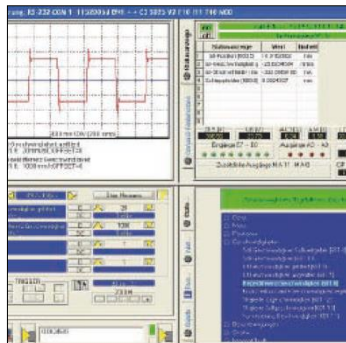




Application Note

Electromechanical Automation Europe
Application Team Offenburg

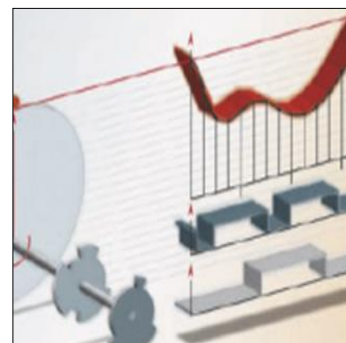


aerospace
climate control
electromechanical
filtration
fluid & gas handling
hydraulics
pneumatics
process control
sealing & shielding



C3 - Communication

ControlManager FB45 for TIA-Portal



Author: K. Woloschin
Application: C3_Comm_AN0012
Version: V1.2
Last change: 17 December 201818



ENGINEERING YOUR SUCCESS.



C3 - Communication

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 supersedes 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 Europe

Business Development & Applications

Copyright © 2010 - 2018

All rights reserved.



C3 - Communication

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	5
3.	SETTING UP	6
3.1.	Compax3 configuration	6
3.2.	SIMATIC - HW Config	7
3.3.	Application interface of "C3ControlManager"	9
3.4.	Settings for external master	Fehler! Textmarke nicht definiert.
4.	APPLICATION EXAMPLE	15
4.1.	Overview of the connection:	15
4.2.	Cyclic channel (PZD)	16
4.3.	Acyclic channel (PKW).....	16
4.4.	Parameter channel	17

C3 - Communication

1. Device supplement

1.1. With the option

- F10 F11 F12
- I20 I32
- T11

1.2. And the master plc

- SIMATIC S7-1500
- SIMATIC S7-1200

2. Purpose of the Block

2.1. Overview

Absolute	Symbol	Comment	Vers	Date	Device	Application
FB45	C3ControlManager For TIA -Portal	C3 I20 I32 T11 ControlManager	0.7	2018-02-09	C3 I20 I32 T11	Positioning, absolute / relative

2.2. Restrictions and application

This block simplifies the control of a C3 T11 (with PROFIdrive profile) with the S7 PLCs. 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.

For Profibus projects with TIA Portal is special gsd- file (PAR_C320_TIA.gsd) and firmware version >= R09-80 are necessary.

Download Compax3 firmware: http://solutions.parker.com/c3_support

C3 - Communication

2.3. History of modifications

V0.1 2004-04-26 09:31:13 PM

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“).

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)

V0.4 2009-06-10 08:45:45 AM

- Delete <bPosRunning> during disabling power stage. Otherwise it would necessary to while until <bPosErr> = True before a new order is accepted!

