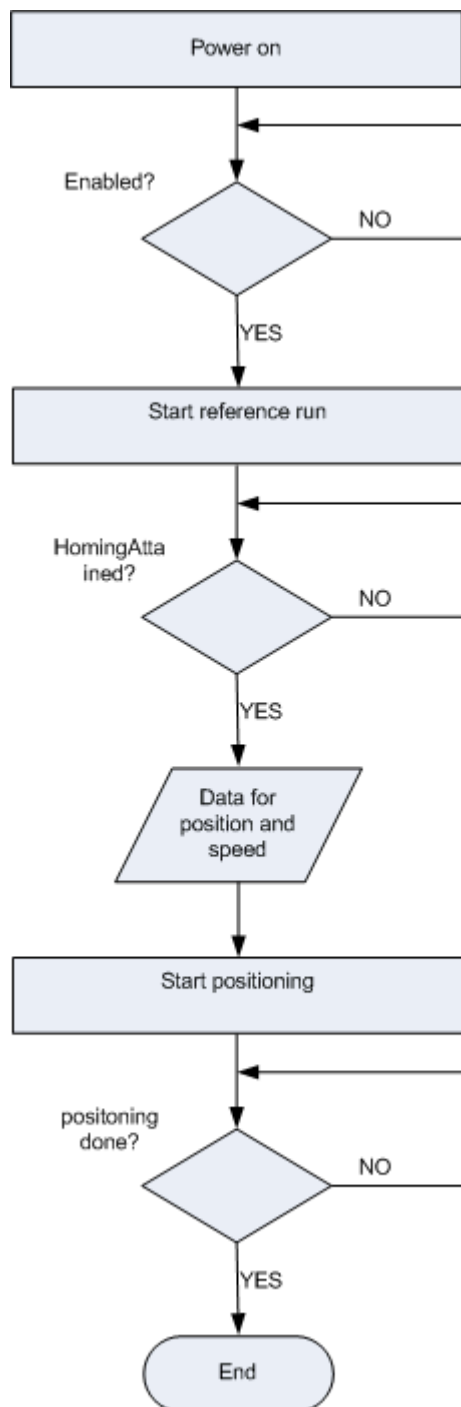
4.3. Acyclic channel (PKW)

Via the 8 Byte PKW interface the user is able to transfer additional parameter to or from C3. The block is using this interface to transfer deceleration and Failure number.

S7	= >	C3	= >	S7
	In		Out	
DBB64	0		0	DBB56
DBB65	1		1	DBB57
DBB66	2		2	DBB58
DBB67	3		3	DBB59
DBB68	4		4	DBB60
DBB69	5		5	DBB61
DBB70	6		6	DBB62
DBB71	7		7	DBB63

4.4. Application Positioning

In the S7 Program the FB45 is call up with all necessary parameters for positioning. The sequence is shown in following picture:



pic 2

The sequence is done in FC400. With setting of

M 0.4 "start"
to TRUE The sequence is started
with

M 0.5 "reset"

The sequence could be stopped.

4.5. Parameter channel

The plc (HMI) is asking for the value of actual torque [683.1]

Also the value of stiffness [2100.2] should be changed. The procedure is explained with help of SIMATIC Variable table.

4.5.1. Used area

		FB45	
DBW28	nCmd	bTransErr	DBX3.1
DBW30	nPNUIIndex		
DBW32	nPNUSubindex		
DBD34	iParameterValue	iParameterValue	DBD34

4.5.2. procedure for reading the value of actual torque [683.1]

1. look for "status of actual current value" in the Table „Object overview sorted by object name“
Fehler! Keine gültige Verknüpfung.

pic 3

2. Open Object description:

Status of actual current value (Object)

Profibus No.	PNU:112	Object No.	683.1
Object name	C3.StatusDevice_ActualCurrent		
Unit of Travel	%	Access:	Read only
Bus format:	E2_6	Valid after:	-
Minimum value	0 n/a	Maximum value	-- n/a
Remark:	Actual current value (actual torque value) The reference is the rated motor current		

pic 4

3. With this table you get the data for following parameters:

nPNUIIndex <112>
nPNUSubindex <0>

4. Edit this two values and "activate modify values" (look in "variable").
5. Next edit the commando for "read" (1) !

nCmd <1>

6. "activate modify values" once more!

Var - VAT 007 PNU Command

Table Edit Insert PLC Variable View Options Window Help

VAT 007 PNU Command -- @ControlManager Applicati\SIMATIC 315-2\CPU 315-2 DP\ControlManager ONLINE

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

pic 5

you see the result at

iParameterValue <1199>

7. the Bus format is **E2_6** (see pic 4) that means we need to divide this value by 64. The result is in per cent rated to nominal value current.

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

8. to verify this result we use the status of the optimization window:

on Cyclic status display: ON

off JOG operation with time monitoring

	Status Display	Value	Units
1			
2			
3			
4	C3.StatusDevice _ActualCurrent [683.1]	15.3023005	%
5			
6			
7	Target RMS motor current (torque-producing	39.9973150	mA
8			
9			
10			

Inputs I7 - I0

Outputs O3 - O0

Analog Input 0/1

0.04 V 0.04 V

Auxiliary outputs M.O 11 - M.O 0

[ms]

pic 6

4.5.3. sequence for changing value of stiffness

1. search for necessary Parameters

Name: Stiffness (control loop dynamics / Stiffness)
C3.ControllerTuning_Stiffness

Object number: 2100.2

Bus format: U16

unit: %

nPNUIIndex: <402>

nPNUSubindex: <2>

2. transfer values via variable table

nPNUIIndex: <402>

nPNUSubindex: <2>

the default value for stiffness is 100 % increased by 10 % you get the value „110“:

iParameterValue: <110>

„activate modify values“ (see “Variable”).

3. Activate the commandos:

The Bus format is „U16“ a Word Format. So the command for write is „2“ (for Double Word Format is it „3“).

nCmd: <1>

„activate modify values“ (see “Variable”).

4. The Object now is transferred, but you need to activate it by *VP validate parameter*.

5. For this reason you need to write the Object " *ValidateParameter* " (VP Object 210.10 C3.ValidateParameter_Global). The sequence is similar like stiffness.

6. Now the values are validated but not flashed. (with next power on the former value are reloaded). To write them to the *Flash* there's the command *Write Flash* (WF Object 20.1 "C3.ObjectDir_Objekts-->FLASH").

caution:

VP and WF use system resources, which decrease the effectiveness of internal communication. This may have the effect that there occur error messages like „cycle time overrun“. So you should use this two commands not too often.

Try to use them during the axis is not energised. It would be better to change first a couple of parameters and activate them together by writing VP only one time.