

Drive Controller SD4x PROFINET IO Connection

A getting started guide for users



D-00000394.3 2024-09-16



Copyright

Original instructions, Copyright © 2024 SIEB & MEYER AG

All rights reserved.

This manual or extracts thereof may only be copied with the explicit authorization by SIEB & MEYER AG.

Trademarks

All product, font and company names mentioned in this manual may be trademarks or registered trademarks of their respective companies.

SIEB & MEYER Worldwide

For questions regarding our products and technical problems please contact us.

SIEB & MEYER AG Auf dem Schmaarkamp 21 21339 Lueneburg Germany

Phone: +49 4131 203 0 Fax: +49 4131 203 2000 info@sieb-meyer.de http://www.sieb-meyer.com

SIEB & MEYER Shenzhen Trading Co. Ltd. Room A208 2/F, Internet Innovation and Creation Services Base Building (2), No.126, Wanxia road, Shekou, Nanshan district, Shenzhen City, 518067 P.R. China

Phone: +86 755 2681 1417 / +86 755 2681 2487 Fax: +86 755 2681 2967 info@sieb-meyer.cn http://www.sieb-meyer.cn

SIEB & MEYER Asia Co. Ltd. 5 Fl, No. 578, Sec. 1 Min-Sheng N. Road Kwei-Shan Hsiang Guishan Dist., Taoyuan City 33393 Taiwan

Phone: +886 3 311 5560 Fax: +886 3 322 1224 info@sieb-meyer.tw

1 1.1	About this Manual. Abbreviations	
2	Parameterization in drivemaster4	6
3 3.1 3.2 3.3 3.4 3.4.1 3.4.2 3.5 3.5.1 3.5.2	Control via PROFINET IO. PROFINET IO Network. Software Connection to the PLC. PROFINET IO Network Management (NMT). Cyclic Process Data (PDO). Setpoint Telegram. Actual Value Telegram. Acyclic Service Data (SDO). Write and Read Operations. Program Example for Siemens PLC.	. 8 8 15 15 15 16 17 18
4	Drive Control	24
4.1	State Machine	25
4.1.1	Commands	
4.1.2	Controlword (0x6040)States	
4.1.2	Statusword (0x6041)	
4.1.3	Option Codes	28
	Abort Connection Option Code (0x6007)	
	Quick Stop Option Code (0x605A) Shutdown Option Code (0x605B)	
	Disable Operation Option Code (0x605C)	29
	Halt Option Code (0x605D) Fault Reaction Option Code (0x605E)	
4.2	Operating Modes	
4.2.1	PV	
4.2.2	Velocity Mode (VL)	
4.2.2.1	Target and Actual Values in Velocity Mode	33
	vl Target Velocity (0x6042)vl Velocity Demand (0x6043)	
	vi Velocity Actual Value (0x6044)	
	vl Velocity Min Max Amount (0x6046)	
	vl Velocity Acceleration (0x6048)	
	vl Velocity Deceleration (0x6049)vl Velocity Quick Stop (0x604A)	
	vl Set-point Factor (0x604B)	
	vI Dimension Factor (0x604C)	37
4.2.3	Torque Mode	
4.2.4	Change Operating Mode	
	Supported Drive Modes (0x6502) Mode of Operation (0x6060)	
	Mode of Operation (0x0000) Mode of Operation Display (0x6061)	
4.3	Target and Actual Values	
4.3.1	General Target Values	
	Target Velocity (0x60FF)	
4.3.2	Target Torque (0x6071) Actual Values	
7.0.2	Position Actual Value (0x6064)	
	Velocity demand value (0x606B)	41
	Velocity Actual Value (0x606C)	
	Torque Demand (0x6074) Torque Actual Value (0x6077)	
	Current Actual Value (0x6078)	
	DC-Link Circuit Voltage (0x6079)	42
4.3.3	Motor Parameters	
	Motor Rated Current (0x6075)	
	Motor Rated Torque (0x6076) Motor Type (0x6402)	
4.3.4	SI Units	



7	Index	57
6	Reference	56
5.2	Diagnosis in PLC	55
5.1.1 5.1.2	Telegrams Diagnostic data	52
5 5.1	Diagnosis in drivemaster4	
_	v	52
4.4 4.4.1	Parameter Set Change via Fieldbus	50
	Digital Inputs (0x60FD) Digital Outputs (0x60FE) Error Code (0x603F)	49
4.3.8	Others	48
	Profile Deceleration (0x6084) Quick Stop Deceleration (0x6085) Torque Slope (0x6087)	. 47 . 48
	Max Acceleration (0x60C5) Max Deceleration (0x60C5) Profile Acceleration (0x6083)	. 47
4.3.7.1	Negative Torque Limit (0x60E1) Acceleration and Deceleration Limits	47
	Max Profile Velocity (0x607F) Max Motor Speed (0x6080) Positive Torque Limit (0x60E0)	46 46 46
4.3.7	Limitations Max Torque (0x6072) Max Current (0x6073)	46 .46
4.3.6	Offsets. Velocity Offset (0x60B1) Torque Offset (0x60B2)	45
4.3.5	Scaling Polarity (0x607E)	
	Position (0x60A8) Velocity (0x60A9) Acceleration (0x60AA) Jerk (0x60AB)	44 44

1 About this Manual

This manual describes how to connect a drive controller of the SD4x device family to a higher-ranking control (controller) via PROFINET IO. For safety instructions and more information on your SD4x drive refer to the according hardware and software documentation.

A basic description of the PROFINET IO network can be found on the website of the PROFIBUS Nutzerorganisation e.V. (PNO) (<u>www.profibus.com/</u>).

The implemented protocol is based on the CANopen profiles. The CANopen profiles describe the data exchange between the bus participants. The CANopen standard specifies the basic communication profile and the device profiles of the connected bus devices.

The PROFINET IO connection as described in this document provides the following communication mechanisms:

- object dictionary
- mapping of process data objects (PDO)
- service data objects (SDO)
- network management (NMT)

In addition, users can make use of the CANopen device profiles, for example the drive profile CiA 402. For this purpose, refer to the standards of the user organization CiA, see <u>section "Other documents", page 56</u>.

1.1 Abbreviations

- CAN <u>C</u>ontroller <u>A</u>rea <u>N</u>etwork
- CiA <u>CAN</u> in <u>A</u>utomation
- **GSDML** general station description XML file
 - NMT <u>n</u>etwork <u>m</u>anagemen<u>t</u>
 - PDO process data object
 - PLC programmable logic control
 - PNS PROFINET station
 - PV <u>p</u>rofile <u>v</u>elocity mode
 - RO <u>r</u>ead <u>only</u> (object access)
 - RW <u>read</u> write (object access)
 - Rx receive
 - SDO <u>Service</u> data object
 - Tx transmit
 - VL <u>v</u>elocity mode



2 Parameterization in drivemaster4

Before you can operate the SD4x drive in the PROFINET IO network, you must set the PROFINET IO communication via the configuration software *drivemaster4*.

Drive Control

- 1. Open the *drivemaster4* software and select the page "Drive control" in the device tree under "Parameters".
- 2. Select the control channel "PROFINET IO". The reference value channel (main ref) is automatically set the same.

Drive control		
Drive function		
Drive function	(00) V/f control	\sim
Control		
Control channel	(14) PROFINET IO	\sim
Setpoint channel main ref	(14) PROFINET IO	\sim
Setpoint channel aux ref	(00) Disabled	\sim
Operating mode		
Mode of operation	(03) Velocity mode PV	\sim

Fig. 1: PROFINET IO parameterization in drivemaster4

The set mode of operation "(03) Velocity mode PV" corresponds to profile velocity mode. For velocity mode you must set the mode of operation to "(02) Velocity mode VL".

Fieldbus

- 3. Open the parameter page "Fieldbus" in the device tree of the *drivemaster4* software.
- 4. Open the tab "PDO mapping" und create the mapping according to the selected operating mode.
 - \rightarrow The following figure shows the mapping for the profile velocity mode:



00	mapping	CANopen E	therCAT	POWERLI	NK Profinet	Modbus	
PD	00 0 ma	pping					
8		Index	Size [Byte]	Offset [Byte]	Name	Data type
_	Object 1	0x6040:00	2	2	0	Controlword	UNIT_U16
2	Object 2	0x60ff:00	4	ļ -	2	Target velocity	UNIT_I32
F	Object 3	0x6072:00	2	2	6	Max torque	UNIT_U16
_					8		
°D	0 0 ma		Size 1	Bytel		Name	Data type
× PD	O O ma	pping Index 0x6041:00	Size [8 Offset [Byte] 0	Name Statusword	Data type UNIT_U16
PD		Index		2	Offset [Byte]		
Ř PD ≞	Object 1	Index 0x6041:00	2	2	Offset [Byte] 0	Statusword	UNIT_U16
- - 2	Object 1 Object 2	Index 0x6041:00 0x606c:00	2	2	Offset [Byte] 0 2	Statusword Velocity actual value	UNIT_U16 UNIT_I32

Fig. 2: Example mapping for profile velocity mode

The following figure shows the mapping for the velocity mode:

D	mapping	CANopen	EtherCAT	POWER	LINK Profinet	Modbus	
PC	00 0 ma	apping					
		Index	Size	[Byte]	Offset [Byte]	Name	Data type
ē	Object 1	0x6040:00)	2	0	Controlword	UNIT_U16
1	Object 2	0x6042:00)	2	2	vl target velocity	UNIT_I16
ŀ	Object 3	0x6072:00)	2	4	Max torque	UNIT_U16
_							
Š					б		
PD	00 0 ma		Size	[Bute]		Name	Data turoe
PD	OO O ma	pping Index 0x6041:00		[Byte] 2	6 Offset [Byte] 0	Name Statusword	Data type UNIT_U16
PD		Index)		Offset [Byte]		
PD	Object 1	Index 0x6041:00)	2	Offset [Byte] 0	Statusword	UNIT_U16
PD	Object 1 Object 2	Index 0x6041:00 0x6044:00)))	2 2	Offset [Byte] 0 2	Statusword vl velocity actual value	UNIT_U16 UNIT_I16

Fig. 3: Example mapping for velocity mode

- 5. Save the parameters in the drive via the button 🔛 in the tool bar and restart the drive.
- ✓ Now, the SD4x drive expects to be supplied cyclically with reference values (PDOs process data objects) via PROFINET IO.



3 Control via PROFINET IO

The following chapters describe the addressing of the drives in the PROFINET IO network and the connection to the PLC using the Siemens software TIA PORTAL.

3.1 **PROFINET IO Network**

The PROFINET IO network is ring-shaped. Hereby, the bus is based on the Ethernet protocol and connected via standard Cat 5 network cables.

Within the PROFINET IO network, addressing the SD4x drives is done implicitly by assigning station names in the PROFINET IO controller.

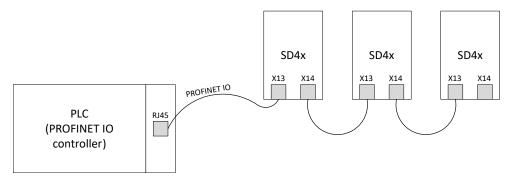


Fig. 4: PROFINET IO network

For information on the connector pin assignment of the PROFINET IO interfaces and the according description of the status LEDs refer to the hardware documentation of the used SD4x drive.

3.2 Software Connection to the PLC

The following example shows the connection to the PLC using the application development system TIA PORTAL by SIEMENS.

1. Copy the GSDML file as well as the BMP file provided by SIEB & MEYER to the project directory

 $\label{eq:constraint} $$ drive: TIA workspace (projectname) Additional Files (GSD) .$

- Example GSDML: gsdml-v2.35-sieb_meyer-0400-sd4x-20230510.xml
- Example BMP: gdsml-0400-0101-sm0sd4x.bmp
- 2. Start the application development system TIA PORTAL and load the project.



3. To install the GSDML file, select the menu "Options" and then the point "Manage general station description files (GSD)".

The following selection window appears.

Mai	nage general station description	ı files			×
Ir	stalled GSDs GSDs in the	project			
So	ource path: C:\TIA\workspace\TIA_	SD4_PNIO\Ad	lditionalFiles\GSD	2) 20	
Co	ontent of imported path				
	File	Version	Language	Status	Info
	gsdml-v2.35-sieb_meyer-0400-sd	V2.35	English, Ger	Already installed	PROFINET I
-	1				
<			1111		>
a a				Delete	Cancel

Fig. 5: Manage GSD files

- a. In the top, select the source path, in which the GSDML files are saved.
- b. Select the required file via the check box on the left side and click "Install".
 - \rightarrow The GSDML file is included in the project, now.
- c. Close the window after successful installation.
 - \rightarrow The hardware catalog is updated now.
- 4. Switch to the network view.
- 5. Select the SD4x drive in the hardware (Other field devices \rightarrow PROFINET IO \rightarrow I/ O \rightarrow SIEB & MEYER AG \rightarrow pns \rightarrow sd4x) und drag it to the network window.