V0.5 2013-07-29

- Acyclic access to PNUs with Subindex was corrected

V0.6 2013-12-18

- Project migration in TIA Portal, based on former application note A1039

V0.6 2014-07-21

- Extended start address nLaddrPZD and nLaddrPKW instead of nLaddr

V0.7 2018-02-09

- PNU nCMD (1 - read) sequence modified/improved

C3 - Communication

3. Setting up

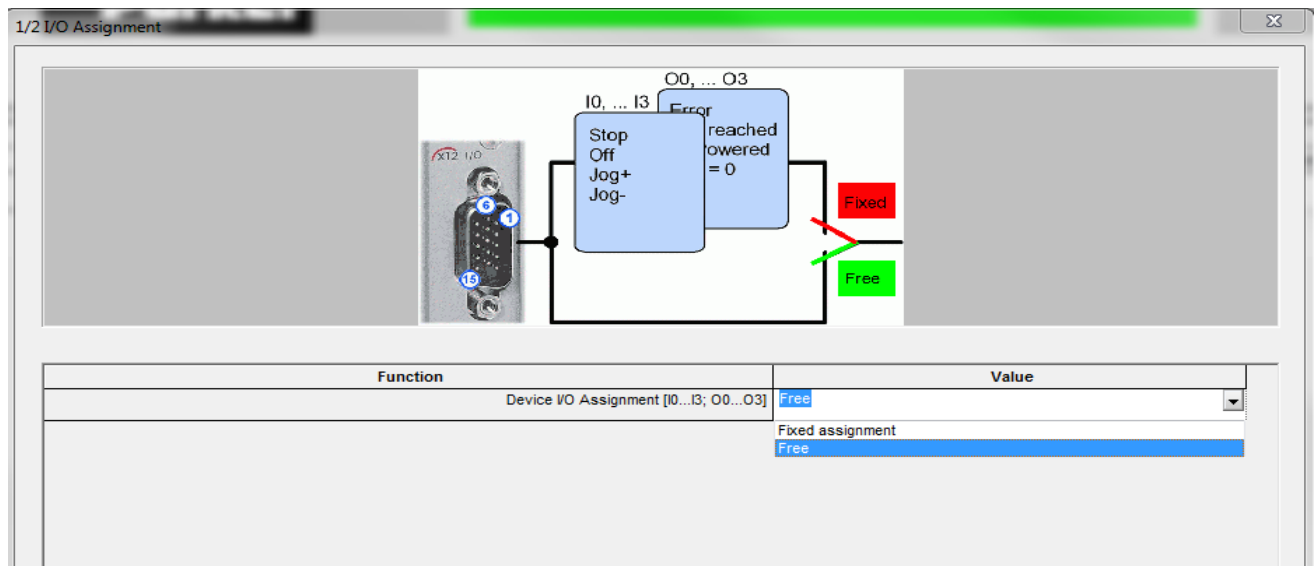
3.1. Compax3 configuration

With C3ServoManager perform 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

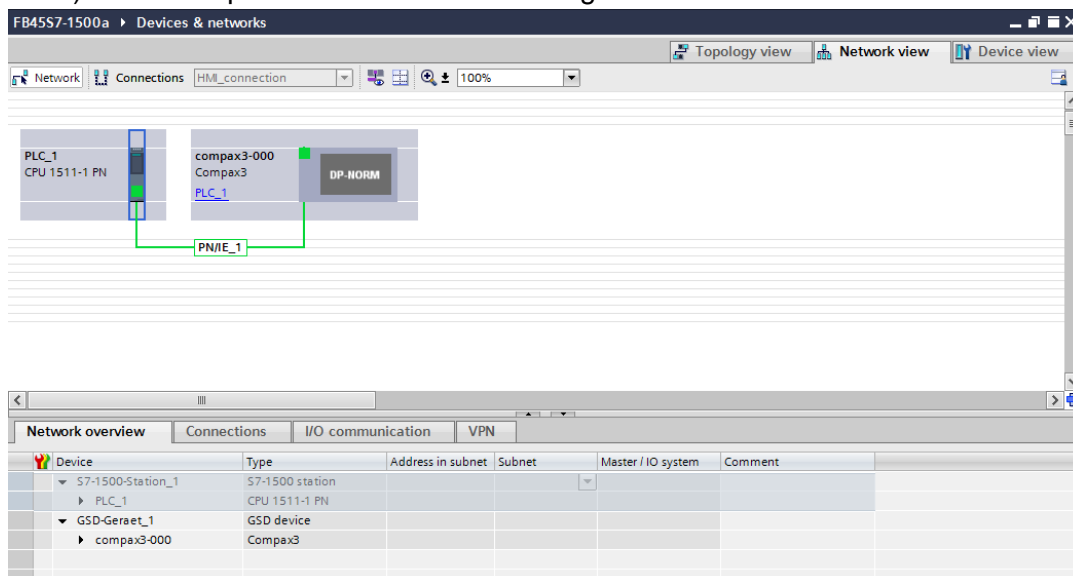
C3 - Communication



It is recommended to select „free“. With „Fixed assignment“ there are 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.

