

EtherCAT[®]

PROFI[®]
NET

EtherNet/IP[™]

ETHERNET
POWERLINK

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



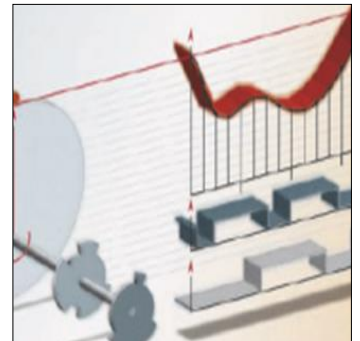
Application Note

Electromechanical Division Europe
Application Team Offenburg



PSD - Communication

ControlManagerAdvanced FB51 for TIA-Portal



Author: K. Woloschin
Application: PSD_Comm_AN0036
Version: V2.0
Last change: 4 May 2023



ENGINEERING YOUR SUCCESS.

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 - 2023

All rights reserved.



Table of Contents

| | | |
|------|---|----|
| 1.1. | History of modifications | 3 |
| 2. | DEVICE SUPPLEMENT | 4 |
| 2.1. | With the option | 4 |
| 2.2. | And the master plc | 4 |
| 3. | PSD_CONTROLMANAGERADV | 4 |
| 3.1. | Inputs and Outputs..... | 4 |
| 3.2. | Description and Advices | 5 |
| 4. | SETTING UP | 5 |
| 4.1. | PSD configuration..... | 5 |
| 4.2. | SIMATIC - HW Config..... | 5 |
| 4.3. | Application interface of "PSDControlManagerAdv" | 8 |
| 5. | APPLICATION EXAMPLE..... | 15 |
| 5.1. | Overview of the connection:..... | 15 |
| 5.2. | Parameter channel | 15 |

1.1. History of modifications

| Date | Change(s) |
|------------|--|
| 17.10.2017 | V1.1 Initial version |
| 25.07.2018 | V1.2 PSD1M2/M3 support. Input iAxisNumber |
| 26.09.2018 | V1.3 Support RegMove/RegSearch function |
| 10.01.2019 | V1.4 blnMode modification |
| 30.02.2019 | V1.5 Velocity mode setpoint change without bStartMode possible, APA response modification |
| 11.08.2020 | v1.6 ResetPositionMode, Simulation Mode activated, gearing ratio mapped by using position and speed as numerator and denominator |
| 21.06.2021 | V1.7 Remaining positioning mode |
| 14.11.2022 | V1.8 iPosStep 2 added |
| 03.05.2023 | V2.0 Extended/buffered APA, nCMD = 4 (read string 5-12byte) |

2. Device supplement

2.1. With the option

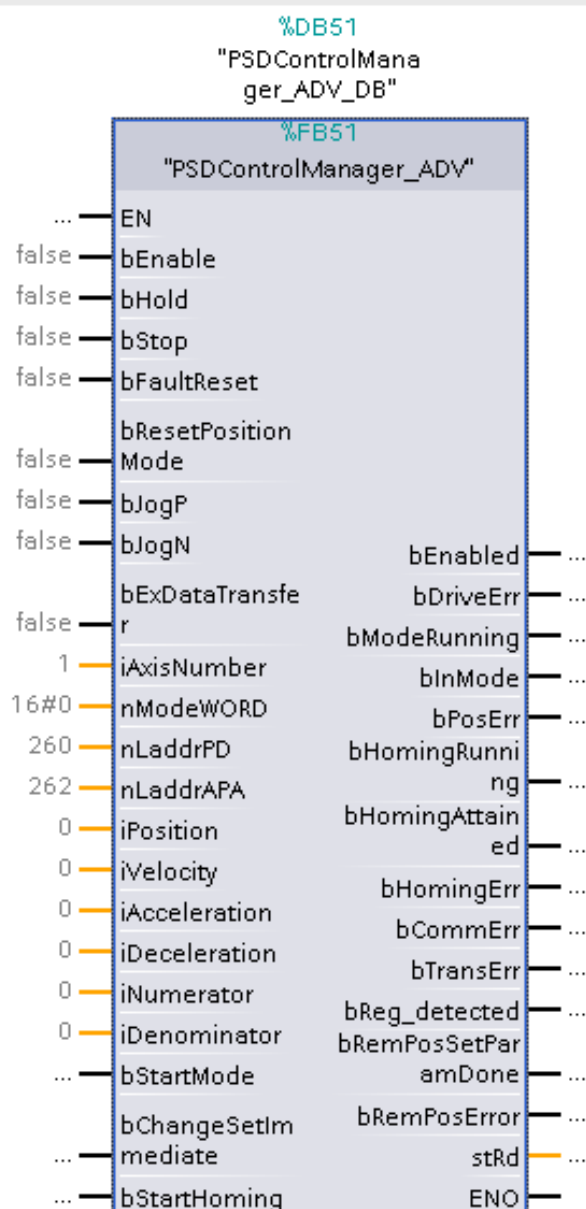
- PSD1S....B2, PSD1M....B2, (PSD1S....C2 and PSD1M....C2 with PSD_ProfiDRIVE_FB)

