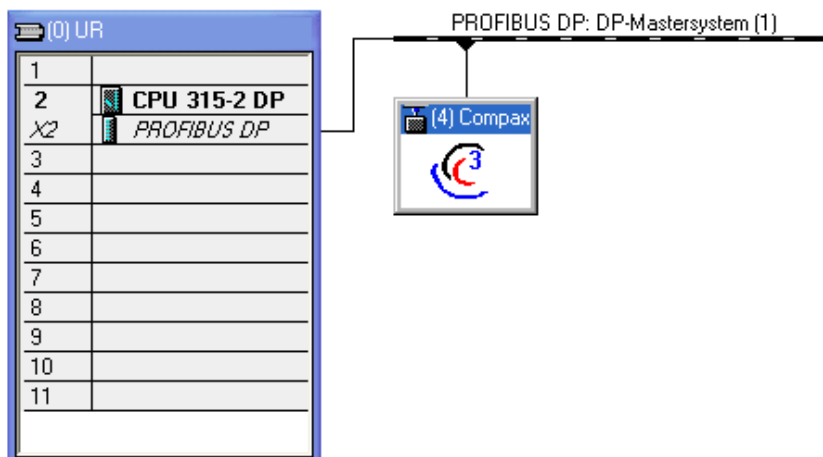


Last update: 11.02.2008 Klaus Zimmer
Application example:

C3 I20 T11 ControlManager



February 08

Copyright © 2008 Parker Hannifin GmbH & Co. KG EME
All Rights reserved.
Windows NT®, Windows 2000™, Windows XP™ are trademarks of Microsoft Corporation.

EME - Electromechanical Automation Europe

Germany: Parker Hannifin GmbH & Co. KG
Electromechanical Automation
Postfach: 77607-1720
Robert-Bosch-Str. 22
D-77656 Offenburg
Tel.: +49 (0)781 509-0
Fax: +49 (0)781 509-98176
E-mail: sales.hauser@parker.com <mailto:sales.hauser@parker.com>
Internet: www.parker-eme.com <http://www.parker-eme.com>

England: Parker Hannifin plc
Electromechanical Automation
Arena Business Centre
Holy Rood Close
Poole, Dorset BH17 7BA UK
Tel.: +44 (0)1202 606300
Fax: +44 (0)1202 606301
E-mail: sales.digiplan@parker.com <mailto:sales.digiplan@parker.com>
Internet: www.parker-eme.com <http://www.parker-eme.com>

Italy: Parker Hannifin S. p. A
Electromechanical Automation
Via Gounod 1
I-20092 Cinisello Balsamo (MI), Italy
Tel.: +39 (0)2660 12459
Fax: +39 (0)2660 12808
E-mail: sales.sbc@parker.com <mailto:sales.sbc@parker.com>
Internet: www.parker-eme.com <http://www.parker-eme.com>

EMN - Electromechanical Automation North America

USA: Parker Hannifin Corporation
Electromechanical Automation
5500 Business Park Drive
Rohnert Park, CA 94928
Phone #: (800) 358-9068
FAX #: (707) 584-3715
E-mail: emn_support@parker.com mailto:emn_support@parker.com
Internet: www.compumotor.com <http://www.compumotor.com>