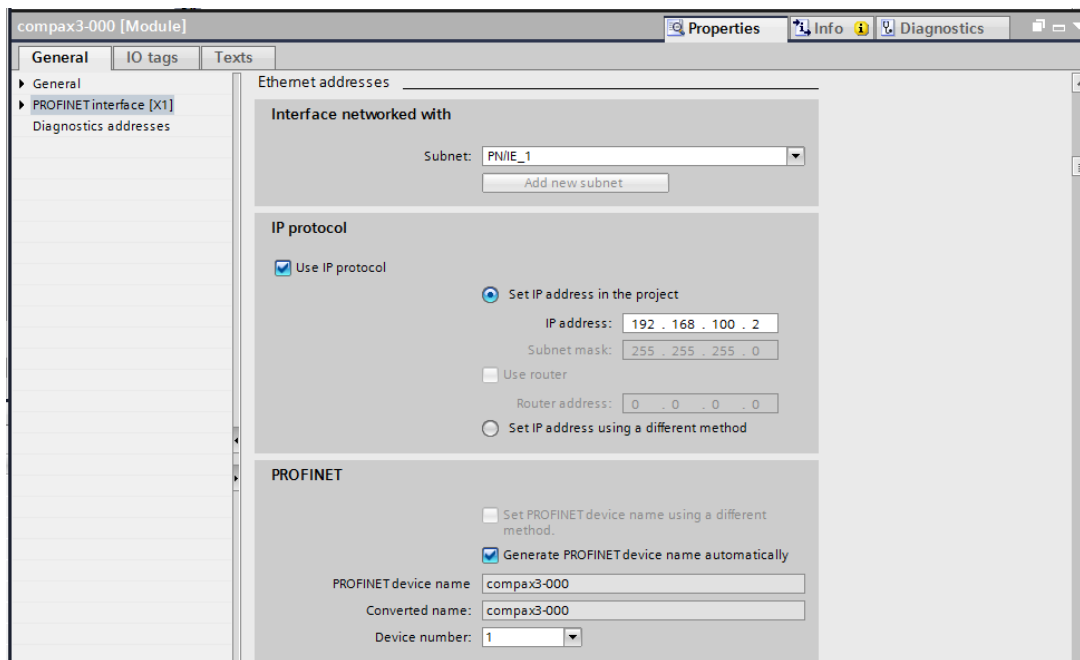
3.2. SIMATIC - HW Config

1) Add Compax3 from Hardware catalog



2) Assign IP-address and device name

C3 - Communication



3) Edit HardwareID of PZD and PKW (here 265 PZD and 267 PKW) in Instance Variable

<nLaddrPZD> (DB45.DBW6).

<nLaddrPKW> (DB45.DBW8).

C3 - Communication

FB45S7-1500a ▶ PLC_1 [CPU 1511-1 PN] ▶ PLC tags					
Tags User constants System constants					
PLC tags					
	Name	Data type	Value	Comment	
25	TPA 23	Pip	23		
26	TPA 24	Pip	24		
27	TPA 25	Pip	25		
28	TPA 26	Pip	26		
29	TPA 27	Pip	27		
30	TPA 28	Pip	28		
31	TPA 29	Pip	29		
32	TPA 30	Pip	30		
33	TPA 31	Pip	31		
34	TPA OB Servo	Pip	32768		
35	PLC_1[MC]	Hw_SubModule	51		
36	PLC_1[Common]	Hw_SubModule	50		
37	PLC_1[Display]	Hw_SubModule	54		
38	PLC_1[Exec]	Hw_SubModule	52		
39	PLC_1	Hw_SubModule	49		
40	PROFINET-Schnittstelle_1	Hw_Interface	64		
41	Port_1[PN]	Hw_Interface	65		
42	Port_2[PN]	Hw_Interface	66		
43	OB_Main	OB_PCYLE	1		
44	PROFINET_IO-System[IOSystem]	Hw_IoSystem	258		
45	compax3-000[Proxy]	Hw_SubModule	268		
46	compax3-000[IODevice]	Hw_Device	264		
47	Compax3	Hw_Interface	259		
48	Port_1[PN](1)	Hw_Interface	260		
49	Port_2[PN](1)	Hw_Interface	261		
50	compax3-000[Head]	Hw_SubModule	263		
51	L_O_7_Worte_1[AI/AO]	Hw_SubModule	265	nLaddrPZD	
52	ParamCh_4_Worte_1[AI/AO]	Hw_SubModule	267	nLaddrPKW	
53	OB_I/O_FLT1	OB_DIAG	82		
54	OB_RACK_FLT	OB_Any	86		
55	OB_COMPLETE RESTART	OB_STARTUP	100		
56	OB_MOD_ERR	OB_Any	122		