Save project 📑 🐰 🗓 🗎 👔 🗙							~						
t tree	E (TIA_SD4_PNIO ► Der	vices & netw	orks								_ # # ×	Hardware catalog 📑
ices									2 T	opology view	A Network view	Device view	Options
	🔤 🐋	Network L Connect	tions HM cor	nnection 💌 🔛	* 180	⊕ ±						8	
												-	V Catalog
IA_SD4_PNIO	^												<search></search>
Add new device													
Devices & networks		PLC_1				sd4x							Filter Profile: Alb
PLC_1 [CPU 1511C-1 PN]		CPU 1511C-1 PN				sd4x							
Device configuration		10				Not assigned	01						→ 📴 HM
& Online & diagnostics													PC systems
Software units													Drives & starters
Program blocks													Network components
Add new block													Detecting & Monitoring
- Main (081)													Distributed NO
RdData [DB1]													• Power supply and distribution
WrData [DB4]													▼ Tield devices
• System blocks													AS-Interface
 B Program resources 		<								2	100%		Commanding and signaling
RDREC_DB [DB2]							PO#11 21#11						SIPLUS HCS
WRREC_DB [DB3]		Network overview	Connect	ions I/O commun	ication VP	N TeleCont	rol						Other field devices
Technology objects			-	-	Address in subnet			La contra c	Comment				Additional Ethernet devices
External source files		W Device		Туре		Subnet	Master / IO system	Device number	Comment				- PROFINETIO
PLC tags		 \$71500/ET200/ 	MP-Station_1	S71500/ET200MP station									> Drives
PLC data types		▶ PLC_1		CPU 1511C-1 PN									Encoders
Watch and force tables		CP 1542-5_1	1	CP 1542-5					_				Gateway
Gig valch and lorce tables Goline backups		 GSD device_1 		GSD device									- III 10
		▼ sd4x		sd4x									Hilscher Gesellschaft
Garage Traces		▼ PNHO		sd4x	192.168.0.1	Not connected		0					SIEB & MEYER AG
OPC UA communication		Port		Port 1									• m pns
Device proxy data		Port :	2	Port 2									sd4x
Program info													SIEMENS AG
PLC supervisions & alarms													Sensors
PLC alarm text lists		GSD device 1 [Device	1							Properties	Info Dia	anostics 👘 🗆 🗖	
Local modules										Shoperdes	179 mile 179 pres	gnosues	PROFIBUS DP
Jungrouped devices		General 10 tags	s System	n constants Texts									
Security settings		General											
Cross-device functions			(Seneral									
Unassigned devices													
Common data					Name: GSD der	vice 1							
Documentation settings					Author: JApfel	-							
👩 Languages & resources													
Version control interface			1	Co	omment:							~	
Online access													
Card Reader/USB memory	~		6										
tails view												~	
													< =

Fig. 6: Network view



6. Right-click on the text "Not assigned" and select the entry "Assign to new IO controller" from the context menu.

The following window appears:

elect IO controlle		
Name		
PLC_1.PROFINET-	Schnittstelle_1	
	OK	Cancel

Fig. 7: Select IO controller

- a. Select the PROFINET interface in the window and click "OK".
 - \rightarrow The drive is connected with the IO controller now.
- 7. Switch to the device view.

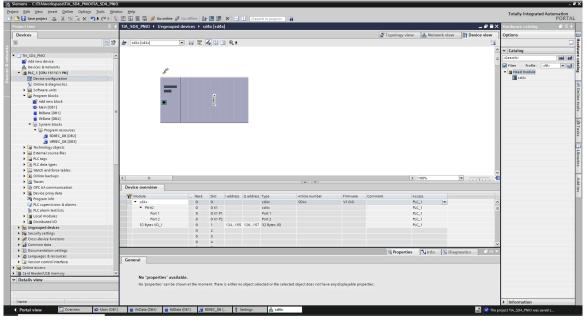


Fig. 8: Device view

- 8. To rename the drive, proceed as follows:
 - a. Click the module name "sd4x" and enter the desired name.

b. Then, assign this device name to the drive via the icon "Name".The window "Assign PROFINET device name" appears.

IP add	PROFINET device name: 5d4x Device type: 5d4x Device type: 5d4x Device type: 5d4x Device filter PGIPC interface: PGIPC interfa	Device type: Colline access Type of the PG/PC interface: PG/PC interface: PG/PC interface: PG/PC interface: Poly Show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: Paddress MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-8A-PD NAC address Device PROFINET device name Status OK 192.168.0.1 00-01-84-01-8A-PD NAC address Device PROFINET device name Status OK MAC address Device PROFINET device name Status OK Status OK Status OK Status OK Status OK	Device type: sdax Online access Type of the PGIPC interface: PGIPC interface: PGIPC interface: Point access Device filter Only show devices of the same type Only show devices with bad parameter settings IP address MAC address Device PROFINET device name Status 192.168.0.1 0001-84-01-AA-FD NetX sd4x V The status information: Search completed.1 of 9 devices were found.			Configured PROF	riner de	vice		
IP add	Online access Type of the PGIPC interface: PGIPC interface: PGIPC interface: Realtek PCIe GbE Family Controller Device filter Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Online access Type of the PG/PC interface: Povice filter Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device IP address Update list	Online access Type of the PGIPC interface: PGIPC interface: PGIPC interface: Period of the PGIPC interface: Point of the PGIPC interface: Point of the PGIPC interface: Point of the PGIPC interface: Provide filter Provide filter Provide filter Provide filter Provide status Provide filter Provide filter Provide status Provide status </th <th></th> <th></th> <th></th> <th></th> <th>sd4x</th> <th></th> <th>-</th>					sd4x		-
IP add	Type of the PG/PC interface: PG/PC interface: PG/PC interface: PG/PC interface: PG/PC interface: PG/PC interface: PG/PC GDE Family Controller POIly show devices of the same type Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Type of the PG/PC interface: Provice filter Poly show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: Paddress MAC address Pevice PROFINET device name Status 192.168.0.1 001-84-01-AAFD NetX sd4x OK It is the intervent	Type of the PCIPC interface: PCIPC interface: PCIPC interface: PCIPC interface: PCIPC interface: Povice filter Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status 192.168.0.1 001-84-01-AAFD NetX sd4x V In status information: Search completed.1 Search completed.1 19 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AAFD".			Devi	ice type:	sd4x		
IP add	PG/PC interface: Realtek PCIe GbE Family Controller Device filter Image: Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	PGIPC interface: Realtek PCIe GbE Family Controller	PGIPC interface: Realtek PCIe GbE Family Controller			Online access				
IP add	Device filter Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Device filter Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: Paddress MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AA-FD Netx sd4x OK 192.168.0.1 00-01-84-01-AA-FD Netx sd4x OK Update list Assign name ine status information: Search completed. 1 of 9 devices were found.				Type of the PG/PC in	nterface:	🖳 PN/IE		-
IP add	Only show devices of the same type Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Fish LED Only show devices of the same type Only show devices with bad parameter settings Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK Assign name Update list Assign name status information: Search completed. 1 of 9 devices were found.	I construction			PG/PC ir	nterface:	Realtek PCIe GbE Fami	ly Controller	- 🖲 🖸
IP add	Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Inly show devices with bad parameter settings Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK Isotropy of the network: It address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK Isotropy of the network: It address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK Isotropy of the network: Isotropy of the network: It address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK Isotropy of the network: Isotropy of the networ	Only show devices with bad parameter settings Only show devices without names Accessible devices in the network: Paddress MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x			Device filter				
IP add	Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Only show devices without names Accessible devices in the network: P address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AA-FD Netx sd4x OK Iso of K Iso of K Update list Assign name update list Assign name Search completed. 1 of 9 devices were found.	Only show devices without names Accessible devices in the network: Paddress MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x			🛃 Only show i	devices of	the same type		
IP add	Only show devices without names Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Only show devices without names Accessible devices in the network: P address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AA-FD Netx sd4x OK Iso of K Iso of K Update list Assign name update list Assign name Search completed. 1 of 9 devices were found.	Only show devices without names Accessible devices in the network: Paddress MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x			Only show (devices wit	th bad parameter settings		
IP add	Accessible devices in the network: IP address MAC address Device PROFINET device name Status	Accessible devices in the network: P address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AA-FD Netx sd4x OK 192.168.0.1 00-01-84-01-AA-FD Netx sd4x OK Update list Assign name update list Assign name search completed. 1 of 9 devices were found.	Accessible devices in the network: IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK IP address MAC address Device PROFINET device name Status IP address MAC address Status Update list Acsign name Update list Acsign name							
IP add	IP address MAC address Device PROFINET device name Status	IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AAFD NetX sd4x OK If lash LED Image: Complete the status information: Image: Complete the status information: Image: Complete the status information:	IP address MAC address Device PROFINET device name Status 192.168.0.1 00-01-84-01-AA-FD NetX sd4x Sd4x OK Image: Status			Onlyshow	devices wi	unouchames		
		192.168.0.1 00-01-84-01-AA-FD NetX sd4x OK Flash LED Image: Status information: Image: Status information: Image: Status information: Search completed. 1 of 9 devices were found. Image: Status information: Image: Status information:	192.168.0.1 00-01-84-01-AA-FD NetX sd4x OK Flash LED Image: Complete the status information: Image: Complete the status information: Image: Complete the status information: Search completed.1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".		Accessible dev					
	192.168.0.1 00-01-84-01-AA-FD NetX sd4x 💟 OK	Flash LED Update list Assign name Image: Search completed. 1 of 9 devices were found. Image: Search completed. 1 of 9 devices were found.	Flash LED Image: Complete List Acsign name Ine status information: Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".		IP addross	MAC address	Device	PROFINET device name	Status	
192.		Update list Assign name update list Search completed. 1 of 9 devices were found.	Update list Update list Assign name Update list Assign name Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".							
		Update list Assign name update list Search completed. 1 of 9 devices were found.	Update list Update list Assign name Update list Assign name Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".			00-01-84-01-AA-FD	NetX	sd4x	🗸 ок	
		Update list Assign name update list Search completed. 1 of 9 devices were found.	Update list Update list Assign name Update list Assign name Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	1		00-01-84-01-AA-FD	NetX	sd4x	🕑 ок	
Flach LED		Update list Assign name Update list Search completed. 1 of 9 devices were found.	Update list Update list Assign name Update list Assign name Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".			00-01-84-01-AA-FD	NetX	sd4x	🕑 ок	
		Update list Assign name ne status information: Search completed. 1 of 9 devices were found.	Update list Assign name ne status information: Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	Elarbied		00-01-84-01-AA-FD	NetX	sd4x	<table-cell> ОК</table-cell>	
		ine status information: Search completed. 1 of 9 devices were found.	ine status information: Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	Flach LED	192.168.0.1	00-01-84-01-AA-FD	NetX		🕑 ОК	
<		Search completed. 1 of 9 devices were found.	Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	Flash LED	192.168.0.1	00-01-84-01-AA-FD	NetX	11		
		Update list Assign name tion: eted, 1 of 9 devices were found.	Update list Assign name tion: eted. 1 of 9 devices were found. device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".			00-01-84-01-AA-FD	NetX	sd4x	<table-cell> ок</table-cell>	
		ne status information: Search completed. 1 of 9 devices were found.	ne status information: Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	flash LED	192.168.0.1	00-01-84-01-AA-FD	NetX		🕑 ок	
<		Search completed. 1 of 9 devices were found.	Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	Flash LED	192.168.0.1	00-01-84-01-AA-FD	NetX	11		Assign name
<		Search completed. 1 of 9 devices were found.	Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	Flash LED	192.168.0.1	00-01-84-01-AA-FD	NetX	11		Assign name
<		Search completed. 1 of 9 devices were found.	Search completed. 1 of 9 devices were found. The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	Flash LED	192.168.0.1	00-01-84-01-AA-FD	NetX	11		Assign name
	Update list Assign na		The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".		192.168.0.1	00-01-84-01-AA-FD	NetX	11		Assign name
ne status information:	Update list Assign na	The PROFINET device name "sd4x" was successfully assigned to MAC address "00-01-84-01-AA-FD".	· · ·	ine status informatio	192.168.0.1		NetX	11		Assign name
ne status information: Search completed. 1 of 9 o	Update list Assign na mation: npleted. 1 of 9 devices were found.		m	ine status informatio Search complete	192.168.0.1	ere found.		m 	pdate list	Assign name
ne status information: Search completed. 1 of 9 o	Update list Assign na mation: npleted. 1 of 9 devices were found.		II	ine status informatio Search complete	192.168.0.1	ere found.		m 	pdate list	Assign name
ne status information: Search completed. 1 of 9 o	Update list Assign na mation: npleted. 1 of 9 devices were found.	m		ine status informatio Search complete	192.168.0.1	ere found.		m 	pdate list	Assign name

Fig. 9: Assign device name

- c. Click the button "Update list".
 - \rightarrow A list with accessible network devices appears.
- d. Select the drive in the list and click the button "Assign name".
- 9. Switch to the topology view.

ct tree	□	& networks								_ # # X	Hardware catalog	POR
rices								Topology view	A Network view	Douise view	Options	_
Aces								G Topology view	Inetwork view	- Inter -	Options	
	🔲 🖻 🐨 🐨 🐨 🔍 Q. ±									1		
										^		
TA_SD4_PNIO	^	_	_							1	<search></search>	644
Add new device	And a second sec			d4x							Filter Profile: All>	
Devices & networks	PLC_1 CPU 1511C-1 PN			14x	1							
PLC_1 [CPU 1511C-1 PN]											Controllers HM	
Device configuration					1						PC systems	
Section 2 Contine & diagnostics											C systems Drives & starters	
Software units											Network components	
 Program blocks 											Detecting & Monitoring	
P Add new block												
41 Main (081)	-										Distributed NO	
RdData [DB1]	-										Power supply and distribute	ution
UrData (DB4)											Field devices Other field devices	
 System blocks 											Uner held devices	
 Program resources 	< .								100%			
RDREC_DB [DB2]	s					1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-		2	100%	💌 🔍		
WRREC_DB (DB3)	Topology overview	opology compa	arison									
Technology objects											1	
External source files	area 27 1											
PLC tags	Provice / port	Slot	Partner station	Partner device	Partner interface	Partner port	Cable data					
 Eg PLC data types 	 \$71500/ET200MP-State 	ion_1										
Watch and force tables	 PLC_1 	1										
Online backups	 PROFINET-Schn 	ttstelle_1 1 X1										
🕨 🎑 Traces	Port_1	1 X1 P	1 GSD device_1	sd4x	PN-IO	Port 1	< 100 m (0.6 µs)					
OPC UA communication	Port 2	1 X1 P	2			Anypartner						
Device proxy data	 GSD device_1 											
Program info	▼ sd4x	0										
PLC supervisions & alarms	· PNHO	0 X1										
PLC alarm text lists	Port 1	O X1 P	1 \$71500/ET200	PLC 1	PROFINET-Schni	Port 1	< 100 m (0.6 µs)					
Local modules	Port 2	0 X1 P	2			Any partner						
Distributed I/O				-							-	
Ungrouped devices	S71500/ET200MP-Station_1							Properties	Linfo Diag	nostics 🛛 🗆 🖵		
Security settings	General IO tags	System constan	ts Texts									
Cross-device functions		0									1	
Common data	General	General										
Documentation settings		-										
👩 Languages & resources												
Version control interface			N	lame: S71500/E	T200MP-Station_1							
Online access			AL	uthor: cnchardw	are							
Card Reader/USB memory	~		Comr	ment:						~		
stails view												
										~		

Fig. 10: Topology view



- 10. Connect the controller ports and the drive port with each other.
 - \rightarrow This gives you additional options for diagnosis.
- 11. Create new PLC data types for the input ad output data with an "Array[0..31] of Byte", e.g. DataInStruct and DataOutStruct.