2.2. And the master plc

- SIMATIC S7-1500
- SIMATIC S7-1200

3. PSD_ControlManagerAdv

3.1. Inputs and Outputs



3.2. Description and Advices

This function block simplifies the control of a PSD drive (with PROFIdrive profile) with the S7. The function block requires the commands and setpoints. The block provides the messages and actual values from the drive. The PD and APA channels are used in both directions.

- The function block is in the RefProject_FB51v1.7_1500_1200 example and must be included in the user project.
- All PLC data types from RefProject_FB51v1.7_1500_1200 are necessary for the FB51.
- For the first test, the control bits and the movement parameters could be manipulated by using the watch table from the RefProject.



This block is only usable with PPO14 (APA_IO = 8BYTE, PD_IO = 18BYTE)

4. Setting up

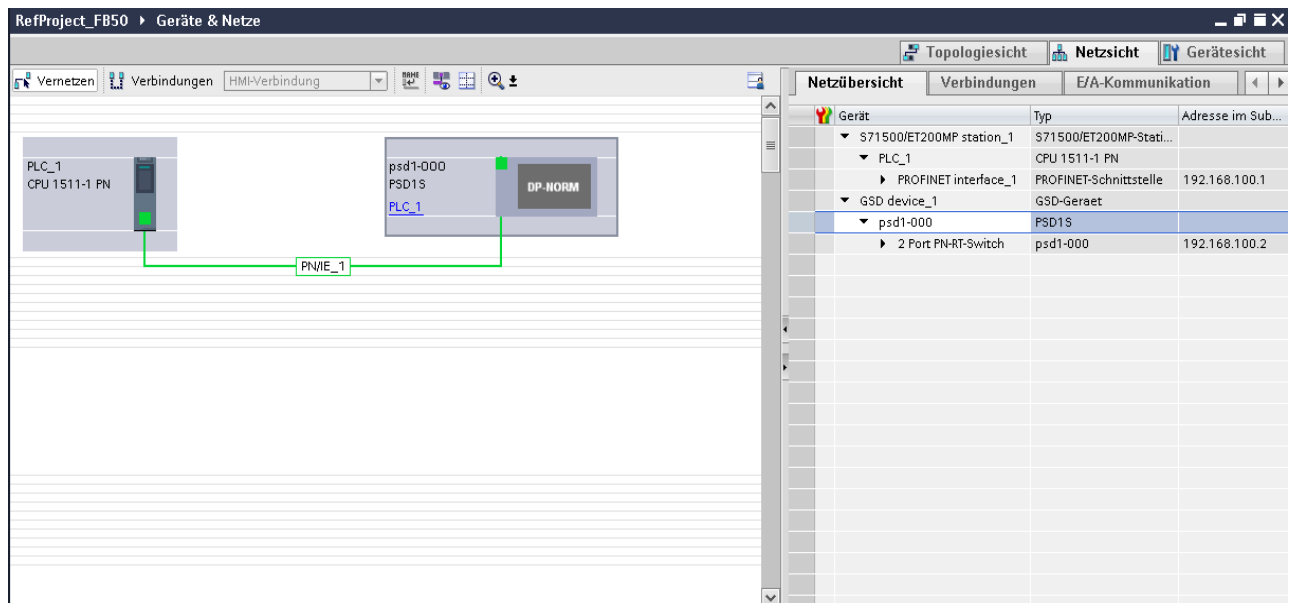
4.1. PSD configuration

Object mapping and fieldbus configuration:

| | |
|---|---|
| Settings | |
| Drive Profile | PROFIdrive |
| Axis1 operation mode | 2 Positioning mode (AC3) with submode MDI (Manual Data Input) |
| Acyclic Parameter Access (APA) | ParameterChannel_4words |
| Error reaction 0x7583: Fieldbus Timeout | Ramp_down_with_quick_down_ramp_Disable_control_loops |
| Edit Mapping | |
| Process Data Size | |
| Axis 1 RxPDO 1 | 9 16 bit words |
| Axis 1 TxPDO 1 | 9 16 bit words |
| Communication Parameters | |
| APA_IO | 4 16 bit words |
| A1_PD_IO | 9 16 bit words |
| Output mapping RxPDO1 (0x1600) | |
| Entry 1 @ ByteOffset 8 | 0x6040.0 Control word 1 (STW1, PNU 967) [16b] |
| Entry 2 @ ByteOffset 10 | 0x607A.0 MDI target position (MDI_TARPOS) [incr] [32b] |
| Entry 3 @ ByteOffset 14 | 0x6081.0 MDI velocity (MDI_VELOCITY) [incr/s] [32b] |
| Entry 4 @ ByteOffset 18 | 0x6083.0 MDI acceleration [incr/si ^{1/2}] [32b] |
| Entry 5 @ ByteOffset 22 | 0x6084.0 MDI deceleration [incr/si ^{1/2}] [32b] |
| Input mapping TxPDO1 (0x1A00) | |
| Entry 1 @ ByteOffset 8 | 0x6041.0 Status word 1 (ZSW1, PNU 968) [16b] |
| Entry 2 @ ByteOffset 10 | 0x6064.0 Position actual value A (XIST_A) [incr] [32b] |
| Entry 3 @ ByteOffset 14 | 0x606C.0 Speed actual value B (NIST_B) [incr/s] [32b] |
| Entry 4 @ ByteOffset 18 | 0x603F.0 Last added (newest) fault code of actual fault situation (PNU 945.x) [16b] |
| Entry 5 @ ByteOffset 20 | 0x3322.2 Active current (ITIST) 16bit [mA] [16b] |
| Entry 6 @ ByteOffset 22 | 0x0007.0 DummyObj_32Bit [32b] |

4.2. SIMATIC - HW Config

Add PSD device from Hardware catalog



Assign IP-address and device name.

Check device module configuration (APA_IO = 4 words, PD_IO = 9 words).

Geräteübersicht

| Baugruppe | Baugr... | Steck... | E-Adres... | A-Adres... | Typ |
|-----------------------|----------|----------|------------|------------|----------|
| ▼ psd1-000 | 0 | 0 | | | PSD1S |
| ► 2 Port PN-RT-Switch | 0 | 0 X6x | | | psd1- |
| APA_IO 4 Words_1 | 0 | 1 | 270...277 | 270...277 | APA_I... |
| PD_IO 9 Words_1 | 0 | 2 | 278...295 | 278...295 | PD_IO... |

psd1-000 [Module]

Eigenschaften

IP-Protokoll

☒ IP-Protokoll verwenden

☒ IP-Adresse im Projekt einstellen

IP-Adresse: 192 . 168 . 100 . 2

Subnetzmaske: 255 . 255 . 255 . 0

☐ Router verwenden

Router-Adresse: 0 . 0 . 0 . 0

☐ Anpassen der IP-Adresse direkt am Gerät erlauben

PROFINET

☐ PROFINET-Geräte name automatisch generieren

PROFINET-Geräte name: psd1-002

Konvertierter Name: psd1-002

Gerätenummer: 1

Check hardware identifier for APA_IO in the device view

Then adjust hardware identifier for APA_IO in <nLaddrAPA> (DB51.DBW8)