1	introduction	4
1.1	Product liability	4
1.1.1	<i>Nonwarranty clause</i>	4
1.1.2	<i>Product monitoring liability</i>	4
1.1.3	<i>Right to make changes</i>	4
1.1.4	<i>Warranty Disclaimer</i>	4
1.1.5	<i>Product monitoring liability</i>	4
1.1.6	<i>Limitation of Liability</i>	5
1.1.7	<i>Copyright</i>	5
1.2	Device assignment	5
1.2.1	<i>This manual applies to the following devices:</i>	5
1.2.2	<i>With the supplement:</i>	5
1.2.3	<i>and the Master plc:</i>	6
2	purpose of the Block	6
2.1	overview	6
2.2	restrictions and application	6
2.3	history of modifications	6
3	adjustments	6
3.1	Compax3 configuration	6
3.2	Compax3 Hardware	7
3.3	3.3 SIMATIC - HW Config	7
3.4	Application interface of " C3ControlManager "	8
3.4.1	<i>Schematic drawing for in- and output</i>	8
3.4.2	<i>Declaration of In- and Output</i>	9
3.4.3	<i>sequence of process data</i>	10
3.5	settings for external Master	12
4	Application example	13
4.1	Overview of the connection:	13
4.2	cyclic channel (PZD)	13
4.3	Acyclic channel (PKW)	13
4.4	Application Positioning	14
4.5	Parameter channel	16
4.5.1	<i>Used area</i>	16
4.5.2	<i>procedure for reading the value of actual torque [683.1]</i>	16
4.5.3	<i>sequence for changing value of stiffness</i>	19

1 introduction

1.1 Product liability

1.1.1 Nonwarranty clause

Parker Hannifin - Automation Group - does not give any guarantee that the modules for SIMATIC S7 at hand will function properly under all conditions. From today's point of view there is generally no Software that will work properly under all conditions and requirements. The manufacturer therefore shall not be liable for direct and indirect damages of all kinds caused by the use of the software modules, even if the modules are used in accordance with the description in the manual at hand.

1.1.2 Product monitoring liability

Parker Hannifin - Automation Group - try, within the scope of the product monitoring liability, to identify and describe dangers arising from the use of our software modules. Due to the complexity and our limited insight into the plants of the end customers, where also products of other manufacturers are integrated, we can, however, not identify all possible dangers. Furthermore, not all properties of the software modules are described in this manual.

1.1.3 Right to make changes

Parker Hannifin - Automation Group - claims the right to update the manual and the software modules at any time without advance notice. Software modules may also be blocked without advance notice if dangers are detected, that endanger the proper functioning of the modules. We are not liable to eliminate errors immediately or provide new functions on customer's request.

1.1.4 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.

1.1.5 Product monitoring liability

Parker Hannifin - Automation Group - try, within the scope of the product monitoring liability, to identify and describe dangers arising from the use of our software modules. Due to the complexity and our limited insight into the plants of the end customers, where also products of other manufacturers are integrated, we can, however, not identify all possible dangers. Furthermore, not all properties of the software modules are described in this manual.

1.1.6 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.

1.1.7 Copyright

This manual and the accompanying software modules contain information which is a spiritual property of the author. The authorised user commits to use this information exclusively for the operation with C3 positioning controller and their integration. The duplication and disclosure of the documentation or from extracts needs to be permitted separately. Duplication of the software modules is permitted for the purposes of a data protection.

1.2 Device assignment

1.2.1 This manual applies to the following devices:

- Compax3S025V2 + supplement
- Compax3S063V2 + supplement
- Compax3S100V2 + supplement
- Compax3S025V2 + supplement
- Compax3S063V2 + supplement
- Compax3S100V2 + supplement
- Compax3S150V2 + supplement
- Compax3S015V4 + supplement
- Compax3S038V4 + supplement
- Compax3S075V4 + supplement
- Compax3S150V4 + supplement
- Compax3S300V4 + supplement
- Compax3H050V4 + supplement
- Compax3H090V4 + supplement
- Compax3H125V4 + supplement
- Compax3H155V4 + supplement

1.2.2 With the supplement:

- F10 (Resolver)
- F11 (SinCos®)
- F12 (linear and rotary direct drives)
- I20
- T11

1.2.3 and the Master plc:

- SIMATIC S7-300 or
- SIMATIC S7-400
- with integrated PROFIBUS DP Master (e. g. CPU315-2DP)

2 purpose of the Block

2.1 overview

Absolute	Symbol	Comment	Vers	Date	Device	Application
FB45	C3ControlManager	C3 I20 T11 ControlManager	0.3	2007-06-28 04:22:56 PM	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 energised“).
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 adjustments

3.1 Compax3 configuration

With C3ServoManager few following adjustments:

At folder:

\\ Communication \ PROFIBUS DP - node settings

[PROFIBUS – operation mode]
Positioning

[PLC -> Compax3]
CW 1
Commanded positioning value (XCOMMANDED_A)
Commanded motion speed D (32 Bit)
Commanded acceleration B (32 Bit)

[Compax3 -> PLC]
SW 1
Actual position value XACTUAL_A
Actual speed value NACTUAL_B (32 Bit)

[Operation Mode Settings]
Acyclic process data channel / Parameter channel
Select with "PKW"

Error response on fieldbus failure is up to user.