DataInStruct:

💁 🔚 Save project 🚜 🔏 🗓 🗟 🗙 🎝 🗄 (?										Tasks	DRT/
	L C HA_SD4_PNIO + I	ALC_1 [CPU 1511C-1 PN] > PLC	data types ► D	ataInStruct					_ # = ×		100
Devices										Options	
ĥ	🔤 🖻 🔮 🗣 💺 🖿 🗮										1
	DataInStruct									 Find and replace 	
TA_SD4_PNIO	Name	Data type	Default value	Accessible f	Write N	/isible in	Setpoint	Comment			
Add new device	1 🚭 🕶 ActValue	Array[031] 🔳								Find:	
Devices & networks	2 - ActValue	[0] Byte	16#0	V	V	V					
PLC_1 [CPU 1511C-1 PN]	3 📲 🔹 ActValue	[1] Byte	16#0		Image: A start and a start	V				Whole words only	
T Device configuration	4 - ActValue	[2] Byte	16#0	V	Image: A start and a start	Image: A start and a start					
😼 Online & diagnostics	5 📲 🔹 ActValue	[3] Byte	16#0		V	V				Match case	
 Software units 	6 - ActValue	[4] Byte	1680	V	Image: A start and a start	 Image: A start of the start of				Find in substructures	
Add new software unit	7 📹 🔹 ActValue	[5] Byte	16#0	Image: A start and a start	V	V				Find in hidden texts	
 Program blocks 	8 🔩 = ActValue	[6] Byte	16#0		V					Use wildcards	
Add new block	🗏 9 📲 🔹 ActValue	[7] Byte	16#0		1	V					
4 Main [081]	10 🔩 = ActValue	[8] Byte	16#0			Image: A start and a start				Use regular expressions	
🛢 AdData (DB1)	11 - ActValue	[9] Byte	16#0		Image: A start and a start	2				Down	
WrData [D84]	12 - ActValue	[10] Byte	16#0								
 System blocks 	13 🚭 🍨 ActValue	[11] Byte	16#0	Image: A start and a start		1				⊖ Up	
 Program resources 	14 😋 = ActValue	[12] Byte	16#0		Image: A start and a start	Image: A start and a start				Find -	
RDREC_DB (DB2)	15 - ActValue	[13] Byte	16#0								
WRREC_DB [DB3]	16 🔩 = ActValue	[14] Byte	16#0			Image: A to				Replace with:	
Technology objects	17 - ActValue	[15] Byte	16#0								
External source files	18 😋 = ActValue		16#0			2				Whole document	
PLC tags	19 - ActValue		16#0								
PLC data types	20 <table-cell-columns> = ActValue</table-cell-columns>	[18] Byte	16#0							From current position	
Add new data type	21 - ActValue		16#0							O Selection	
8 DataInStruct	22 - ActValue		16#0							Replace Replace all	
B DataOutStruct	23 - ActValue	[21] Byte	16#0			2				incharge incharge an	
8 RdRecParaStruct	24 - ActValue		16#0								
B SD4x Datain	25 - ActValue		1680	Ĩ	2	Ĩ					
B SD4x DataOut	26 - ActValue		16#0								
B ServiceDataStruct	27 - ActValue		1680								
R WrRecParaStruct	28 - ActValue		16#0								
Watch and force tables	29 - ActValue		1680		Ĩ	Ÿ					
Contine backups	30 C ActValue		16#0	v V							
Traces	31 di ActValue		16#0		Ĩ	Ĩ					
OPC UA communication	32 ActValue		16#0	v V							
Bevice proxy data	33 C ActValue		16#0		×	×.					
2 Program info	- Action	(a.) oyle			0	2					
PLC supervisions & alarms						_				-	
PLC alarm text lists	ActValue							Properties Linfo Diagnos	tics 👘 🗆 🖛 🔻		
Local modules	General Text	s Supervisions									
Distributed I/O											
Details view	General Attributes	General							-	-	
become meth	Attributes										
			Name A								
Name Offset Data type			Data type As	ray[0_31] of Byte						> Languages & resources	

Fig. 11: PLC data types for the input data

		TIA_SD4_	PNIO + PLC_1 [CPL	1511C-1 PN] > PLC d	ata types 🕨 Da	ataOutStruct			- 0	X Tasks
vices										Options
	1	99								8
	¥									
		DataO	utStruct							 Find and replace
TIA_SD4_PNIO	^	Nem		Data type	Default value	Accessible f.			Comment	
Add new device			SptValue	Array[031] 🔳						Find:
Devices & networks		2 💶 🔹	SptValue[0]	Byte	16#0					
PLC_1 [CPU 1511C-1 PN]		3 📢 🔹	SptValue[1]	Byte	16#0	Image: A start and a start	V	~		Whole words only
Device configuration		4 🕙 •	SptValue[2]	Byte	16#0					Match case
V Online & diagnostics		5 📲 🔹	SptValue[3]	Byte	16#0		V			
 Software units 		6 🔩 🖷	SptValue[4]	Byte	1680	1	Image: A start and a start	Image: A start of the start		Find in substructures
Add new software unit		7 📲 🔹	SptValue[5]	Byte	16#0	2	Image: A start and a start			Find in hidden texts
 Program blocks 		8 🔩 🖷	SptValue[6]	Byte	16#0	1	V	1		Use wildcards
Add new block		9 🚭 🔹	SptValue[7]	Byte	16#0	1	1	2		
4 Main [081]		10 🔩 🔹	SptValue[8]	Byte	16#0	1	V	v		Use regular expressions
🝵 RdData (DB1)		11 📲 🔹	SptValue[9]	Byte	16#0		Image: A start and a start	2		Down
WrData [DB4]		12 💶 =	SptValue[10]	Byte	16#0		Image: A start and a start	Image: A start and a start		
 System blocks 		13 📲 🔹	SptValue[11]	Byte	16#0	1	V	1		Oup
 Program resources 		14 🔩 =	SptValue[12]	Byte	16#0					Find
RDREC_DB (DB2)		15 - 1	SptValue[13]	Byte	16#0	Image: A start and a start	Image: A start and a start	Image: A start and a start		
WRREC_D8 [D83]		16 📲 🔹	SptValue[14]	Byte	16#0			Image: A start and a start		Replace with:
Technology objects		17 - 17	SptValue[15]	Byte	16#0			Image: A start and a start		
External source files		18 📲 =	SptValue[16]	Byte	16#0	v				(*) Whole document
PLC tags		19 📲 🖷	SptValue[17]	Byte	16#0	V		Image: A start and a start		
PLC data types		20 🔩 🔹	SptValue[18]	Byte	16#0	Image: A start and a start				From current position
Add new data type		21	SptValue[19]	Byte	16#0	Image: A start and a start		2		Selection
B DataInStruct		22 💶 =	SptValue[20]	Byte	16#0			N		Replace Replace all
E DataOutStruct		23	SptValue[21]	Byte	16#0			Y		meprace meprace an
RdRecParaStruct		24 🕣 🔹	SptValue[22]	Byte	16#0					
SD4x_Datain		25	SptValue[23]	Byte	16#0			< </td <td></td> <td></td>		
8 SD4x DataOut		26 -	SptValue[24]	Byte	16#0			2		
8 ServiceDataStruct		27	SptValue[25]	Byte	16#0					
B WrRecParaStruct		28 -	SptValue[26]	Byte	16#0					
Watch and force tables		29	SptValue[27]	Byte	16#0					
Online backups		30	SptValue[28]	Byte	16#0			N N N		
Traces		31 4	SptValue[29]	Byte	16#0		Ĩ	 Image: A mail of the second sec		
OPC UA communication		32	SptValue[30]	Byte	16#0	v V		¥		
Device proxy data		33 4		Byte	16#0		Ĩ			
22 Program info			sheer of a class				۲	2		
PLC supervisions & elerms		-								
PLC alarm text lists		SptValue							Properties 📜 Info 😩 Diagnostics	
Local modules		General	Texts Sug	ervisions						
Distributed I/O	~		Tients John	0						100
		General		General						<u>^</u>
etails view		Attribute:								

DataOutStruct:

Fig. 12: PLC data types for the output data



12. Add new PLC variables in the standard variable table and link these to the input and output data structures.

Example:

InData_0	DataInStruct	%I124.0
OutData_0	DataOutStruct	%Q126.0

13. Create your Main program now.

How to assign the program variables to the process data of the drive is described in <u>chapter 3.4 "Cyclic Process Data (PDO)"</u>, page 15.

🎐 🔚 Save project 📲 🐰 🗓 🗋 🗙 🍋 🛨	(* ± 🐻 🛄 🛅 🚆 🕼 💋 Goonline)	🖉 Go offline 🕌 🖪 🖪 🗶 🚍	Search in projects		Totally Integrated Auto	PORT
roject tree	□	U 1511C-1 PN] Program block	s + Main [OB1]	_ # = ×	Instructions	11
Devices					Options	_
8	비장 운영 등: 씨는 영상	2 + 😥 🍋 📞 🖉 🖓 👘 🖏	= 표 표 밖 느 날 10 십 년 8 약 약 일	8	est est	r • 🗆 I
N	Main			-	> Favorites	
TA_SD4_PNIO	Name	Data type Default	value Comment			_
Add new device	1 - Input	our gre			✓ Basic instructions	
A Devices & networks	2 😋 = Initial_Call	Bool	Initial call of this OB		Name	Descr.
* PLC_1 (CPU 1511C-1 PN)	3 - Remanence	Bool	=True, if remanent data are available		Bit logic operations	
Device configuration	4 🤜 🔻 Temp				Timer operations	
& Online & diagnostics	5 - ErrorCode	Word			Counter operations	
 Software units 	6 - Pattern	Byte			Comparator operations	
Add new software unit	7 - bError	Array[01] of Byte			Math functions	
Program blocks	8 - NetX InData	"DataInStruct"			Move operations	
Add new block	9 C NetX OutData	"DateOutStruct"			Conversion operations	
P Main [OB1]	10 - Statusword	Wheel			 Program control operati 	
RdData [DB1]	11 - OK	DWord		N	 Word logic operations 	
WData [D84]			Part Part		Shift and rotate	
 Bystem blocks 	IF OF TO.D., DO (*) REGION			K Lenacy	
B Program resources						
RDREC_DB [DB2]	1 // get cyclic a 2 #NetX InData :=				 Extended instructions 	-
WRREC_DB [DB2]			a.ActValue[0]) OR SHL(IN := BYTE_TO_WORD(IN := #NetX_InData.ActValue[1]),	N 1- 011	Name	Desc
Technology objects			<pre>.ActValue[10]) OR SHL(IN := BYTE_TO_WORD(IN := #NetX_InData.ActValue[11]), .ActValue[10]) OR SHL(IN := BYTE_TO_WORD(IN := #NetX_InData.ActValue[11]),</pre>		D_ACT_DP	Enal
Gennology objects General source files	2 5	inclusion in the factor included	second a (10)) of public to provide the provide second a (11)),	1 1- 077	ReconfigIOSystem	Reco
PLC tags	6 // set cyclic s	etpoint data			 Others 	
Show all tags		<pre>"CounterVall" + 1;</pre>			RD_REC	Read
	8 #NetX_OutData.S				WR_REC	White
💣 Add new tag table 🙀 Standard-Variablentabelle [76]	9 #NetX_OutData.S				DPRD_DAT	Read
PLC data types		ptValue[2] := WORD_TO_BYIE("Co			DPWR_DAT	White
			<pre>(IN:=("CounterVal1" AND 16#FF00), N:=8));</pre>		iDevice / iSlave	
Watch and force tables	12 "OutData_0" :=	Metx_OutData;	10:26	CI: 28 INS 100%	RCVREC	Rec
Conline backups					PRVREC	Mak
Traces	Main [OB1]		<u>S</u> , Pr	roperties 🚺 Info 🕦 🖫 Diagnostics 👘 🗆 🚽	PROFIBUS	
OPC UA communication	General Texts				DPSYC_FR	Syne
 Device proxy data 		0			DPNRM_DG	Read
22 Program info	General	Compilation				Dire
PLC supervisions & alarms					<	
PLC alarm text lists	Time stamps	Status			✓ Technology	
Local modules	Compilation				Name	Desc
Distributed I/O	Protection		The loadable binary component is up to date.		Counting and measurem	G.
🖳 Ungrouped devices	Attributes	Compiled	on: 6/20/2023 - 12:09:02 PM		PID Control	
becurity settings		* Last load	ed: 6/20/2023 - 12:09:17 PM		Motion Control	
Cross-device functions					SINAMICS	
🖬 Common data		, Lengths			Time-based IO	
Documentation settings	~	Lenguis			1	
Details view		tradework.	10/05 0			
			ry: 19695 Bytes			-
			iny: 971 Bytes			
		Work mem	ny. Exit oper			
		Work mem	nj. Ezitupiez		>	