RefProject_FB51v1.6_1500_1200_V14 ▶ Ungrouped devices ▶ psd1-Master

Topology view Network view **Device view**

psd1-Master [PSD1S]

psd1-Master

Device overview

| Module | Rack | Slot | I address | Q addr... | Type |
|-----------------------|------|-------|-----------|-----------|----------------|
| psd1-Master | 0 | 0 | | | PSD1S |
| ▶ 2 Port PN-RT-Switch | 0 | 0 X6x | | | psd1-000 |
| APA_IO 4 Words_1 | 0 | 1 | 270...277 | 270...277 | APA_IO 4 Words |
| PD_IO 9 Words_1 | 0 | 2 | 278...295 | 278...295 | PD_IO 9 Words |
| | 0 | 3 | | | |
| | 0 | 4 | | | |
| | 0 | 5 | | | |
| | 0 | 6 | | | |
| | 0 | 7 | | | |
| | 0 | 8 | | | |
| | 0 | 9 | | | |
| | 0 | 10 | | | |
| | 0 | 11 | | | |
| | 0 | 12 | | | |
| | 0 | 13 | | | |
| | 0 | 14 | | | |
| | 0 | 15 | | | |
| | 0 | 16 | | | |
| | 0 | 17 | | | |
| | 0 | 18 | | | |
| | 0 | 19 | | | |

APA_IO 4 Words_1 [Module]

Properties Info Diagnostics

General IO tags System constants Texts

General

Catalog information

Inputs

I/O addresses

Hardware identifier

Hardware identifier

Hardware identifier: 266

Check hardware identifier for PD_IO in the device view
Then adjust hardware identifier for PD_IO in <nLaddrPD> (DB51.DBW6).

RefProject_FB51v1.6_1500_1200_V14 ▶ Ungrouped devices ▶ psd1-Master

Topology view Network view **Device view**

psd1-Master [PSD1S]

psd1-Master

Device overview

| Module | Rack | Slot | I address | Q addr... | Type |
|-----------------------|------|-------|-----------|-----------|----------------|
| psd1-Master | 0 | 0 | | | PSD1S |
| ▶ 2 Port PN-RT-Switch | 0 | 0 X6x | | | psd1-000 |
| APA_IO 4 Words_1 | 0 | 1 | 270...277 | 270...277 | APA_IO 4 Words |
| PD_IO 9 Words_1 | 0 | 2 | 278...295 | 278...295 | PD_IO 9 Words |
| | 0 | 3 | | | |
| | 0 | 4 | | | |
| | 0 | 5 | | | |
| | 0 | 6 | | | |
| | 0 | 7 | | | |
| | 0 | 8 | | | |
| | 0 | 9 | | | |
| | 0 | 10 | | | |
| | 0 | 11 | | | |
| | 0 | 12 | | | |
| | 0 | 13 | | | |
| | 0 | 14 | | | |
| | 0 | 15 | | | |
| | 0 | 16 | | | |
| | 0 | 17 | | | |
| | 0 | 18 | | | |
| | 0 | 19 | | | |

PD_IO 9 Words_1 [Module]

Properties Info Diagnostics

General IO tags System constants Texts

General

Catalog information

Inputs

I/O addresses

Hardware identifier

Hardware identifier

Hardware identifier: 264

4.3. Application interface of "PSDControlManagerAdv"

4.3.1. Schematic drawing for in- and output

Areas of FB51 / DB51

| | | | |
|----------|------------|-----------------------|--------------------|
| in input | out output | in_out In- and Output | stat Static Memory |
|----------|------------|-----------------------|--------------------|

DB51

| | EN | ENO | |
|---------|---------------------|---------------------|---------|
| DBX0.0 | bEnable | bEnabled | DBX34.0 |
| DBX0.1 | bHold | bDriveErr | DBX34.1 |
| DBX0.2 | bStop | bModeRunning | DBX34.2 |
| DBX0.3 | bFaultReset | bInMode | DBX34.3 |
| DBX0.4 | bResetPositionMode | | |
| DBX0.5 | bJogP | bPosErr | DBX34.4 |
| DBX0.6 | bJogN | bHomingRunning | DBX34.5 |
| DBX0.7 | bExDataTransfer | bHomingAttained | DBX34.6 |
| DBW2.0 | iAxisNumber | bReg_detected | DBX35.2 |
| DBW4.0 | nModeWord | bHomingErr | DBX34.7 |
| DBW6.0 | nLaddrPD | bCommErr | DBX35.0 |
| DBW8.0 | nLaddrAPA | bRemPosParamDone | DBX35.3 |
| DBD10.0 | iPosition | bRemPosError | DBX35.4 |
| DBD14.0 | iVelocity | | |
| DBD18.0 | iAcceleration | | |
| DBD22.0 | iDeceleration | | |
| DBD26.0 | iNumerator | | |
| DBD30.0 | iDenominator | | |
| DBX48.0 | bStartMode | | |
| DBX48.1 | bChangeSetImmediate | stRd.iPositionValue | DBD36 |
| DBX48.2 | bStartHoming | stRd.iVelocityValue | DBD40 |
| DBD50 | iInPosWindowAbs | stRd.nActualError | DBW44 |
| DBW54 | nCmd | stRd.ActiveCurrent | DBW46 |
| DBW56 | nObjectIndex | bTransErr | DBX35.1 |
| DBW58 | nObjectSubindex | | |
| DBD60 | iParameterValue | | |
| DBD64 | TonTimer1 | iParameterValue | DBD60 |
| DBD68 | TonTimer2 | | |
| DBD72.0 | stPSD_APA_In_Dint | | |
| DBD80.0 | stPSD_APA_Out_Dint | | |