\\ I20T11 drive configuration \ I/O - assignment
Selection:

- free
- Fixed assignment

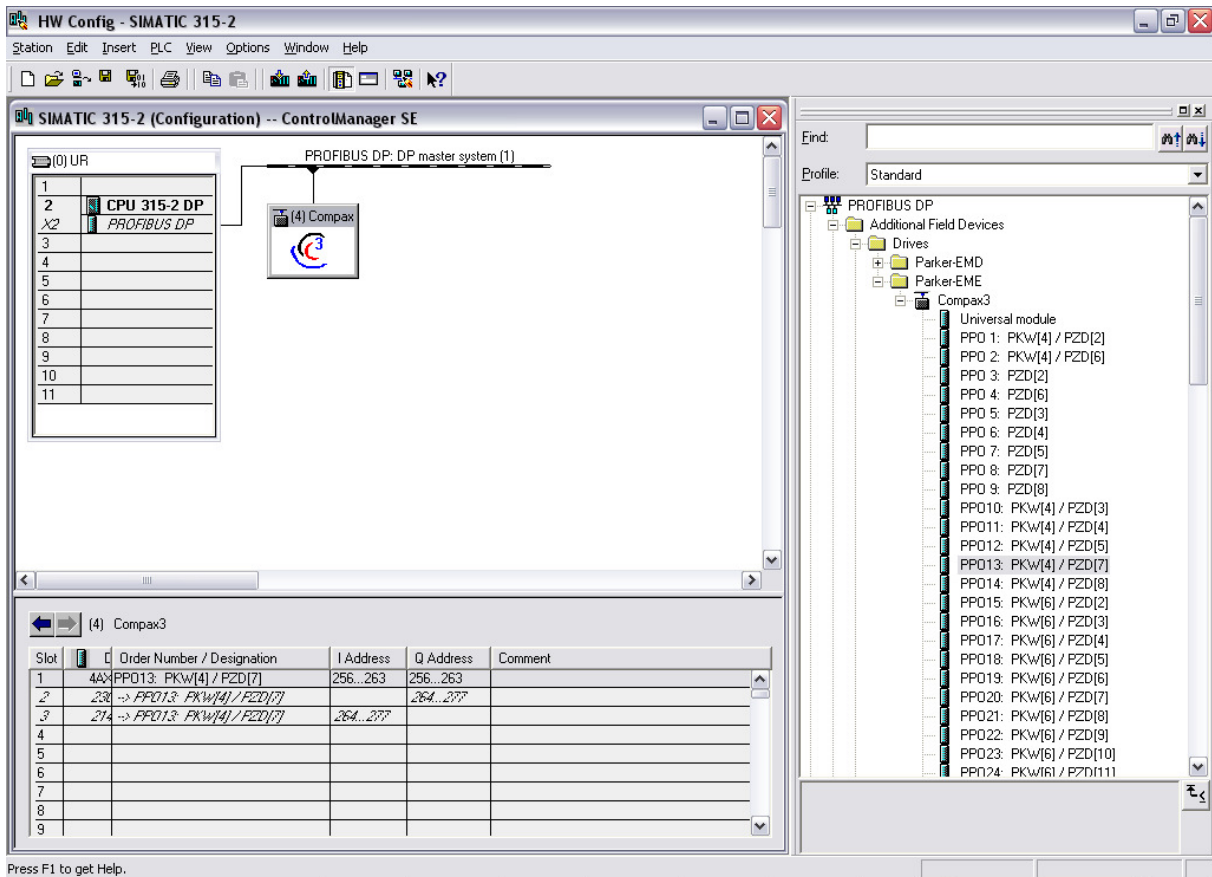
It is recommend to select „free“. With „Fixed assignment“ are used some Binary connections between control word and digital inputs (E0 ... E3), that cases if there is no plug at X12 the C3 is not possible to be controlled.