Fig. 13: Create Main program



14. Compile the hardware and the software.

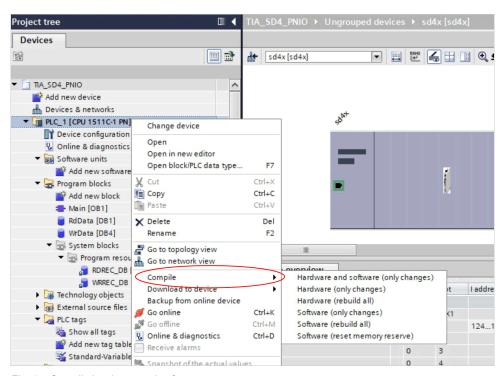


Fig. 14: Compile hardware and software

15. Download the hardware and the software to the PLC.

Project tree		TIA_SD4	_PNIO ▶ Ung	rouped devices	→ sd4x [sd4
Devices					
1 Alexandree Alexandre		sd4>	[sd4x]	-	
TIA_SD4_PNIO	^				
Add new device					
Devices & networks				+	
PLC_1 [CPU 1511C-1 PN]	Channes devices		çà	x+	
Device configuration	Change device				
😵 Online & diagnostics	Open				
▼ BB Software units	Open in new editor			_	
Add new software	Open block/PLC data type	F7			2
Program blocks	X Cut	Ctrl+X		1	i.
Add new block	Copy	Ctrl+C	_		<u>.</u>
Hain [OB1]	Paste	Ctrl+V			
📒 RdData [DB1]	X Delete	Del			
👅 WrData [DB4]	Rename	F2			
 System blocks 	R c				
🔻 🕞 Program resou	🚰 Go to topology view 🚠 Go to network view				
BRDREC_DB			overview		
WRREC_DB					
Technology objects	Download to device	L L		nd software (only c	hanges) t
External source files	Backup from online device	10052-027-0	Hardware co	-	
🕶 🌄 PLC tags	💋 Go online	Ctrl+K	Software (or		(1
how all tags	Go offline	Ctrl+M	Software (al		
	9. Online & diagnostics	Ctrl+D			0 2

Fig. 15: Load hardware and software

- 16. Start your PLC in RUN mode.
- 17. Use the *drivemaster4* software to check that your data are correctly transmitted to the drive.
- 18. With the PLC in the online mode, check that the drive receives the data correctly.



3.3 **PROFINET IO Network Management (NMT)**

The drive communication is controlled by the PLC via the PROFINET IO network management. The PROFINET IO network management coordinates the controller and device applications during startup and operation.

Status	Value	Description
UNKNOWN	0	Unknown status
OFFLINE	1	The drive configuration is not valid.
STOP	2	The drive cannot communicate with the controller. No connection is established. The bus status of the drive can be set to on or off .
IDLE	3	The communication is being established.
OPERATE	4	The I/O connection is established. The controller and the drive ex- change valid I/O data.

3.4 Cyclic Process Data (PDO)

The process data channel enables cyclic data exchange between the PLC as *controller* and drive as *device*.

The meaning of the drive data is defined in the *drivemaster4* software via the parameter page "Fieldbus \rightarrow PDO mapping".

3.4.1 Setpoint Telegram

The setpoint telegram for controlling the drive consists of 32 byte user data. The setpoint of the PLC program must be mapped to these bytes.

Byte	Name	Bit	Description
0	Controlword	0	Switch on
	(low byte)	1	Enable voltage
	Bits coded acc. to DS402 standard (object 6040h)	2	Quick stop
		3	Enable operation
		4	
		5	Operating mode specific
		6	
		7	Fault reset
1	Controlword	8	Hold
	(high byte)	9	Operating mode specific
	Bits coded acc. to DS402 standard (object 6040h)	10	Reserved
		11	
		12	
		13	Manufacturer specific
		14	
		15	
2 3 4 5	Target velocity (object 60FFh)		Target velocity byte 0 Target velocity byte 1 Target velocity byte 2 Target velocity byte 3
6 7	Max torque (object 6072h)		Max torque byte 0 Max torque byte 1
8 31	Not used		= 0

Example of the setpoints in profile velocity mode:



If a 2-byte object is to be sent, this object must be split into 2 bytes of the setpoint telegram.

Example: DS402 max torque (object 6072h)

```
MaxTorqueByte0 = WORD_TO_BYTE (MaxTorque AND 16#00FF)
MaxTorqueByte1 = WORD_TO_BYTE (SHR (IN := (MaxTorque AND 16#FF00), N := 8))
```

If a 4-byte object is to be sent, this object must be split into 4 bytes of the setpoint telegram.

Example: DS402 target velocity (object 60FFh)

TargetVelocityByte0	=	DWORD TO BYTE	(Targ	getVe	eloc	ity AND 16#0000	OOFF);			
TargetVelocityByte1	=	DWORD TO BYTE	(SHR	(IN	:=	(TargetVelocity	AND	16#0000FF00),	Ν	:=	8));
TargetVelocityByte2	=	DWORD TO BYTE	(SHR	(IN	:=	(TargetVelocity	AND	16#00FF0000),	Ν	:=	16));
TargetVelocityByte3	=	DWORD TO BYTE	(SHR	(IN	:=	(TargetVelocity	AND	16#FF000000),	Ν	:=	24));

3.4.2 Actual Value Telegram

The actual value telegram for controlling the drive consists of 32 byte user data. The actual values of the PLC program must be mapped to these bytes.

Byte	Name	Bit	Description
0	Statusword	0	Ready to switch on
	(low byte) Bits coded acc. to DS402	1	Switched on
	standard (object 6041h)	2	Operation enabled
		3	Fault
		4	Voltage enabled
		5	Quick stop
		6	Switch on disabled
		7	Warning
1	Statusword	8	Manufacturer specific: setup mode active
	(high byte)	9	Remote operation
	Bits coded acc. to DS402 standard (object 6041h)	10	Target reached
		11	Internal limit reached
		12	Operating mode an effect record
		13	Operating mode specific: reserved
		14	Manufacturer specific: status Safe Torque Off (STO)
		15	Manufacturer specific: initialization finished
2 3 4 5	Velocity actual value (object 606Ch)		Velocity actual value byte 0 Velocity actual value byte 1 Velocity actual value byte 2 Velocity actual value byte 3
6 7 8 9	Position actual value (object 6064h)		Position actual value byte 0 Position actual value byte 1 Position actual value byte 2 Position actual value byte 3
10 11	Error code (object 603Fh)		Error code byte 0 Error code byte 1

Example of the actual values in profile velocity mode:

If a 2-byte object is to be read, this object must be split into 2 bytes of the actual value telegram.

Example: DS402 error code (object 603Fh)

ErrorCode = BYTE TO_DWORD (ErrorCodeByte0) OR SHL (IN := BYTE_TO_DWORD (ErrorCodeByte1), N := 8);



If a 4-byte object is to be read, this object must be split into 4 bytes of the actual value telegram.

Example: DS402 position actual value (object 6064h)

```
PositionActualValue = BYTE_TO_DWORD (PositionActualValueByte0) OR
SHL (IN := BYTE_TO_DWORD (PositionActualValueByte1), N := 8) OR
SHL (IN := BYTE_TO_DWORD (PositionActualValueByte2), N := 16) OR
SHL (IN := BYTE_TO_DWORD (PositionActualValueByte3), N := 24);
```

3.5 Acyclic Service Data (SDO)

The service data channel enables acyclic data exchange between the PLC as *controller* and drive as *device*. Via this channel the PLC can acyclically read individual objects from the drive or write them to the drive.

The transmission is executed via the communication function blocks RDREC and WR-REC of the PLC. For information on the functionality, refer to the document "PROFIBUS and PROFINET Guideline, Communication Function Blocks on PROFIBUS DP and PROFINET IO".

Complying with the standard, the index is limited to 16 bit. If the PLC manufacturer uses a 32-bit index value, only the lower 16 bit can be used.

A maximum length of 1024 byte can be read or written with one call.

Using the 16-bit index, the PLC can address each object in the object dictionary of the drive. Simple objects with a maximum length of 4 byte are read or written directly. If the PLC must address an object with subindex or an array object, the subindex or array offset must be transmitted in advance. For this purpose, the object 158 (0x009E) "PROFINET IO – Subindex and offset" is available. The lowest byte of this object is used for the subindex and the upper 3 bytes are used for the offset:

Byte 3	Byte 2	Byte 0	
	Offset		Subindex

The following data apply to the offset and subindex of object 158 "PROFINET IO – Subindex and offset":

Object 158	Offset	Subindex
Value range	0 to 16777215 (0x00 to 0xFFFFF)	0 to 255 (0x00 to 0xFF)
Default value	0	0

After each write or read operation, except for object 158, the subindex and the offset are internally reset to zero.



3.5.1 Write and Read Operations

The following figure show the principle operations for writing and reading acyclic service data.

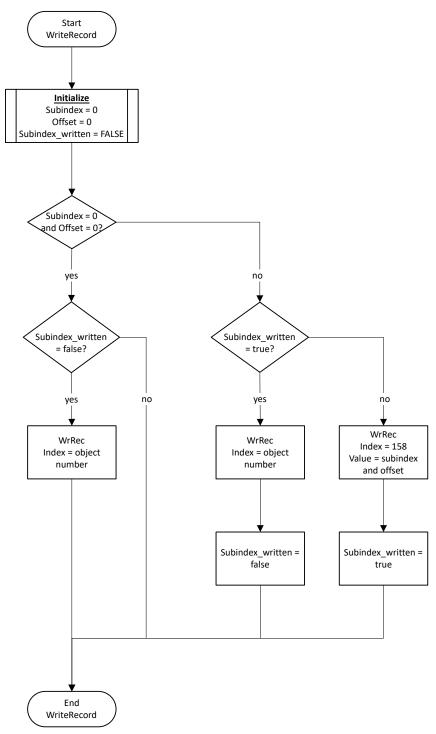


Fig. 16: Writing operation in principle



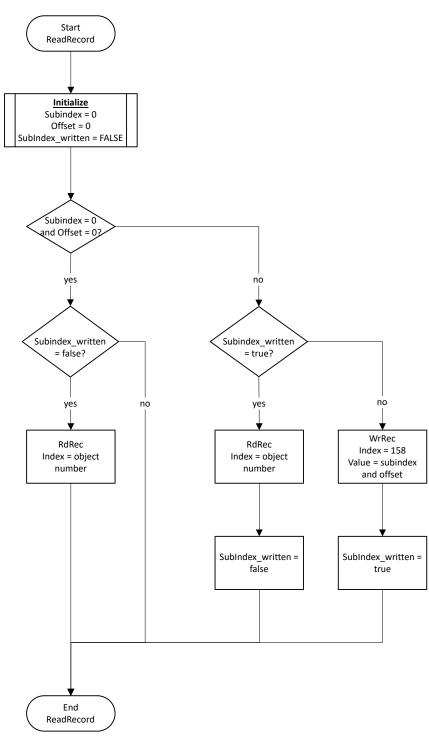


Fig. 17: Reading operation in principle

3.5.2 **Program Example for Siemens PLC**

The following program example was created with TIA PORTAL for a Siemens PLC S7-1500.

- 1. Add another block in "Program blocks".
 - a. Select the type "Global-DB" for the new block.
 - b. Add a title and comment as needed.



- c. Click "OK" to confirm your input.
- \rightarrow The new block is created below "Program blocks".
- 2. Select the block and add the following variables:

Name	Туре
ReadRequest	BOOL
WriteRequest	BOOL
ReadRecord	ARRAY[01200] OF BYTE
WriteRecord	ARRAY[01200] OF BYTE

Now, you must add the write and read operations to the program example "Main (OB1)" by means of the instances of the functions for WRREC und RDREC.

- 3. Add WriteRecord in the program example "Main (OB1)" as follows:
 - a. Enter "WRR" in Main.
 - \rightarrow A selection list appears.
 - b. Select the entry "WRREC" in the selection list.
 - \rightarrow The dialog window "Call options" appears.
 - c. In the window "Call options", enter the desired name for the data block, e.g. . "WRREC_DB_9". Click "OK" to confirm the input.

Call options	×
	Data block
	Name WRREC_DB_9
DB	Number 9
Single	🔘 Manual
	 Automatic
	If you call the function block as a single instance, the function block saves its data in its own instance data block.
	more
	OK Cancel

Fig. 18: Call options - WRREC



→ The block is created below "Program blocks → System blocks → Program resources".

In addition, the call is added in the Main program:

💌 🛃 Program blocks				CASE	FOR	WHILE	
📑 Add new block			IF	OF	TO DO	DO	(**) REGION
🍲 Main [OB1]	=			1	WRREC	DB 9	"(REQ:= bool in ,
🥃 Global_Data [DB3]				2			ID:= hv io in ,
 System blocks 				3			INDEX:=_dint_in_,
 Program resources 				4			LEN:=_uint_in_,
RDREC_DB [DB2]				5			DONE=>_bool_out_,
WRREC_DB [DB1]				6			BUSY=>_bool_out_,
WRREC_DB_1 [DB4]				7			ERROR=> bool out,
WRREC_DB_2 [DB5]				8 9			STATUS=>_dword_out_, RECORD:= variant inout);
WRREC_DB_3 [DB6]				10			RECORD Vallant_Indut_),
WRREC_DB_4 [DB7]		S I		11			
WRREC_DB_5 [DB8]		8	1	12			
WRREC_DB_9 [DB9]		2	1	13			

Fig. 19: Main – WRREC

d. You can adapt the call parameters for your application, now.