4.3.2. Declaration of In- and Output

| Parameter | Declaration | Data type | description |
|----------------------------|--------------------|------------------|---|
| nMode | IN | BOOL | =1 absolute, =2 relative, =4 velocity, =8 gearing =16 RegMove, =32 RegSearch, =64 RemPos |
| bEnable | IN | BOOL | =1 enable =0 disable |
| bExDataTransfer | IN | BOOL | =0 internal DP interface with DPWR_DAT/DPRD_DAT (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 <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 |
| bResetPositionMode | IN | BOOL | =0 Normal-, =1 Reset mode selected (for nMode 1, 2 or 64) When activated, every new positioning command will result a jump of the actual and setpoint position to “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 <bModeRunning>. A new command is also accepted if the actual movement is not finished (<bInMode> =1). |
| bStartHoming | IN_OUT | BOOL | Rising edge starts referencing movement, if permitted, neg. edge stops referencing movement, bStartHoming may only be reseted with <bHomingAttained>. |
| bStartMode | IN_OUT | BOOL | With the Rising edge a new profile is activated, <bStartMode> is reseted from the block itself. This command is acknowledged from with <bModeRunning>. A new command is not accepted if the actual movement is not finished (<bInMode> =1). |
| bCommErr | OUT | BOOL | =1 communication failure (Failure from DPWR_DAT / DPRD_DAT) (all other messages are invalid), Wrong hardware identifier selected. |

| Parameter | Declaration | Data type | description |
|--------------------------------------|--------------------|------------------|---|
| bDriveErr | OUT | BOOL | =1 failure from PSD (device / motor) |
| bEnabled | OUT | BOOL | =1 Axis enabled =0 Axis disabled |
| bHomingAttained | OUT | BOOL | =1 reference ok. |
| bHomingErr | OUT | BOOL | =1 watchdog/ timeout for referencing (TonTimer2) |
| bHomingRunning | OUT | BOOL | =1 referencing |
| bInMode | OUT | BOOL | =1 axis in target position, in gear, in velocity |
| bPosErr | OUT | BOOL | =1 watchdog timeout for positioning (TonTimer1) |
| bModeRunning | OUT | BOOL | =1 mode active |
| bReg_detected | OUT | BOOL | =1 Reg detected |
| bRemPosSetParamDone | OUT | BOOL | =1 Parameter setting successful |
| bRemPosError | OUT | BOOL | 1= Reg not detected |
| bTransErr | OUT | BOOL | =1 format-, commando failure at transfer from / to PSD |
| iAcceleration | IN | DINT | acceleration in UINT32 -Format |
| iDeceleration | IN | DINT | deceleration in UINT32 -Format |
| iInPosWindowAbs | STATIC | DINT | Position window in additional monitoring of <bInPosition> with absolute positioning. The actual position value compared with the position set value. That is only possible with absolute positioning (<bAbsoluteRelative> = false) and position Reset mode (<bPositionResetMode> = false) is not activated. |
| iParameterValue | STATIC | DINT | PSD object value (source and destination) - write-/read value of the Parameters |
| iPosition | IN | DINT | target position / distance in increments |
| iVelocity | IN | DINT | speed in increments |
| iNumerator | IN | DINT | Gear numerator |
| iDenominator | IN | DINT | Gear denominator (only positive values) |
| stRd.iPositionValue | OUT | DINT | actual position in increments |
| stRd.iVelocityValue | OUT | DINT | actual speed in increments |
| stRd.iActiveCurrent | OUT | INT | Active current in mA |
| iRemPos_Position | STATIC | DINT | Remaining position (RegMove) |
| iSensorWindowOn | STATIC | DINT | Within the sensor/reg window a reg signal will be detected |
| iSensorWindowOff | STATIC | DINT | Within the sensor/reg window a reg signal will be detected |
| iRemPos_StatusPosition Offset | STATIC | DINT | PositionOffset 0x34C8.00 |
| iRegistrationPosition | STATIC | DINT | Registration Position 0x34C9 |
| nCmd | STATIC | INT | PSD object transfer: command: 1 read 2 write WORD |