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

C3 - Communication

	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	nLaddrPZD	stRd.nActualError	DBW54
DBW8	nLaddrPKW		
DBD10	iPosition		
DBD14	iVelocity		
DBD18	iAcceleration		
DBD22	iDeceleration		
DBD26	iInPosWindowAbs		
DBW30	nCmd	bTransErr	DBX3.1
DBW32	nPNUIindex		
DBW34	nPNUSubindex		
DBD36	iParameterValue	iParameterValue	DBD34
DBD40	TonTimer1		
DBD44	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

C3 - Communication

Parameter	Declaration	Data type	description
bEnable	IN	BOOL	=1 energise =0 disenergize 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 reseted from the block itself. This command is acknowledged from the block with <bPosRunning>. A new command is also accepted if the actual movement is not finished (<bInPosition> =1).
bStartHoming	IN_OUT	BOOL	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 with C3 (Failure from SFC14 / SFC15) (all other messages are invalid)

C3 - Communication

Parameter	Declaration	Data type	description
bDriveErr	OUT	BOOL	=1 failure from C3 (device / Motor)
bEnabled	OUT	BOOL	=1 Axis energised =0 Axis not 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 through 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:

C3 - Communication

Parameter	Declaration	Data type	description
			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
nLaddrPZD	STATIC	WORD	= 0100 _{hex} (=256 _{dez}) Hardware identifier of the DP standard slave / PROFINET IO device to those PIQ range it is to be written.
nLaddrPKW	STATIC	WORD	Separate ID for PKW and PZD channel
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. <nLaddrPZD> and <nLaddrPKW>

Device overview									
	Module	Rack	Slot	I address	Q address	Type	Order no.	Firmware	
	compax3-000	0	0			Compax3	C3***V**F**I32T**M**	V1.51.0	
	Compax3	0	0 X2x			compax3-000			
	ParamCh 4 Worte_1	0	1	270...277	270...277	ParamCh 4 Words			
	I/O 7 Worte_1	0	2	256...269	256...269	I/O 7 Words			

- Parameter from (System constants): HW-ID of the IO-device for PZD and PKW channel.
- Forward the values to <nLaddrPZD> and <nLaddrPKW>.

C3 - Communication

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 PROFINET / PROFIBUS DP Master
 - High: settings for CPU with external PROFINET / 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

C3 - Communication

- 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

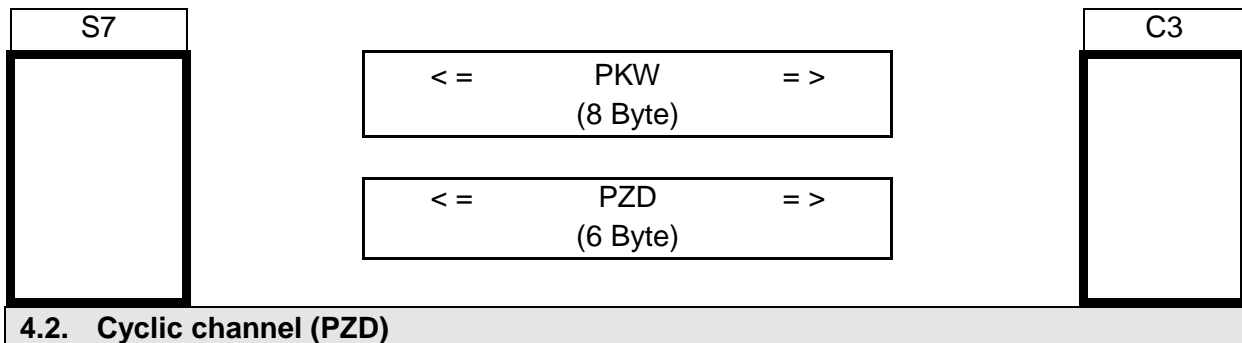
- There are Parameters to be read or changed over bus. These are listed in the table „Objects for the Parameter channel“ (see C3 Help). Here is the correlating between Objects (C3) and PNU (PROFIdrive-Profile).
 - the Parameters are selected with *<nPNUIndex>* and *<nPNUSubindex>*
 - the value is at *<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.