Example:

```
"WRREC_DB_9"(REQ := "Global_Data".WriteRequest,
ID := "sd4x~Head",
INDEX := 16#60C5,
LEN:=4,
RECORD:="Global Data".WriteRecord[0]);
```

Program code in principle:

```
// set acyclic data
"Global_Data".WriteSubIndexOffset := 16#0200000;
IF "Global_Data".WriteSubIndexOffset = 0 THEN
IF NOT "Global_Data".ReadSIOwritten THEN
"WRREC_DB_1"(REQ := "Global_Data".WriteRequest,
ID := "sd4x~Head",
INDEX := 16#6073,
LEN := 2,
RECORD := "Global_Data".WriteCurrent);
END IF;
ELSE
IF "Global_Data".WriteSIOwritten THEN
"WRREC_DB_2"(REQ := "Global_Data".WriteRequest,
ID := "sd4x~Head",
INDEX := 16#604A,
LEN := 2,
RECORD := "Global_Data".WriteRecord);
IF "WRREC_DB_2".DONE THEN
"Global_Data".WriteSIOwritten := FALSE;
END_IF;
ELSE
"WRREC_DB_3"(REQ := "Global_Data".WriteRequest,
ID := "sd4x~Head",
INDEX := 158,
LEN := 4,
RECORD := "Global_Data".WriteSubIndexOffset);
IF "WRREC_DB_3".DONE_THEN
"Global_Data".WriteSIOWritten := TRUE;
END_IF;
END_IF;
END_IF;
END_IF;
END_IF;
```

- 4. Add **ReadRecord** in the program example "Main (OB1)" as follows:
 - a. Enter "RDR" in Main.
 - \rightarrow A selection list appears.
 - b. Select the entry "RDREC" in the selection list.
 - \rightarrow The dialog window "Call options" appears.



c. In the window "Call options", enter the desired name for the data block, e.g. . "RDREC_DB_10". Click "OK" to confirm the input.

Call options	×
Call options B Single instance	Data block Name RDREC_DB_10 Number 0 Manual Automatic If you call the function block as a single instance, the function block saves its data in its own instance data block. more
	OK Cancel

Fig. 20: Call options - RDREC

→ The block is created below "Program blocks → System blocks → Program resources".

In addition, the call is added in the Main program:

💌 🙀 Program blocks	CASE FOR WHILE
💣 Add new block	IF CASE FOR WHILE (**) REGION
🖶 Main [OB1]	■ 1 "RDREC DB 10"(REQ:= bool in ,
🥃 Global_Data [DB3]	2 ID:= hv io in ,
🔻 🔙 System blocks	3 INDEX:= dint_in_,
🔻 🕞 Program resources	4 MLEN:= uint_in_,
RDREC_DB [DB2]	5 VALID=>_bool_out_,
RDREC_DB_10 [DB10]	6 BUSY=>_bool_out_,
WRREC_DB [DB1]	7 ERROR=>_bool_out_,
WRREC_DB_1 [DB4]	8 STATUS=> dvord_out_,
WRREC_DB_2 [DB5]	9 LEN=>_uint_out_, 10 RECORD:= variant inout);

Fig. 21: Main – RDREC

d. You can adapt the call parameters for your application, now.

Example:

```
"RDREC_DB_10"(REQ:="Global_Data".ReadRequest,
ID:="sd4x~Head",
INDEX:=16#604A,
MLEN:=4,
RECORD:="Global_Data".ReadRecord[0]);
```

Program code in principle:

```
// get acyclic data
"Global_Data".ReadSubIndexOffset := 16#01000000;
IF "Global_Data".ReadSubIndexOffset = 0 THEN
IF NOT "Global_Data".ReadSIOwritten THEN
"RDREC_DB"(REQ := "Global_Data".ReadRequest,
ID := "sd4x~Head",
INDEX := 16#6073,
MLEN := 2,
RECCRD := "Global_Data".ReadRecord);
END_IF;
ELSE
IF "Global_Data".ReadSIOwritten THEN
"RDREC_DB"(REQ := "Global_Data".ReadRequest,
ID := "sd4x~Head",
INDEX := 16#604A,
```



```
MLEN := 4,
RECORD := "Global_Data".ReadRecord);
IF "RDREC_DB".VALID THEN
"Global_Data".ReadSIOwritten := FALSE;
END_IF;
ELSE
"WRREC DB"(REQ := "Global_Data".ReadRequest,
ID := "sd4x~Head",
INDEX := 158,
LEN := 4,
RECORD := "Global_Data".ReadSubIndexOffset);
IF "WRREC_DB".DONE THEN
"Global_Data".ReadSIOwritten := TRUE;
END_IF;
END_IF;
END_IF;
```

 Add the processes for the write and read operation described above in your program. For this purpose, put the data to be written into Global_Data.WriteRecord. Control the reading and writing of the objects via Global_Data.WrRequest and Global_Data.RdReqest.



4 Drive Control

The drive control in devices of the product family SD4x is implemented according to the DS402 standard.

This standard defines objects that are divided into process data objects (PDO) and service data objects (SDO). PDOs are exchanged cyclically (real-time capable) and SDOs are exchanged acyclically (upon request, not real-time capable) via fieldbus.

The drive follows a predefined state machine, which is controlled by the commands of the <u>controlword</u> and returns the current status via the <u>statusword</u>. By means of the <u>op-tion codes</u> you can change the shutdown behavior of the drive for the corresponding shutdown commands. The used target and actual values depend on the selected operating mode (<u>profile velocity mode</u>, <u>velocity mode</u> or <u>torque mode</u>). How to switch the operating mode is described under <u>change operating mode</u>.

The error codes are hardware-dependent. For a detailed description of the error codes refer to the hardware documentation of your device.

Object descriptions

All relevant objects are described in table form with the following information:

	Short name of the object		Short description of the object		
Explicit name of the object Object			Object index		
	Access ⁽¹⁾	Data memory ⁽²⁾	Unit	Data type	

⁽¹⁾ Possible values:

RW = read/write RO = read only

itto – read only

⁽²⁾ Possible Values: Volatile = data is lost after power cut-off

Non-volatile = data is retained after power cut-off



4.1 State Machine

The state machine (in the DS402 standard: finite state automaton, FSA) controls the power electronics of the drive. The following diagram shows the states with their meaning and the sequence of the state machine.

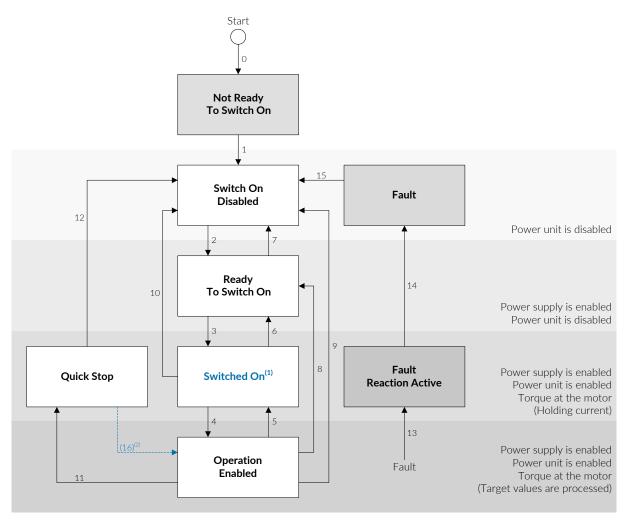


Fig. 22: Drive states and state transitions

- (1) Unlike in the DS402 standard, the output stage of SD4x drives is already enabled in the status "Switched On" so that torque is applied to the motor. In this status the target values are not yet enabled and the motor is held.
- (2) The transition 16 is not recommended by the DS402 standard anymore. Therefore, SD4x drives do not support transition 16.

The number at the arrows are transitions triggered by commands. Which command triggers which transition is described in <u>chapter 4.1.1 "Commands"</u>, <u>page 26</u>. The boxes represent the states of the state machine. On the right side of the figure, you find the corresponding states of the power electronics. The gray-colored boxes contain states that the state machine reaches automatically and not via commands.



4.1.1 Commands

Command	Bit 7 Fault reset	Bit 3 Enable operation	Bit 2 Quick stop	Bit 1 Enable voltage	Bit 0 Switch on	Transition
Shutdown	0	Х	1	1	0	2, 6, 8
Switch on	0	X ⁽²⁾	1	1	1	3
Disable voltage	0	Х	Х	0	Х	7, 9, 10, 12
Quick stop	0	Х	0	1	Х	7, 10, 11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, (16) ⁽³⁾
Fault reset	£1 ⁽⁴⁾	Х	Х	Х	Х	15

The control commands depend on the currently active status. They are triggered via setting the following bit patterns in the controlword:

 $^{\left(1\right)}$ The bits labeled with "X" have no relevance at the corresponding position in the table.

⁽²⁾ If the bit is 1, the state machine automatically switches after execution in "Switched on" to the next status "Operation enabled".

⁽³⁾ The transition 16 is not recommended by the DS402 standard anymore. Therefore, SD4x drives do not support transition 16.

⁽⁴⁾ Positive edge (change from 0 to 1, pulse)

The bits not mentioned in the controlword are irrelevant for the commands. If a command is unknown for a status, it is ignored. The transitions not mentioned happen automatically in the state machine and cannot be triggered via the commands.

Controlword (0x6040)

The controlword is a bit-coded WORD and transmits control demands to the drive.

Controlword		Control commands: Bit 0: Switch on Bit 1: Enable voltage Bit 2: Quick stop Bit 3: Enable operation Bit 4: Mode-specific 1 Bit 5: Mode-specific 2 Bit 6: Mode-specific 3 Bit 7: Fault reset Bit 8: Halt Bits 9 to 15: Reserved		
DS402_0x6040_	CONTROLWORD			ID: 0x6040
RW	Volatile	Unit: –	U16	

The commands for controlling the state machine are made up of the bits 0 (switch on), 1 (v), 2 (quick stop), 3 (enable operation) and 7 (fault reset).

The function "Halt" (bit 8) is only active in the status "Operation enabled". This function is used to stop the motor with the halt option code. Then, the motor is held. The status "Operation enabled" is maintained the entire time.

Information on the mode-specific bits 4 to 7 can be found under the respective operating mode.



4.1.2 States

The statusword returns the states of the state machine according to the following bit patterns. For the different states, the following bits are set:

	Bits of statusword ⁽¹⁾						
Status	Bit 6	Bit 5	Bit 3 Fault	Bit 2	Bit 1	Bit 0	
	Switch on disabled	Quick stop		Operation enabled	Switched on	Ready to switch on	
Not ready to switch on (firmware not ready)	0	х	0	0	0	0	
Switch on disabled	1	х	0	0	0	0	
Ready to switch on	0	1	0	0	0	1	
Switched on	0	1	0	0	1	1	
Operation enabled	0	1	0	1	1	1	
Quick stop active	0	0	0	1	1	1	
Fault reaction active	0	Х	1	1	1	1	
Fault	0	Х	1	Х	0	Х	

 $^{(1)}$ The bits labeled with "X" have no relevance at the corresponding position in the table.

The bits not mentioned in the statusword are irrelevant for the states.

Statusword (0x6041)

The statusword is a bit-coded WORD and contains status information of the drive.

Statusword		Device states:				
		Bit 0: Ready to switch on				
		Bit 1: Switched on				
		Bit 2: Operation enabled				
		Bit 3: Fault				
		Bit 4: Voltage enabled				
		Bit 5: Quick stop				
	Bit 6: Switch on disabled					
		Bit 7: Warning				
Bit 8: Drive setup tool (DRS) active*						
		Bit 9: Remote				
		Bit 10: Target reached				
		Bit 11: Current limit reached*				
		Bit 12: Mode-specific 1				
		Bit 13: Mode-specific 2				
		Bit 14: Safe Torque Off (STO)*				
		Bit 15: Initialization finished*				
		*Bit assignments shown in italics are SIEB & MEYER specific and				
may differ with other drive manufacturers.						
DS402_0x6041_	STATUSWORD			ID: 0x6041		
RO	Volatile	Unit: –	U16			

The bits 0 (ready to switch on), 1 (switched on), 2 (operation enabled), 3 (fault), 5 (quick stop) und 6 (switch on disabled) return the current status of the state machine via bit patterns. In the event of an error, the object *error code* (ID 0x603F) returns the error information.

Bit 4 (enable voltage) is active when the voltage for supplying the DC link is applied.

Bit 7 (warning) is active in the event of a warning message. The object *error code* (ID 0x603F) returns information on the current warning message.

Bit 8 (Drive Setup Tool (DRS) active) is SIEB & MEYER-specific and active when the *drivemaster4* software or another setup tool controls the drive.

Bit 9 (remote) is active when the drive is controlled by the control commands of the controlword.



Bit 10 (target reached) is active when the drive has reached the target value. This bit depends on the operating mode and the current status:

- Velocity modes VL/PV: bit 10 is set when the set target speed is reached.
- Status "Quick stop active": if the drive does not automatically exit the status "Quick stop active", bit 10 indicates that the quick stop function has been completed and the motor is being held.
- Changing the operating mode: the bit is set after the operating mode has been successfully changed.

Bit 11 (current limit reached) is SIEB & MEYER-specific and active when the drive is operated at the set current limit.

The bits 12 and 13 are mode-specific. For more information refer to the chapters on the operating modes.

Bit 14 (Safe Torque Off (STO)) is SIEB & MEYER-specific and active when the safety function STO keeps the drive switched off. An example would be an emergency stop that is connected with the SAFE inputs of the drive.

Bit 15 (initialization finished) is SIEB & MEYER-specific und becomes active as soon as the drive has been successfully initialized.

4.1.3 Option Codes

The option codes define the behavior of the drive on receipt of a stop command (e.g. disable operation) or in case of a shutdown event (e.g a fault or loss of connection to the control unit).

Abort Connection Option Code (0x6007)

This object defines the reaction of the drive to a loss of connection due to one of the following events: bus off status, heartbeat/node guarding error, NMT stopped status, reset of application or configuration.

Abort connectio	on option code	Selection of the reaction to a field 00: No action 01: Fault signal 02: Disable voltage command 03: Quick stop command	01: Fault signal 02: Disable voltage command			
Object: DS402_0	Object: DS402_0X6007_ABORT_CONNECTION_OPTION_CODE ID: 0x6007					
RW	Volatile	Unit: –	116			

Quick Stop Option Code (0x605A)

This object defines the reaction of the drive to a quick stop command.