| Parameter | Declaration | Data type | description |
|----------------------|-------------|-----------|---|
| | | | 3 write DWORD |
| nObjectIndex | STATIC | INT | PSD object index |
| nObjectSubindex | STATIC | INT | PSD object subindex |
| TonTimer1 | STATIC | TIME | time value for timeout of Positioning |
| TonTimer2 | STATIC | TIME | time value for timeout Reference run |
| nLaddrPD | IN | WORD | = Hardware identifier of the (PD_IO) Process Data module. |
| nLaddrAPA | IN | WORD | = Hardware identifier of the APA_IO (Acyclic Parameter Access) data module. |
| stRd.nActualError | STATIC | WORD | Actual Failure (see PSD - Manual) =1 no Error! |
| stPSDPKWInDint.nPKE | STATIC | Word 4 | Local PKW - input area for external CP |
| stPSDPKWOutDint.nPKE | STATIC | Word 4 | Local PKW - output area for external CP |
| stPSDPZDIn.nStatus | STATIC | Word 9 | Local PZD - input area for external CP |
| stPSDPZDOut.nControl | STATIC | Word 9 | Local PZD - output area for external CP |

4.3.3. Sequence of process data

4.3.3.1. Settings of the static Operands at the Block

1. <nLaddrPD> and <nLaddrAPA>

- Parameter from (System constants): HW-ID of the IO-device for PD and APA channel.
- Forward the values to <nLaddrPD> and <nLaddrAPA>.

2. <bExDataTransfer>

- Low: settings for CPU with integrated Master
- High: settings for CPU with external Master

3. <iInPosWindowAbs>

- additional control window for the status <bInPosition / bInMode>

4. <TonTimer1>

- Time value for watchdog positioning, if this time is too short there is shown the error message <bPosError>.

5. <TonTimer2>

- Time value for watchdog homing, if this time is too short there is shown the error message <bHomingErr>.

4.3.3.2. 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>*.
- Reset *<bStartHoming>*.
- At reaching home position, the status *<bInPosition>* is set.
- With some home modes (e.g. MN-37) or high velocity *<bHomingRunning>* is set so short, that is not visible.
- For a motor with absolute multiturn position feedback (Hiperface DSL®) it is only necessary to activate homing once. *<bHomingAttained>* remains at "True" even if the drive is switched off and on again.
- With bJogP or bJogN an jog movement can be performed

2. Select mode

- Set nMode:
1 = MoveAbsolute
2 = MoveRelative
4 = MoveVelocity
8 = Gearing
16 = RegMove
32 = RegSearch
64 = RemPos

3. Positioning

- Set parameters for positioning
- *<nMode=1 or 2>*
- *<iPosition>*
- *<iVelocity>*
- *<iAcceleration>*
- *<iDeceleration>*
- Start the positioning with activating *<bStartMode>*
- *<bStartMode>* is reset by the block itself
- The block notifies *<bModeRunning>*
- With reaching the target position the status *<bInMode>* is set.
- A new target position is only possible after *<bInMode>* was set.

4. Velocity mode

- Set parameters for velocity mode
- *<nMode=4>*
- *<iVelocity>*
- *<iAcceleration>*
- Start velocity mode with activating *<bStartMode>*
- *<bStartMode>* is reset by the block itself
- The block notifies *<bModeRunning>*
- With reaching the target velocity the status *<bInMode>* is set.