4. Application example

4.1. Overview of the connection:

Connection between one PLC SIMATIC S7 1500 as MASTER / IO-CONTROLLER and one drive C3 I20/I32T11 as SLAVE / IO-DEVICE.

C3 - Communication



4.2. Cyclic channel (PZD)

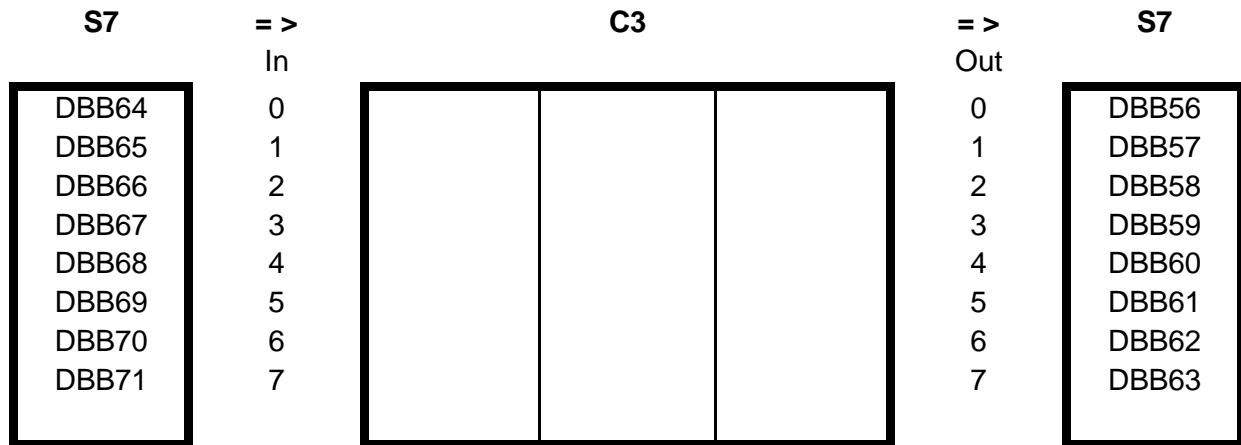
The in- and output parameters are selected in the C3 servo manager wizard (folder communication). All tags are either word or double word format. The settings are instructed in the following order to assure the FB is working.

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

C3 - Communication



4.4. 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.