Quick stop optio	on code	Selection of the reaction to a quick stop command: 00: Disable drive function 01: Slow down on slow down ramp and transit into 'switch on disabled' 02: Slow down on quick stop ramp and transit into 'switch on disabled' 03: Slow down on current limit and transit into 'switch on disabled'			
DS402_0X605A_	DS402_0X605A_QUICK_STOP_OPTION_CODE ID: 0x605A				
RW	Volatile	Unit: –	116		

The ramps are defined in the operating mode.



Shutdown Option Code (0x605B)

This object defines the reaction of the drive to a shutdown command, i.e. the drive switches from the status "operation enabled" to the status "ready to switch on".

Shutdown optio	n code	Selection of the reaction to the com -1: Slow down via short-circuiting th 00: Disable drive function (switch of 01: Slow down on slow down ramp,	he motor phases ff the drive power	output stage)
DS402_0X605B_SHUTDOWN_OPTION_CODE ID: 0x605B			ID: 0x605B	
RW	Volatile	Unit: –	116	

The slow down ramp is defined in the operating mode.

Disable Operation Option Code (0x605C)

This object defines the reaction of the drive to a disable operation command, i.e. the drive switches from the status "operation enabled" to the status "switched on".

Disable operation	,	Selection of the reaction to the command "disable operation": 01: Slow down on slow down ramp, disable drive function			
DS402_0X605C_DISABLE_OPERATION_OPTION_CODE ID: 0x6				ID: 0x605C	
RW	Volatile	Unit: –	116		

The slow down ramp is defined in the operating mode.

Note

The value is not intended to be changed.

Halt Option Code (0x605D)

This object defines the reaction of the drive to a halt command.

Halt option code	9	Selection of the reaction to the command "halt": 01: Slow down on slow down ramp and stay in 'operation enabled' 02: Slow down on quick stop ramp and stay in 'operation enabled' 03: Slow down on current limit and stay in 'operation enabled'		
DS402_0X605D	HALT_OPTION_	CODE		ID: 0x605D
RW	Volatile	Unit: –	116	

The ramps are defined in the operating mode.

Fault Reaction Option Code (0x605E)

This object defines the reaction of the drive to a fault.

Fault reaction o	ption code	Selection of the reaction to a fault: 00: Disable drive function, motor is 01: Slow down on slow down ramp 02: Slow down on quick stop ramp 03: Slow down on current limit 04: Slow down on voltage limit		
DS402_0X605D_HALT_OPTION_CODE				ID: 0x605E
RW	Volatile	Unit: –	116	

The ramps are defined in the operating mode.



4.2 Operating Modes

The drive supports several operating modes, which depend on the device variant, the motor, the connected sensors and the parameterized drive function.

The following chapters describe the available operating modes and how to change the operating mode.

4.2.1 PV

In order to control the drive in the operating mode 'profile velocity mode', at least the following objects are required: *controlword*, *statusword* and *target velocity*. It is recommended to use the *velocity actual value* as return value. If the motor is not equipped with a speed sensor, the drive calculates the actual speed from the rotating field that is currently applied to the motor.

The following diagram shows the used objects and examples of their function:

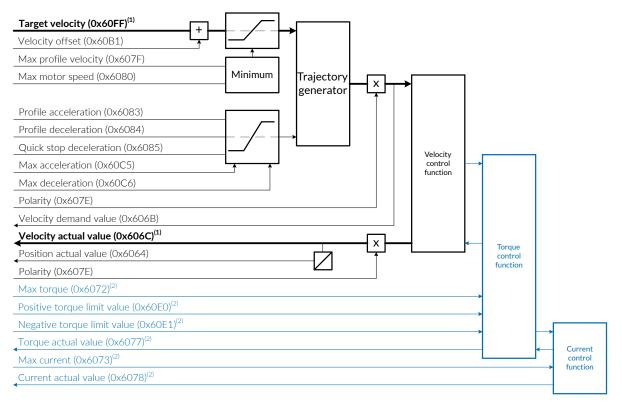


Fig. 23: Block diagram for profile velocity mode (PV)

- (1) The objects marked in bold are required and recommended process data (PDO) for the profile velocity mode.
- (2) SD4x drives support these objects (marked in blue) additionally to the standard objects in the DS402 drive profile.

PV-specific bits in the controlword and the statusword

The mode-specific bits 4, 5 and 6 in the controlword are optional bits and not supported in this operating mode.

The mode-specific bits in the statusword are assigned as follows:

- Bit 10 (target reached) is active when the difference between actual speed and target speed is longer than the speed window time within the speed window.
 You can set the speed window in the *drivemaster4* software under "Messages → Messages → Reference value reached [M51]".
- Bit 12 (speed zero) is deactivated when the speed actual value exceeds the speed threshold longer than the speed threshold time. Below this threshold the bit is active and indicates that the axis is at standstill.

You can set the speed window in the *drivemaster4* software under "Messages \rightarrow Messages \rightarrow Speed zero [M15]".

• Bit 13 (maximum slip) is active when the drive has reached the parameterized slip.



4.2.2 Velocity Mode (VL)

The velocity mode uses its own objects for scaling the target and actual values. By default, the speeds are specified in *revolutions per minute* [rpm]. You can change the unit via the scaling object **vI dimension factor**. In addition, the object **vI Set-point Factor** is used to scale the target and actual speed. Due to the object size, the resolution of the target and actual speed is limited to 16 bits in this operating mode. For applications that require a higher resolution, you can use the operating mode <u>PV, page 31</u>, which offers a 32 bit resolution for target and actual speeds.

In order to control the drive in the operating mode 'velocity mode', at least the following objects are required: *controlword*, *statusword* and *vI target velocity*. It is recommended to use the *vI velocity actual value* as return value.

The following diagram shows the used objects and examples of their function:

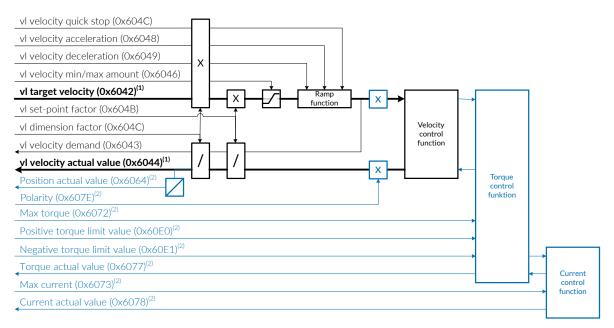


Fig. 24: Block diagram for velocity mode VL

- (1) The objects marked in bold are required and recommended process data (PDO) for the velocity mode.
- (2) SD4x drives support these objects (marked in blue) additionally to the standard objects in the DS402 drive profile.

VL-specific bits in the controlword and the statusword

The mode-specific bits 4, 5 and 6 in the controlword are optional bits and not supported by SD4x drives.

The mode.specific bits in the statusword are reserved in velocity mode according to the DS402 drive profile. However, SD4x drives use the bits 10 and 12 as follows:

- Bit 10 (target reached) is active when the difference between actual speed and target speed is longer than the speed window time within the speed window.
 You can set the speed window in the *drivemaster4* software under "Messages → Messages → Reference value reached [M51]".
- Bit 12 (speed zero) is deactivated when the speed actual value exceeds the speed threshold longer than the speed threshold time. Below this threshold the bit is active and indicates that the axis is at standstill.

You can set the speed window in the *drivemaster4* software under "Messages \rightarrow Messages \rightarrow Speed zero [M15]".



4.2.2.1 Target and Actual Values in Velocity Mode

vl Target Velocity (0x6042)

This object contains the target speed in velocity mode.

vl target velocit	У	Target speed in velocity mode When the objects <i>vI set-point factor</i> (0x604B) and <i>vI dimension fac- tor</i> (0x604C) are set to the value 1 (default), this object returns the speed value in rpm. Positive values cause forward motion, negative values cause backward motion.		
DS402_0x6042_	VL_TARGET_VEL	.OCITY		ID: 0x6042
RW	Volatile	Unit: rpm (default)	116	

vl Velocity Demand (0x6043)

This object returns the resulting target speed in velocity mode after ramp and limiting functions.

vl velocity dema	and	Speed setpoint in velocity mode The value indicates the speed setpoint after the ramp and limiting functions. The unit and direction correspond to object <i>vI target veloc-</i> <i>ity</i> (0x6042).		0
DS402_0x6043_VL_VELOCITY_D		EMAND		ID: 0x6043
RO	Volatile	Unit: rpm (default)	116	

vl Velocity Actual Value (0x6044)

This object returns the speed actual value in velocity mode.

vl velocity actua	al value	Actual speed in velocity mode The unit and direction correspond to object <i>vI target velocity</i> (0x6042).		
DS402_0x6044_VL_VELOCITY_ACTUAL_VALUE			ID: 0x6044	
RO	Volatile	Unit: rpm (default)	116	

Depending on the application (with or without a sensor) the speed actual value is either determined by a sensor or derived from the calculated rotating field.

vl Velocity Min Max Amount (0x6046)

Via this object you can limit the speed in velocity mode.

vl velocity min r	nax amount	Speed limitation in velocity mode Sub-IDs: • 0x6046:00 = Highest subindex supported (2) • 0x6046:01 = vl velocity min amount • 0x6046:02 = vl velocity max amount If the object vl dimension factor (0x604C) is set to the value 1 (de- fault), this object indicates the speed limits in rpm.		ne value 1 (de-
DS402_0X6046_VL_VELOCITY_M		IIN_MAX_AMOUNT		ID: 0x6046
RW	Volatile	Unit: rpm (default)	BYTEARRAY	



In its subindex 0, this object contains the last subindex of the object (in this case 2).

The following figure shows the limitation of the minimum and maximum speed via the object *vI velocity min max amount*. The dotted line shows the speed setpoint and the blue, continuous line shows the resulting target speed.

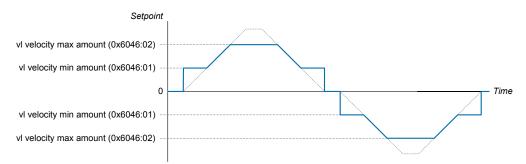


Fig. 25: Speed limitation in velocity mode

vl Velocity Acceleration (0x6048)

Via this object you can limit the acceleration in velocity mode.

vl velocity acce	leration	 Acceleration ramp in velocity mode Sub-IDs: 0x6048:00 = Highest subindex supported (2) 0x6048:01 = Delta speed If the object <i>vI dimension factor</i> (0x604C) is set to the value 1 (default), this object indicates the speed in rpm. 0x6048:02 = Delta time 		o the value 1
DS402_0x6048_	VL_VELOCITY_A	CCELERATION		ID: 0x6048
RW	Volatile	Unit: rpm/s (default)	DS402_ACCERLERATION_REC	

In its subindex 0, this object contains the last subindex of the object (in this case 2).

The following figure shows the limitation of the acceleration via the object *vl velocity acceleration*. The dotted line shows the speed setpoint and the blue, continuous line shows the resulting target speed.

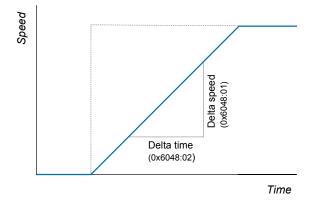


Fig. 26: Limitation of acceleration in velocity mode

Thus, the resulting acceleration ramp with default settings is specified in *revolutions per minute per second*.



vl Velocity Deceleration (0x6049)

Via this object you can limit the deceleration in velocity mode.

vl velocity dece	leration	 Deceleration ramp in velocity mode Sub-IDs: 0x6049:00 = Highest subindex supported (2) 0x6049:01 = Delta speed If the object <i>vI dimension factor</i> (0x604C) is set to the value 1 (default), this object indicates the speed in rpm. 0x6049:02 = Delta time The time is indicated in s. The value 0 is not permitted, in this case the minimum value (1 s) is used. 		
DS402_0x6049_VL_VELOCITY_D		DECELERATION ID: 0x60		ID: 0x6049
RW	Volatile	Unit: rpm/s (default)	DS402_ACCERL	ERATION_REC

In its subindex 0, this object contains the last subindex of the object (in this case 2).

The following figure shows the limitation of the deceleration via the object *vl velocity deceleration*. The dotted line shows the speed setpoint and the blue, continuous line shows the resulting target speed.

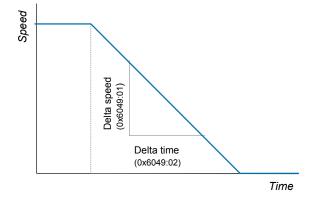


Fig. 27: Limitation of deceleration in velocity mode

Thus, the resulting deceleration ramp with default settings is specified in *revolutions per minute per second*.

vl Velocity Quick Stop (0x604A)

Via this object you can limit the deceleration in the event of a quick stop in velocity mode.

		 Quick stop ramp in velocity mode Sub-IDs: 0x604A:00 = Highest subindex supported (2) 0x604A:01 = Delta speed If the object <i>vI dimension factor</i> (0x604C) is set to the value 1 (default), this object indicates the speed in rpm. 0x604A:02 = Delta time The time is indicated in s. The value 0 is not permitted, in this case the minimum value (1 s) is used. 		
DS402_0x604A_VL_VELOCITY_Q		UICK_STOP		ID: 0x604A
RW	Volatile	Unit: rpm/s (default)	DS402_ACCERL	ERATION_REC



In its subindex 0, this object contains the last subindex of the object (in this case 2).

The following figure shows the limitation of the deceleration in the event of a quick stop via the object *vl velocity quick stop*. The dotted line shows the speed setpoint and the blue, continuous line shows the resulting target speed.