5. Gearing mode (see additional configuration information in the PSD_help)

- Set parameters for gearing mode
- *<nMode=8>*
- *<INumerator>*

- *<iDenominator>*
 - Start gearing mode with activating: *<bStartMode>*
 - *<bStartMode>* is reset by the block itself
 - The block notifies *<bModeRunning>* during acceleration state
 - With reaching the target velocity the status *<bInMode>* is set.
6. RegMove mode (see additional configuration information in the PSD_help)
- Set parameters for RegMove mode
 - *<nMode=16>*
 - *<iPosition>*
 - *<iVelocity>*
 - *<iAcceleration>*
 - *<iDeceleration>*
 - Start the RegMove with activating: *<bStartMode>*
 - *<bStartMode>* is reset by the block itself. The position for RegMove is set in the drive
7. RegSearch mode (see additional configuration information in the PSD_help)
- Set parameters for RegSearch mode (after RegMove was commanded)
 - *<nMode=32>*
 - *<iPosition>*
 - *<iVelocity>*
 - *<iAcceleration>*
 - *<iDeceleration>*
 - Start the positioning with activating: *<bStartMode>*
 - *<bStartMode>* is reset by the block itself
 - The block notifies *<bModeRunning>*
 - When configured “Reg” input is detected the status *<bReg_detected>* is set.
 - With reaching the target position the status *<bInMode>* is set.
8. Remaining Positioning mode (RegMove, RegSearch and configuration in one mode)
- Set parameters for RemPos mode
 - *<nMode=64>*
 - *<iRemPos_Position (Positioning according to reg)>*
 - *<iSensorWindowOn (StopIgnore)>*
 - *<iSensorWindowOff (StartIgnore)>* (if *iSensorWindowOn* & *iSensorWindowOff* are 0, reg/sensor detection is valid at any position.
 - *<iPosition (Positioning for RegSearch)>*
 - *<iVelocity (Velocity for RegMove and RegSearch)>*
 - *<iAcceleration (Acceleration for the RegSearch and RegMove Positioning)>*
 - *<iDeceleration (Deceleration for the RegSearch and RegMove Positioning)>*
 - Start the motion function with activating: *<bStartMode>*
 - *<bStartMode>* is reset by the block itself
 - The block notifies *<bModeRunning>*
 - When “reg/sensor” signal is detected the status *<bReg_detected>* is set and RegMove (*iRemPos_Position*) is executed.
 - With reaching the target position the status *<bInMode>* is set.
 - If no “reg/sensor” signal is detected (within configured window), status *bRemPosError* is displayed.

4.3.3.3. Other Operands at Block

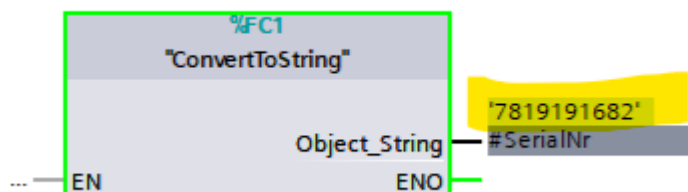
- `<bFaultReset>` acknowledgement of failures of function block (watchdog) or drive.
- `<bStop>` stops a (mode) movement with rising edge
- `<bHold>` interrupt of positioning command, as long as Bit it is *true*; positioning will be finished when the bit is *false* again.
- `<bJogN>` manual mode negative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

4.3.3.4. Messages and display

- `<bCommErr>` communication with PSD not possible
- `<bDriveErr>` PSD is in failure status
- `<bPosErr>` run time error positioning (TonTimer1)
- `<bHomingErr>` run time error homing (TonTimer2)
- `<stRd.nActualError>` actual Failure number of PSD (see PSD Help)
- `<stRd.iPositionValue>` actual position
- `<stRd.iVelocityValue>` actual velocity
- `<stRd.iActiveCurrent>` actual current in mA

4.3.3.5. Read and write drive objects

- the objects are selected with `<nObjectIndex>` and `<nObjectSubindex>`
- the value is at `<iParameterValue>`
- `nCmd` is the command for the transfer
 - 1 command to read (word and double word)
 - 2 command to write word object (16 Bit)
 - 3 command to write double word object (32 Bit)
 - 4 command to read string object (5-12bytes). The requested object data is copied into the ObjectBufferByte, which can then be converted into string format.