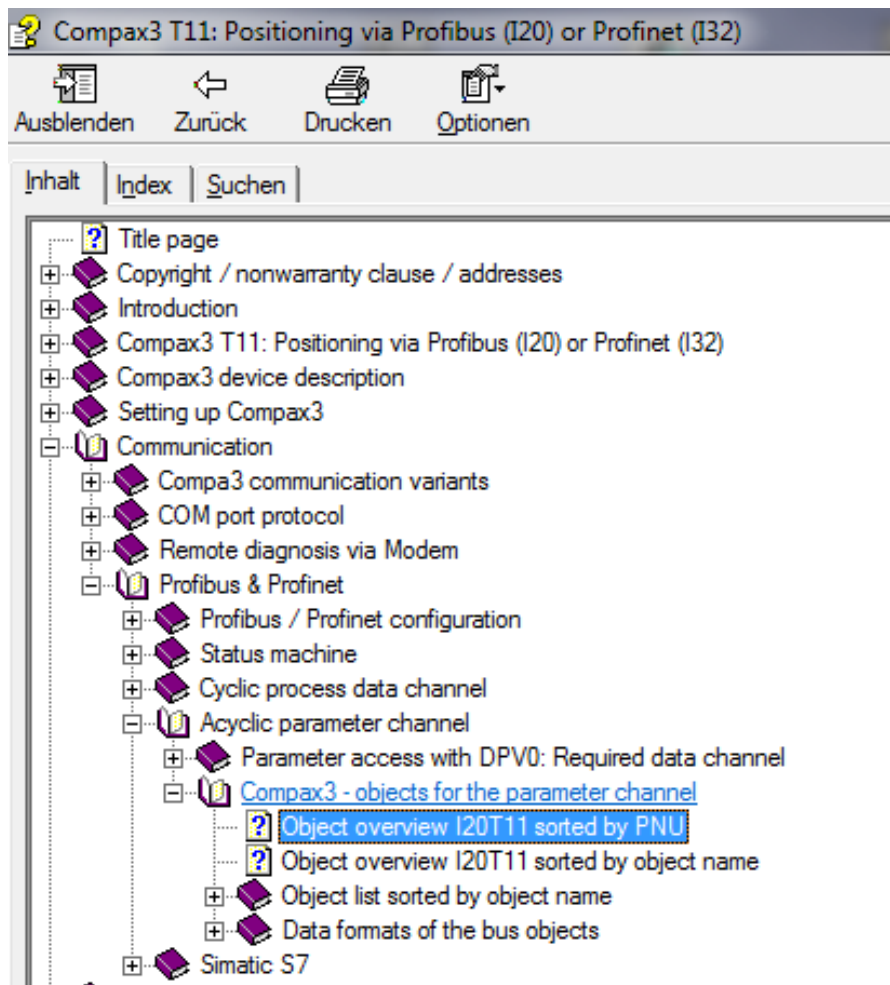
4.4.1. Used area

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

4.4.2. Procedure for reading the value of actual torque [683.1]

1. Look for "status of actual current value" in Compax3 Help at the Table „Object overview sorted by object name“

C3 - Communication



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		

C3 - Communication

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

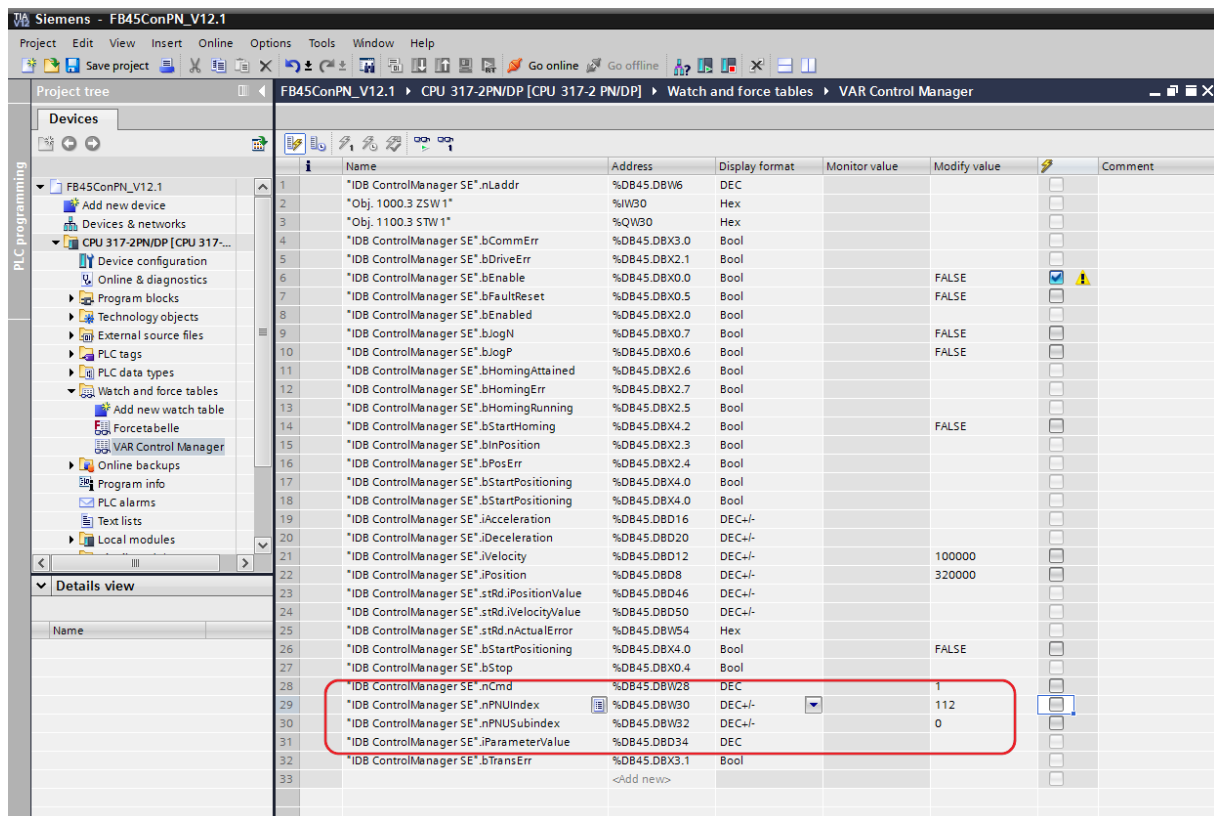
nPNUIndex <112>
nPNUSubindex <0>

4. Edit this two values and “activate modify values” (look in “variable”).

5. Next edit the commando for ”read PNU” (1) !

nCmd <1>

6. “Activate modify values” once more!



Name	Address	Display format	Monitor value	Modify value	Comment
"IDB ControlManager SE".nLaddr	%DB45.DBW6	DEC			
"Obj. 1000.3 ZSW 1"	%IW30	Hex			
"Obj. 1100.3 STW 1"	%QW30	Hex			
"IDB ControlManager SE".bCommErr	%DB45.DBX3.0	Bool			
"IDB ControlManager SE".bDriveErr	%DB45.DBX2.1	Bool			
"IDB ControlManager SE".bEnable	%DB45.DBX0.0	Bool		FALSE	
"IDB ControlManager SE".bFaultReset	%DB45.DBX0.5	Bool		FALSE	
"IDB ControlManager SE".bEnabled	%DB45.DBX2.0	Bool			
"IDB ControlManager SE".bJogN	%DB45.DBX0.7	Bool		FALSE	
"IDB ControlManager SE".bJogP	%DB45.DBX0.6	Bool		FALSE	