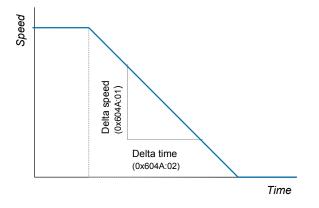


Fig. 28: Limitation of deceleration in the event of a quick stop in velocity mode

Thus, the resulting quick stop ramp with default settings is specified in *revolutions per minute per second*.

vl Set-point Factor (0x604B)

Via this object you can set an additional scaling factor for the speed target and actual values in velocity mode.

 vI set-point factor Scaling factor for the setpoint and the actual value in velocity Sub-IDs: 0x604B:00 = Highest subindex supported (2) 0x604B:01 = Numerator The numerator has no unit. The value 0 is not permitted, in case the last valid value is retained. 0x604B:02 = Denominator The denominator has no unit. The value 0 is not permitted, case the last valid value is retained. 		nitted, in this		
DS402_0X604B_VL_SET_POINT_		FACTOR		ID: 0x604B
RW	Volatile	Unit: –	DS402_ACCERLERATION_REC	

In its subindex 0, this object contains the last subindex of the object (in this case 2).

This unit scaling only applies to the objects *vl target velocity* (0x6042), *vl velocity demand* (0x6043) and *vl velocity actual value* (0x6044), i.e. not the ramp and limiting functions.

Scaling factor for setpoints = $\frac{Numerator (0x604B:01)}{Denominator (0x604B:02)}$

Application examples

- 1. The speed is to be specified in *revolutions per minute* with one decimal position:
 - Numerator remains set to 1.
 - Denominator is set to 10.

The values of the objects *vl target velocity*, *vl velocity demand* and *vl velocity actual value* are immediately reinterpreted: the value 101 means 10.1 rpm.

Note

This setting only allows speed settings/displays between -3276.8 and 3276.7 rpm.

- 2. Speeds above 32767 rpm must be set:
 - Numerator is set to 10.



- Denominator remains set to 1.

The values of the objects *vl target velocity*, *vl velocity demand* and *vl velocity actual value* are immediately reinterpreted: the value 101 means 1010 rpm.

Note

This setting allows speed settings/displays between -327680 and 327670 rpm. But the target and actual speeds can only be specified/displayed in rpm steps of 10.

vl Dimension Factor (0x604C)

Via this object all speed values in velocity mode are scaled. Beside the target and actual values this also applies to the ramp and limiting functions.

vl dimension fac	ctor	 Scaling factor for the speed unit in velocity mode Sub-IDs: 0x604C:00 = Highest subindex supported (2) 0x604C:01 = Numerator The numerator has no unit. The value 0 is not permitted, in this case the last valid value is retained. 0x604C:02 = Denominator The denominator has no unit. The value 0 is not permitted, in this case the last valid value is retained. 		
DS402_0X604C_	VL_DIMENSION	FACTOR		ID: 0x604C
RW	Volatile	Unit: –	DS402_ACCERL	ERATION_REC

In its subindex 0, this object contains the last subindex of the object (in this case 2).

The purpose of this scaling: Each user-specific speed consists of specific units in relation to time units (e.g. 1/s, cams/min, m/s etc.). The dimension factor converts these speed units to a common basis (e.g. revolutions per minute).

Scaling factor to $rpm = \frac{Numerator (0x604C:01)}{Denominator (0x604C:02)}$

When the scaling factor is changed, the ramp and limiting functions are automatically scaled to the new unit. This applies to the following objects: vI velocity min amount (0x6046:01), vI velocity max amount (0x6046:02), vI velocity acceleration (0x6048:01), vI velocity deceleration (0x6049:01) and vI velocity quick stop (0x604A:1). If a higher-ranking control exchanges one of these objects as PDO, the control must perform the scaling of the corresponding object.

Application example

The speed is to be specified in *revolutions per second* with one decimal position:

- Numerator is set to 60.
- Denominator is set to 10.

The values of the objects *vl target velocity*, *vl velocity demand* and *vl velocity actual value* are immediately reinterpreted: the value 32 means 3.2 1/s or 192 rpm.

Note

This setting allows speed settings/displays between -196608 and 196602 rpm. But the target and actual speeds can only be specified/displayed in rpm steps of 6.



4.2.3 Torque Mode

In order to control the drive in the operating mode 'torque mode', at least the following objects are required: *controlword*, *statusword* and *target torque*. It is recommended to use the *torque actual value* as return value.

The following diagram shows the used objects and examples of their function:

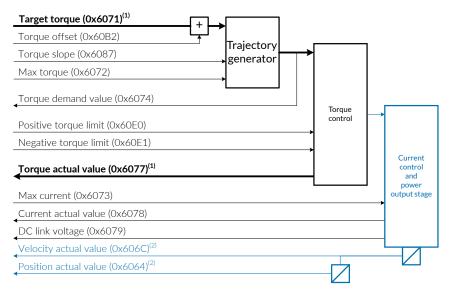


Fig. 29: Block diagram for torque mode

- (1) The objects marked in bold are required and recommended process data (PDO) for the profile velocity mode.
- (2) SD4x drives support these objects (marked in blue) additionally to the standard objects in the DS402 drive profile.

4.2.4 Change Operating Mode

You can easily switch between the available operating modes of the drive.

Risk of injury due to unintentional motor start

Note

As an alternative, SD4x drives allow switching the complete parameter set (see <u>chapter</u> <u>4.4 "Parameter Set Change", page 50</u>). You should consider this option when several parameters must be changed in addition to the operating mode.

A WARNING



Before you change the operating mode, set the target values to 0 in order to prevent unintentional starting of the motor.

The available operating modes are contained in the object **supported drive modes** (0x6502). You can change the operating mode using the object **mode of operation** (0x6060). The currently active operating mode is returned in the object **mode of operation** *ation display* (0x6061).

After the operating mode was successfully changed, bit 10 (target reached) in the *sta-tusword* (0x6041) is set.

The new operating mode is only applied and becomes active in the status "operation enabled". In this status the operating mode cannot be changed again.



A new operating mode setting is only applied when the status switches to "operation enabled". If the state machine is already in this status, it must leave the status "operation enabled" at first and then switch back into it.

Supported Drive Modes (0x6502)

This object contains the currently available operating modes for the drive as bit code. The bits set in the 32-bit object indicate that the respective operating modes are available.

Supported drive	modoo	Available operating modes:		
Supported drive	moues	Available operating modes:		
		Bit 00: pp – profile position mode		
		Bit 01: vl – velocity mode		
		Bit 02: pv – profile velocity mode		
		Bit 03: tq – profile torque mode		
		Bit 04: hmaf – homing mode with additional feedback		
		Bit 05: hm – homing mode		
		Bit 06: ip – interpolated position mode		
		Bit 07: csp – cyclic sync position mode		
		Bit 08: csv – cyclic sync velocity mode		
		Bit 09: cst – cyclic sync torque mode		
		Bit 10: cstca – cyclic sync torque m	ode with commuta	tion angle
		Bit 11: ppaf – profile position mode	with additional fee	dback
		Bit 12: pvaf – profile velocity mode	with additional fee	dback
		Bit 13: cspaf – cyclic sync position i		
		Bit 14: csvaf – cyclic sync velocity r		
		Bit 15: r – reserved		
DS402_0X6502_	SUPPORTED_DF	RIVE_MODES		ID: 0x6502
RO	Volatile	Unit: –	U32	

Mode of Operation (0x6060)

Via this object you can change the operating mode.

Mode of operati	on	Selection of the operating mode:		
		01: Profile position mode		
		02: Velocity mode		
		03: Profile velocity mode		
		04: Profile torque mode		
		06: Homing mode		
		07: Interpolated position mode		
		08: Cyclic sync position mode		
		09: Cyclic sync velocity mode		
		10: Cyclic sync torque mode		
		11: Cyclic sync torque mode with co	ommutation angle	
		12: Profile position mode with addit	ional feedback	
		13: Profile velocity mode with additi	onal feedback	
		14: Homing mode with additional fe	edback	
		15: Cyclic sync position mode with	additional feedbac	k
		16: Cyclic sync velocity mode with a	additional feedbac	k
DS402_0x6060_MODE_OF_OPERATION ID: 0x6			ID: 0x6060	
RW	Volatile	Unit: –	U8	

Not all operating modes are available. As standard, the SD4x drive supports the operating modes 2 (velocity mode), 3 (profile velocity mode) and 4 (profile torque mode).



Mode of Operation Display (0x6061)

This object returns the currently active operating mode.

Mode of operation display	Active operating mode:			
	01: Profile position mode			
	02: Velocity mode			
	03: Profile velocity mode			
	04: Profile torque mode			
	06: Homing mode			
	07: Interpolated position mod	07: Interpolated position mode		
	08: Cyclic sync position mode 09: Cyclic sync velocity mode			
	10: Cyclic sync torque mode	10: Cyclic sync torque mode		
	11: Cyclic sync torque mode	with commutation	angle	
	12: Profile position mode with	additional feedb	ack	
	13: Profile velocity mode with	additional feedba	ack	
	14: Homing mode with addition	onal feedback		
	15: Cyclic sync position mode	e with additional for	eedback	
	16: Cyclic sync velocity mode	with additional fe	eedback	
DS402_0X6061_MODE_OF	OPERATION_DISPLAY		ID: 0x6061	
RO Volatile	Unit: –	U8		

4.3 Target and Actual Values

The following chapters describe the target and actual values relevant for the drive profile (DS402). How the values are used is defined by the selected operating mode

Note

The velocity mode (vI) uses mostly its own target, actual , scaling and limiting values. These are directly described below the operating mode, see <u>page 33</u>.

4.3.1 General Target Values

Target Velocity (0x60FF)

This object indicates the target speed. The motor reaches this target value usually at the end of the acceleration ramp during a profile movement. With positioning operating modes, the target value applies to both moving directions.

Target velocity		Target speed in user units		
DS402_0x60FF_TARGET_VELOCITY ID: 0x60FF			ID: 0x60FF	
RW	Volatile	Unit: –	132	

The value is indicated in user-defined speed units. The default setting for SD4x drives

is 10^{-3} rpm. The speed unit is scaled with the factor group **SI unit velocity** (0x60A9). In addition, the object **velocity offset** (0x60B1) can affect the resulting target value. You can define additional speed and acceleration limits.

By default, you should set the drive system so that positive values cause forward motion and negative values cause backward motion.



Target Torque (0x6071)

This object indicates the target torque.

Target torque		Target torque The value is indicated in parts per thousand (1000 = 100 %) of the rated torque, object <i>motor rated torque</i> (0x6076).		
DS402_0X6071_TARGET_TORQUE ID: 0x607			ID: 0x6071	
RW	Volatile	Unit: ‰	116	

For the target torque you can set an offset via the object *torque offset* (0x60B2). Limitations are set via the objects *torque slope* (0x6087), *positive torque limit* (0x60E0) and *negative torque limit* (0x60E1).

4.3.2 Actual Values

Position Actual Value (0x6064)

This object returns the actual position determined by the position measuring system (e.g. encoder).

Position actual	value	Actual position in user units		
DS402_0X6064_POSITION_ACTUAL_VALUE ID: 0x6064			ID: 0x6064	
RO Volatile Unit: – I32				

As standard, the value is indicated in (encoder) increments, see object *SI unit position* (0x60A8). The number of increments per revolution is defined in the *drivemaster4* software in the feedback settings.

Velocity demand value (0x606B)

This object returns the resulting target speed after ramp and limiting functions.

Velocity deman	d value	Speed setpoint after ramp and limiting functions The unit and direction correspond to object <i>target velocity</i> (0x60FF).		
DS402_0x606B_VELOCITY_DEMAND_VALUE ID: 0x606B				ID: 0x606B
RO	Volatile	Unit: –	132	

Velocity Actual Value (0x606C)

This object returns the actual speed.

Velocity actual	value	Actual speed in user units The unit and direction correspond to object <i>target velocity</i> (0x60FF).		
DS402_0x606C_VELOCITY_ACTUAL_VALUE ID: 0x606			ID: 0x606C	
RO	Volatile	Unit: –	132	

Depending on the application (with or without a sensor) the speed actual value is either determined by a sensor or derived from the calculated rotating field.

Torque Demand (0x6074)

This object returns the resulting target torque after ramp and limiting functions.

Torque demand	Torque setpoint after ramp and limiting functions The unit and direction correspond to object <i>target tor</i>	rque (0x6071).
DS402_0x6074_TORQUE_DEMAND		ID: 0x6074



Torque Actual Value (0x6077)

This object returns the actual torque.

Torque actual va	alue	Torque actual value The unit and direction correspond to object <i>target torque</i> (0x6071		que (0x6071).
DS402_0x6077_TORQUE_ACTUAL_VALUE ID: 0			ID: 0x6077	
RO	Volatile	Unit: ‰	116	

Current Actual Value (0x6078)

This object returns the actual current.

Current actual v	value	Actual current value The value is indicated in parts per thousand (1000 = 100 %) of the rated current, object <i>motor rated current</i> (0x6075).		
DS402_0x6078_CURRENT_ACTUAL_VALUE ID: 0x60		ID: 0x6078		
RO	Volatile	Unit: ‰	116	

DC-Link Circuit Voltage (0x6079)

This object returns the currently applied DC link voltage.

DC link circuit v	oltage	DC link voltage				
DS402_0x6079_	DS402_0x6079_DC_LINK_CIRCUIT_VOLTAGE ID: 0x6079					
RO	Volatile	Unit: mV	U32			

4.3.3 Motor Parameters

Motor Rated Current (0x6075)

This object contains the parameterized motor rated current. The value should match the specification on the motor type plate.

Motor rated current		Rated current of the motor All relative current values refer to this value.		
DS402_0x6075_	ID: 0x6075			
RW	Volatile	Unit: mA	U32	

Motor Rated Torque (0x6076)

This object contains the parameterized motor rated torque. The value should match the specification on the motor type plate.