3.2 Compax3 Hardware

DIP Switch: Bus address
Bus plug: ``ON / OFF`` Bus termination resistance

3.3 SIMATIC - HW Config

Correlated to PPO-Type which is shown in the C3 configuration it is needed to select at SIMATIC - HW Config this type:



pic 1: SIMATIC - HW Config

Edit the Start address of PKW (here 256) in Instance Variable <nLaddr> (DB45.DBW6).

3.4 Application interface of "C3ControlManager"

3.4.1 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

3.4.2 Declaration of In- and Output

Parameter	Declaration	Data type	description
bAbsoluteRelative	IN	BOOL	=0 absolute, =1 relative movement
bEnable	IN	BOOL	=1 energise =0 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 dedection 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 wit C3 (Failure from SFC14 / SFC15) (all other messages are invalid)

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	Istgeschwindigkeit 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 - Subindex (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	Locals PKW - output area for external CP
stC3PZDIn.nStatus	STATIC	Word 7	Localise PZD - input area for external CP
stC3PZDOut.nControl	STATIC	Word 7	Locales PZD - output area for external CP

3.4.3 sequence of process data

3.4.3.1 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

- Here must transferred 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>*.

3.4.3.2 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, die message `<blnPosition>` is set.
- A new target position is only possible after `<blnPosition>` 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.

3.4.3.3 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*.

3.4.3.4 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

3.4.3.5 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 collumn “Bus format“.
- `<bTransErr>` shows if there is failure with the data transfer.

3.5 settings for external Master

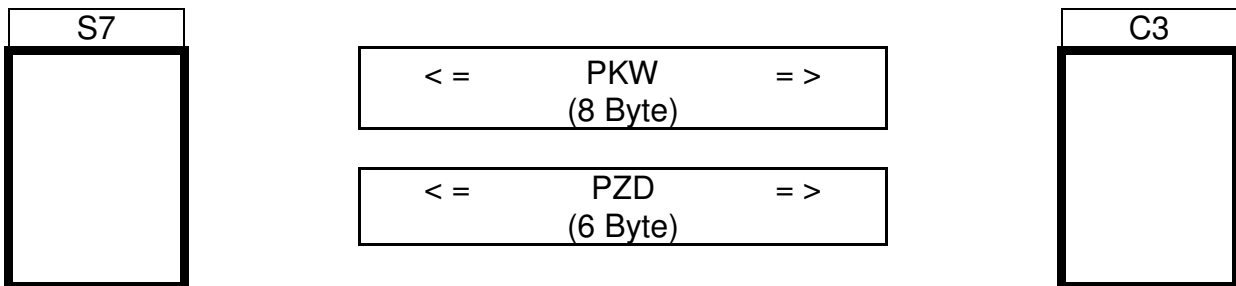
1. The function block is only possible with S7 PLC with integrated PROFIBUS DP Master: Without the block is not useable.
2. If this interface is used for another purpose and the connection to C3 should be realised with an external communication processor (CP342-5) the function block is possible to be 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 included 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)