"IDB ControlManager SE".bHomingAttained	%DB45.DBX2.6	Bool			
"IDB ControlManager SE".bHomingErr	%DB45.DBX2.7	Bool			
"IDB ControlManager SE".bHomingRunning	%DB45.DBX2.5	Bool			
"IDB ControlManager SE".bStartHoming	%DB45.DBX4.2	Bool		FALSE	
"IDB ControlManager SE".binPosition	%DB45.DBX2.3	Bool			
"IDB ControlManager SE".bPosErr	%DB45.DBX2.4	Bool			
"IDB ControlManager SE".bStartPositioning	%DB45.DBX4.0	Bool			
"IDB ControlManager SE".bStartPositioning	%DB45.DBX4.0	Bool			
"IDB ControlManager SE".iAcceleration	%DB45.DB016	DEC+/-			
"IDB ControlManager SE".iDeceleration	%DB45.DB020	DEC+/-			
"IDB ControlManager SE".iVelocity	%DB45.DB012	DEC+/-	100000		
"IDB ControlManager SE".iPosition	%DB45.DB08	DEC+/-	320000		
"IDB ControlManager SE".stRd.iPositionValue	%DB45.DB046	DEC+/-			
"IDB ControlManager SE".stRd.iVelocityValue	%DB45.DB050	DEC+/-			
"IDB ControlManager SE".stRd.nActualError	%DB45.DBW54	Hex			
"IDB ControlManager SE".bStartPositioning	%DB45.DBX4.0	Bool		FALSE	
"IDB ControlManager SE".bStop	%DB45.DBX0.4	Bool			
"IDB ControlManager SE".nCmd	%DB45.DBW28	DEC		1	
"IDB ControlManager SE".nPNUIndex	%DB45.DBW30	DEC+/-		112	
"IDB ControlManager SE".nPNUSubindex	%DB45.DBW32	DEC+/-		0	
"IDB ControlManager SE".iParameterValue	%DB45.DB034	DEC			
"IDB ControlManager SE".bTransErr	%DB45.DBX3.1	Bool			
<Add new>					