Datatype of each object (16 Bit, 32 Bit or String) is shown in the object info "Datatype".

- `<bTransErr>` shows if there is failure with the data transfer.

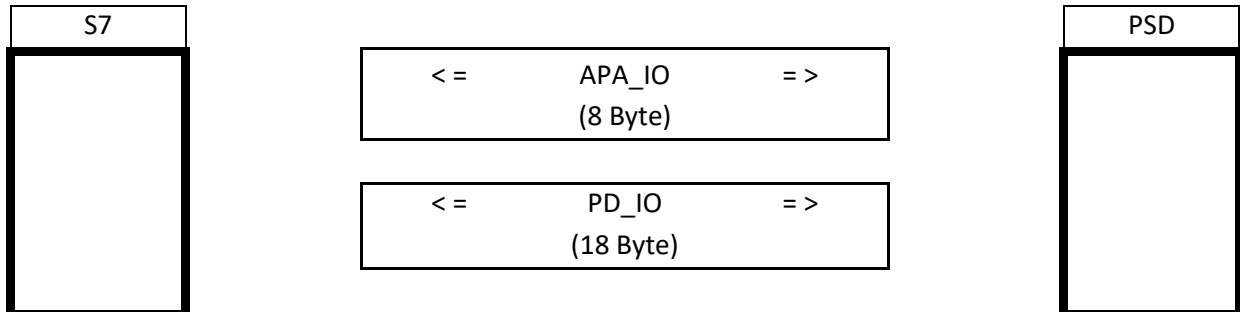


With PSD1M 2 or 3 axes, the object access (APA) may only take place on the first axis!

5. Application example

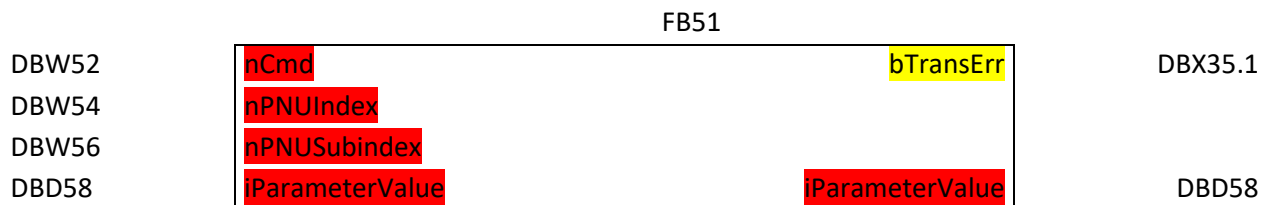
5.1. Overview of the connection:

Connection between one PLC SIMATIC S7 1500/1200 as IO-CONTROLLER and one drive PSD as IO-DEVICE.



5.2. Parameter channel

5.2.1. Used area



5.2.2. Procedure for reading and writing of current limit positive object [0x3212.01]

1. Open Object Editor→object info:

PSD ObjectEditor v1.0.4.25 Objects: 1179

The screenshot shows the PSD ObjectEditor v1.0.4.25 interface. On the left, a tree view shows the configuration structure: Device > Axis 1 > LimitA1 > CurrentLimitAxis1 > PosCurrentAxis1. The right pane shows the configuration for the selected object, **Axis1 - 0x3212.01 PosCurrentAxis1**. The configuration includes a dropdown menu set to '1', a 'User level: Standard' field, and a 'Unit: %' field. A 'Takes effect: Immediately' button is visible. Below the configuration, the 'Object 0x3212.01' description is shown, detailing the object's function and usage.

2. Set requested object index and subindex:
nObjectIndex <3212>
nObjectSubindex <01>
3. Next edit the command for "read" (1)
nCmd <1>
4. Check object value
iObjectValue e.g.<2000>
5. Next set new object value
iObjectValue e.g.<1000>
6. Edit new command for "write 16b" (2)
nCmd <2>
7. Verify the result in PSD ServoMgr (object value 0x3212.01).



Caution:

VP and WF use system resources that reduce the effectiveness of internal communication. This can result in error messages such as "cycle time exceeded". Therefore, do not use these two commands too often.

Try to use them while the axis is not powered. It would be better to change a few parameters first and activate them together by writing VP once.