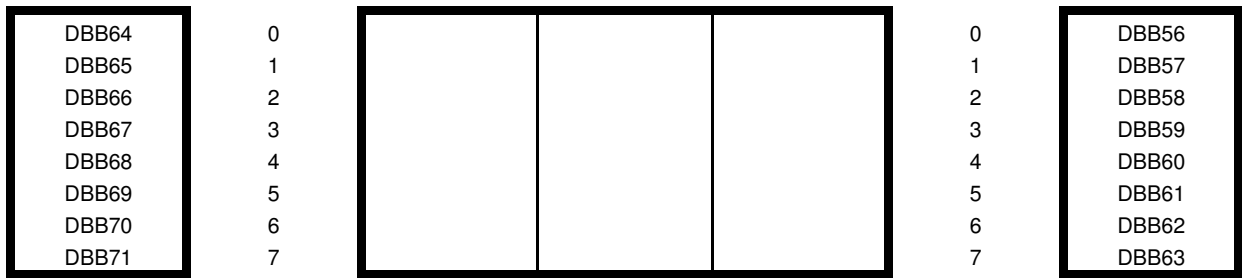
The In- and output parameters are selected in the C3 servo manager wizard (folder communication). All tags are either word or double word format. The settings are instructed in the following order to assure the FB is working.

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

4.3 Acyclic channel (PKW)

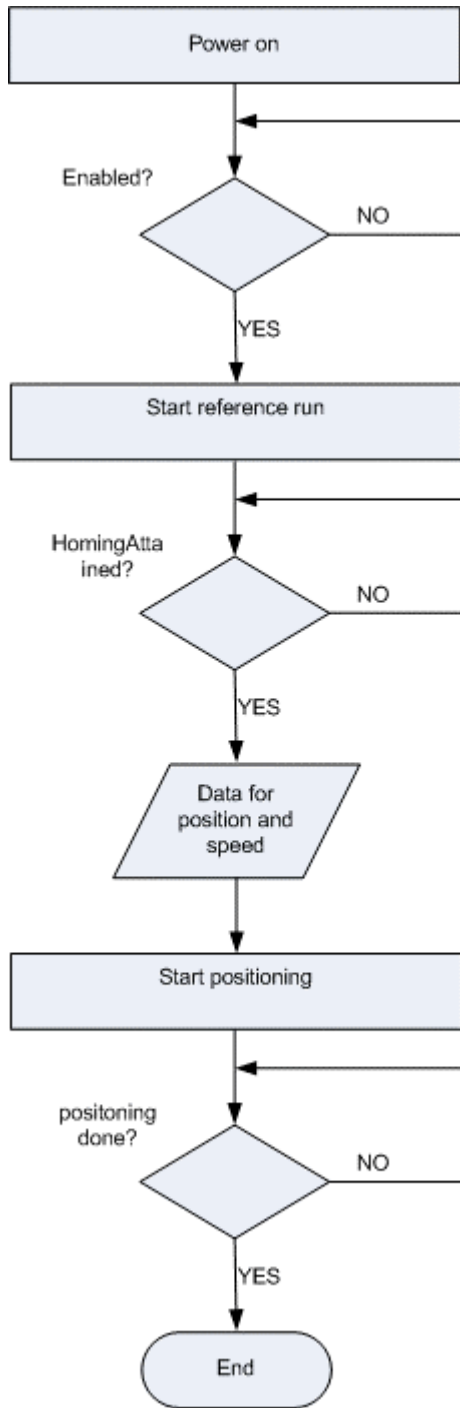
Via the 8 Byte PKW interface the user is able to transfer additional parameter to or from C3. The block is using this interface to transfer deceleration and Failure number.

S7	= >	C3	= >	S7
	In		Out	



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	nPNUIndex		
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“