C3 - Communication

26	"IDB ControlManager SE".bStartPositioning	%DB45.DBX4.0	Bool		FALSE	<input type="checkbox"/>
27	"IDB ControlManager SE".bStop	%DB45.DBX0.4	Bool			<input type="checkbox"/>
28	"IDB ControlManager SE".nCmd	%DB45.DBW28	DEC		1	<input type="checkbox"/>
29	"IDB ControlManager SE".nPNUIIndex	%DB45.DBW30	DEC+/-		112	<input type="checkbox"/>
30	"IDB ControlManager SE".nPNUSubindex	%DB45.DBW32	DEC+/-		0	<input type="checkbox"/>
31	"IDB ControlManager SE".iParameterValue	%DB45.DB34	DEC			<input type="checkbox"/>
32	"IDB ControlManager SE".bTransErr	%DB45.DBX3.1	Bool			<input type="checkbox"/>
33	<Add new>					<input type="checkbox"/>

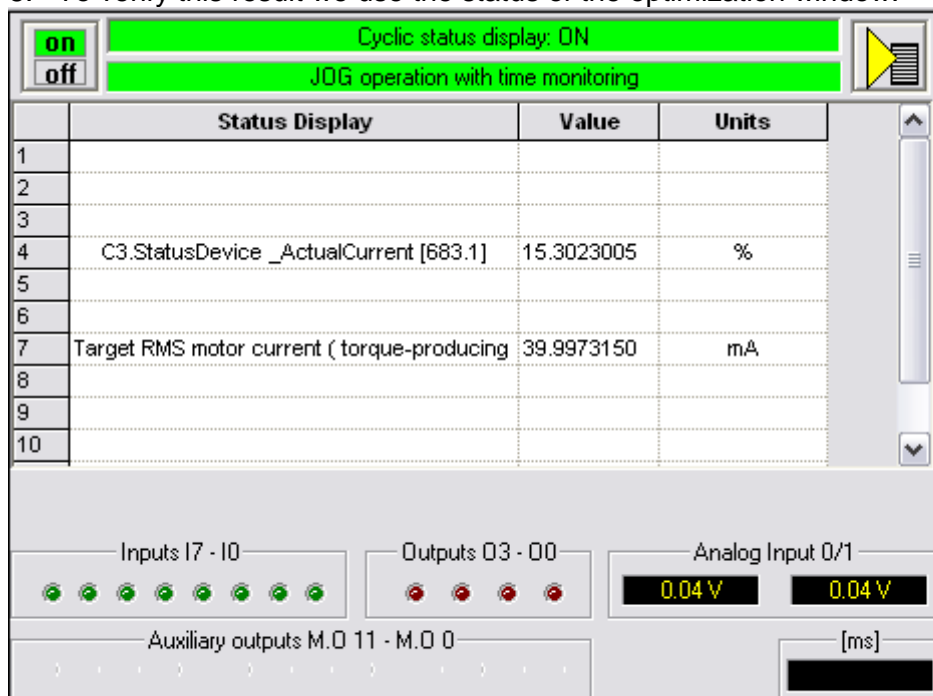
You can see the result at

iParameterValue <1199>

- The Bus format is **E2_6** (see object description) that means we need to divide this value by 64. The result is in per cent rated to nominal value current.

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

- To verify this result we use the status of the optimization window:



The screenshot shows the optimization window with the following status:

- Cyclic status display:** ON
- JOG operation with time monitoring:** ON

Status Display	Value	Units
C3.StatusDevice _ActualCurrent [683.1]	15.3023005	%
Target RMS motor current (torque-producing	39.9973150	mA

Inputs I7 - I0: 7 green LEDs (all on)

Outputs O3 - O0: 3 red LEDs (all off)

Analog Input 0/1: 0.04 V, 0.04 V

Auxiliary outputs M.O 11 - M.O 0: 11 green LEDs (all on)

[ms]: 0.04

4.4.3. Sequence for changing value of stiffness

- Search for necessary Parameters

C3 - Communication

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").

C3 - Communication

**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 these two commands not too often.

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