681.7	Status of the actual speed in the Y2 format	C3.StatusSpeed_Actual_Y2	6	Y2	-
681.8	Status of the actual speed in the Y4 format	C3.StatusSpeed_Actual_Y4	117	Y4	-
681.5	Status actual speed unfiltered	C3.StatusSpeed_Actual	6	C4_3	-
680.5	Status actual position	C3.StatusPosition_Actual	28	C4_3	-
688.2	Status of actual current RMS (torque producing)	C3.StatusCurrent_Actual		E2_6	-
683.3	Status of long-term motor load	C3.StatusDevice_ActualMotorLoad	335	E2_6	-
684.2	Status of motor temperature	C3.StatusTemperature_Motor	336	U16	-
681.6	Status control deviation of speed	C3.StatusSpeed_Error	101	C4_3	-
680.6	Status of following error	C3.StatusPosition_FollowingError	100	C4_3	-
682.4	Status demand acceleration	C3.StatusAccel_DemandValue	325	I32	-
681.10	Status demand speed controller input	C3.StatusSpeed_DemandSpeedController		C4_3	-
681.4	Status demand speed of setpoint generator	C3.StatusSpeed_DemandValue	324	C4_3	-
680.4	Status demand position	C3.StatusPosition_DemandValue	323	C4_3	-
680.12	Status demand position without absolute reference	C3.StatusPosition_DemandController		C4_3	-
688.1	Status of setpoint current RMS (torque forming)	C3.StatusCurrent_Reference		E2_6	-
683.1	Status of actual current value	C3.StatusDevice_ActualCurrent	112	E2_6	-
682.7	Status acceleration feed forward	C3.StatusAccel_FeedForwardAccel		C4_3	-
681.11	Status speed feed forward	C3.StatusSpeed_FeedForwardSpeed		C4_3	-
685.2	Status DC bus voltage	C3.StatusVoltage_BusVoltage	327	E2_6	-
2100.2	Stiffness (speed controller)	C3.ControllerTuning_Stiffness	402.2	U16	VP
1100.4	Control word 2	C3Plus.DeviceControl_Controlword_2	3	V2	Immediate

pic 3

2. Open Object description:

Status of actual current value (Object)

Profibus No.	PNU:112	Object No.	683.1
Object name	<i>C3.StatusDevice_ActualCurrent</i>		
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:

nPNUIndex <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!

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

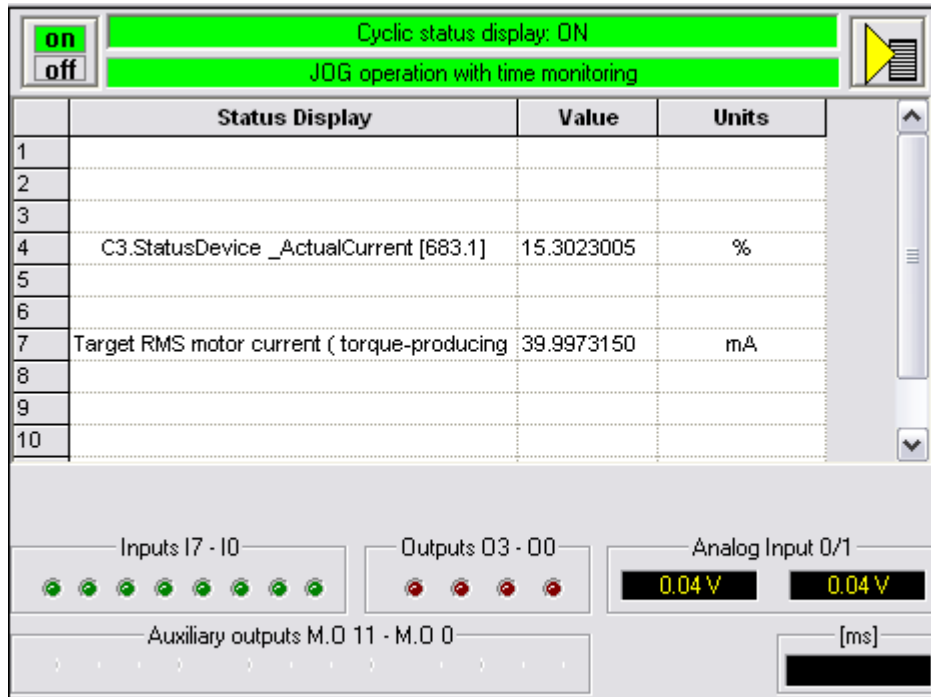
you see the result at

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 optimisation window:



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: %

nPNUIndex: <402>

nPNUSubindex: <2>

2. transfer values via variable table

nPNUIndex: <402>

nPNUSubindex: <2>

the default value for stiffens 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.