Motor rated toro	que	Rated torque of the motor All relative torque values refer to this value.		
DS402_0x6076_	MOTOR_RATED_	TORQUE		ID: 0x6076
RW	Volatile	Unit rotary motor: mNm Unit linear motor: Nm	U32	



Motor Type (0x6402)

Via this object you can select the motor type.

Motor type		Selection of the motor type:			
0x0000: Non-standard motor					
	0x0001: Phase modulated DC motor				
		0x0002: Frequency controlled DC n	notor		
		0x0003: PM synchronous motor			
		0x0004: AC synchronous sinewave	wound field		
		0x0005: AC synchronous reluctanc	e switched		
		0x0006: AC asynchronous induction	n polyphase woun	d rotor	
		0x0007: AC asynchronous induction	n squirrel cage		
		0x0008: AC synchronous step			
		0x0009: Micro-step stepper motor			
		0x000A: AC synchronous sinusoidal PM			
		0x000B: AC synchronous brushless	s PM trapezoidal		
		0x000C: AC synchronous reluctanc	e sync		
		0x000D: DC commutator PM			
		0x000E: DC commutator wound fie	ld series		
		0x000F: DC commutator wound fiel	d shunt		
		0x0010: DC commutator wound field compound			
		0x0011 to 0x7FFF: No motor type assigned			
DS402_0X6402	_MOTORTYPE			ID: 0x6402	
RO	Volatile	Unit: increment	U16		

4.3.4 SI Units

SI (French *Système international d'unités*) is the unit system based on the international system of quantities. The following SI unit objects are the basis for the units of the target and actual values in the operating modes (except velocity mode (VL), see <u>chapter "Target and Actual Values in Velocity Mode"</u>, page 33).

At present, the SI unit objects for SD4x drives are read-only objects.

Structure of the SI unit objects

The objects for SI units are 32 bits in site and divided into 4 parts:

	Bit number										
31		24	23		16	15		8	7		0
	Prefix			SI numerator			SI denominator			Reserved(00)	

The documents CiA 890 and CiA 402 describe, which variable values correspond to which Si units or prefixes. A summarizing table with the possible prefixes and SI units of both standard can be found in the Appendix.

Position (0x60A8)

This object contains the position unit.

Position		Position unit			
DS402_0X60A8_SI_UNIT_POSITION ID: 0x60A8					
RW	Volatile	Unit: –	U32		

With SD4x drives, the position is indicated in increments by default. This corresponds to the following variable content:

Byte	3	2	1	0
Value	00	B5	00	00
Content	10 ⁰	Increments	1	-

Therefore, a position setting of 1244 means 1244 increments.



Velocity (0x60A9)

This object contains the speed unit.

Velocity		Velocity unit		
Object: DS402_0X60A9_SI_UNIT_VELOCITY ID: 0x60A9				
RW	Volatile	Unit: –	U32	

With SD4x drives, the speed is indicated in 10^{-3} × rpm by default. The unit rpm stands for *revolution per minute*. This corresponds to the following variable content:

Byte	3	2	1	0
Value	FD	B4	47	00
Content	10 ⁻³	Revolution	Minute	-

Therefore, a speed setting of 2500000 means 2500 rpm.

Acceleration (0x60AA)

This object the unit for acceleration and deceleration.

Acceleration		Acceleration/deceleration unit			
DS402_0X60AA_SI_UNIT_ACCELERATION ID: 0x60AA					
RW	Volatile	Unit: –	U32		

With SD4x drives, the acceleration/deceleration is indicated in $10^{-3} \times 1/s^2$ (*revolutions per square second*) by default. This corresponds to the following variable content:

Byte	3	2	1	0
Value	FD	B4	57	00
Content	10 ⁻³	Revolution	Square second	-

Therefore, an acceleration setting of 300000 means 300 1/s².

Jerk (0x60AB)

This object contains the jerk unit.

Jerk		Jerk unit				
DS402_0X60AB_	_SI_UNIT_JERK			ID: 0x60AB		
RW	Volatile	Unit: –	U32			

With SD4x drives, the jerk is indicated in $10^{-3} \times 1/s^3$ (*revolutions per cubic second*) by default. This corresponds to the following variable content:

Byte	3	2	1	0
Value	FD	B4	A0	00
Content	10 ⁻³	Revolution	Cubic second	_

Therefore, a ruck setting of 2000 means 2 1/s³.

4.3.5 Scaling

In SD4x drives, only the scaling object *polarity* is implemented.



Polarity (0x607E)

Via this object, you can invert the moving direction of the motor.

Polarity Moving direction of the motor				
DS402_0X607E_	POLARITY			ID: 0x607E
RW	Volatile	Unit: –	U8	

If the polarity parameter is not set, a clockwise rotating motor will move in positive direction, an anticlockwise rotating motor will move in negative direction. If the polarity parameter is set, the moving direction is reversed.

The 8-bit object is divided as follows:

Bit number					
7	5		1	0	
Position polarity	Velocity polarity		Reserved		Global polarity

- Bit 7 applies to positioning modes (e.g. profile position mode, interpolated position mode, cyclic sync position mode).
- Bit 6 applies to velocity modes (e.g. profile velocity mode, cyclic sync velocity mode).
- If bit 0 is set too the value 1, the inversion applies to all operating modes. In this case, the values of the bits 6 and 7 are irrelevant and should not be set.

4.3.6 Offsets

Velocity Offset (0x60B1)

This object indicates the offset for target speeds.

Velocity offset		Offset for target speeds The unit is the same as in object <i>target velocity</i> (0x60FF).		
DS402_0X60B1_	DS402_0X60B1_VELOCITY_OFFSET ID: 0x60B1			
RW	Volatile	Unit: –	132	

In the positioning operating modes, this object contains the input value for the velocity feed forward control. In the velocity modes, it contains the velocity offset of the drive. That means, in the velocity modes (except <u>chapter "Velocity Mode (VL)", page 32</u>) the target speed is made up of the *target velocity* and the *velocity offset*.

Torque Offset (0x60B2)

This object indicates the offset for target torque setting.

Torque offset		Offset for target torque The value is indicated in parts per t rated torque, object <i>motor rated to</i>	``	100 %) of the
DS402_0X60B2_TORQUE_OFFSET ID: 0x60B2			ID: 0x60B2	
RW	Volatile	Unit: ‰	116	

In the positioning and velocity operating modes, this object contains the input value for the torque feed forward control. In the velocity modes, it contains the velocity offset of the drive. That means, in den torque modes the target torque is made up of the *target torque* and the *torque offset*.



4.3.7 Limitations

Max Torque (0x6072)

This object contains the maximum permitted value for the torque.

Max torque		Torque limitation for torque profile The value is indicated in parts per thousand (1000 = 100 %) of the rated torque, object <i>motor rated torque</i> (0x6076).		
DS402_0x6072_MAX_TORQUE				ID: 0x6072
RW	Volatile	Unit: ‰	U16	

Max Current (0x6073)

Via this object you can set a limitation for the torque-generating current in the motor.

Max current		Current limitation of the active current The value is indicated in parts per thousand (1000 = 100 %) of the rated current, object <i>motor rated current</i> (0x6075).		
DS402_0x6073_MAX_CURRENT				ID: 0x6073
RW	Volatile	Unit: ‰	U16	

Max Profile Velocity (0x607F)

Via this object you can set the maximum speed.

Max profile velo	city	Speed limit for the motion profile The value applies to both directions. The unit is the same as in object <i>target velocity</i> (0x60FF).		
DS402_0x607F_MAX_PROFILE_VELOCITY ID: 0x607F			ID: 0x607F	
RW	Volatile	Unit: –	U32	-

Max Motor Speed (0x6080)

Via this object you can set the maximum speed for the motor. The value should be taken from the motor data sheet and is used to protect the motor.

Max motor spee	ed	Maximum speed specified by the motor The value applies to both directions. The unit is the same as in object <i>target velocity</i> (0x60FF).			
DS402_0x6080_MAX_MOTOR_SPEED ID: 0x6080			ID: 0x6080		
RW	Volatile	Unit: –	U32		

Positive Torque Limit (0x60E0)

Via this object you can set the maximum positive torque in the motor. The positive torque causes a positive velocity with motor operation and a negative velocity with regenerative operation.

Positive torque	limit	Upper limit for torque setpoint The value is indicated in parts per thousand (1000 = 100 %) of the rated torque, object <i>motor rated torque</i> (0x6076).		100 %) of the
DS402_0X60E0_POSITIVE_TORQUE_LIMIT_VALUE ID			ID: 0x60E0	
RW	Volatile	Unit: ‰	116	



Negative Torque Limit (0x60E1)

Via this object you can set the maximum negative torque in the motor. The negative torque causes a negative velocity with motor operation and a positive velocity with regenerative operation.

Negative torque	limit	Lower limit for torque setpoint The value is indicated in parts per thousand (1000 = 1 rated torque, object <i>motor rated torque</i> (0x6076).		100 %) of the
DS402_0X60E1_NEGATIVE_TORQUE_		QUE_LIMIT_VALUE		ID: 0x60E1
RW	Volatile	Unit: ‰	116	

4.3.7.1 Acceleration and Deceleration Limits

Max Acceleration (0x60C5)

Via this object you can set the maximum permissible acceleration. This is used to limit the acceleration to an acceptable value in order to prevent damage to the motor and moving parts.

Max acceleratio	n	Parameterized maximum acceleration of the system The unit is set in object <i>acceleration</i> (0x60AA).		
DS403_0X60C5_	MAX_ACCELERA	ATION		ID: 0x60C5
RW	Volatile	Unit rotary motor: 1/s² Unit linear motor: mm/s²	U32	

Max Deceleration (0x60C5)

Via this object you can set the maximum permissible deceleration. This is used to limit the deceleration to an acceptable value in order to prevent damage to the motor and moving parts.

Max deceleratio	n		rized maximum deceleration of the system s set in object acceleration (0x60AA).		
DS403_0X60C6_	_MAX_DECELERA	ATION		ID: 0x60C6	
RW	Volatile	Unit rotary motor: 1/s ² U32 Unit linear motor: mm/s ²			

Profile Acceleration (0x6083)

Via this object you can set the acceleration.

Profile acceleration		Acceleration ramp The unit is set in object <i>acceleration</i> (0x60AA).		
DS402_0x6083_	PROFILE_ACCEL	ERATION		ID: 0x6083
RW	Volatile	Unit rotary motor: 1/s ² Unit linear motor: mm/s ²	U32	

Profile Deceleration (0x6084)

Via this object you can set the deceleration.

Profile deceleration		Deceleration ramp The unit is set in object acceleration (0x60AA).		
DS402_0x6084_	PROFILE_DECEL	ERATION		ID: 0x6084
RW	Volatile	Unit rotary motor: 1/s² U32 Unit linear motor: mm/s²		



Quick Stop Deceleration (0x6085)

Via this object you can set the deceleration in the event of a quick stop. In the <u>Option</u> <u>Codes (p. 28)</u> you can configure when a quick stop is triggered.

Quick stop deceleration		Quick stop ramp The unit is set in object acceleration (0x60AA).		
DS402_0x6085_	QUICK_STOP_DE	ECELERATION		ID: 0x6085
RW	Volatile	Unit rotary motor: 1/s ² U32 Unit linear motor: mm/s ²		

Torque Slope (0x6087)

Via this object you can set the rate of change of the torque.

Torque slope		Torque change rate The value is indicated in parts per thousand (1000 = 100 %) of the rated torque, object <i>motor rated torque</i> (0x6076).		
DS402_0x6087_	TORQUE_SLOPE			ID: 0x6087
RW	Volatile	Unit: ‰/s	U32	

4.3.8 Others

Digital Inputs (0x60FD)

This object returns the states of the digital inputs of the SD4x drive. The object is 32-bit variable. If a bit is logical 1, the function assigned to this bit is active.

Digital inputs		States of the digital inputs Bit assignment: Bit 0: Negative limit switch Bit 1: Positive limit switch Bit 2: Home switch Bit 3: Interlock active Bits 4 to 15: Reserved Bits 16 to 31: Inputs 1 to 16 - config	jurable	
DS402_0X60FD_	_DIGITAL_INPUTS	3		ID: 0x60FD
RO	Volatile	Unit: –	U32	

Bit 0 returns the status of the negative limit switch. With SD4x drives, this bit corresponds to the function "(16) Negative limit switch" for digital inputs.

Bit 1 returns the status of the positive limit switch. With SD4x drives, this bit corresponds to the function "(15) Positive limit switch" for digital inputs.

Bit 3 returns the status of the interlock input. If this input signal is deactivated, the SD4x drive must switch to the status "switch on disabled" or "fault reaction active". This disables the power output stage of the drive and prevents is from being reactivated. With SD4x drives, the bit corresponds to the inverted function "(01) Switch on" for digital inputs. Example: If the digital input 2 of the SD4x is parameterized with the function "(01) Switch on" and the physical input 2 is logical 0, then bit 3 in the object *digital inputs* is logical 1 and bit 17 logical 0. If the physical input 2 is logical 1, then bit 3 is be logical 0 and bit 17 logical 1.

Bit 16 to 31 correspond to the inputs of the device. The used number of bits depends on the actual number of physical inputs provided by the SD4x drive.



Digital Outputs (0x60FE)

Via this object you can retrieve as well as set the states of the digital outputs of the SD4x drive.

It provides the following subobjects: status variable *physical outputs* (0x60FE:1) and masking variable *bit mask* (0x60FE:2). Both subobjects are 32-bis variables and contain the bit assignment of the digital outputs. The masking variable allows taking over the control of individual digital outputs from the device. If the masking variable for an output is logical 0, the device controls this output and its current status is returned via the status variable. As soon as the bit in the masking variable is set to logic 1, the status variable controls the corresponding output.

Digital outputs		Access to the digital outputs Bit assignment: Bit 0: Motor holding brake active Bits 1 to 15: Reserved Bits 16 to 31: Outputs 1 to 16 - con Sub-IDs: • 0x60FE:0 = Highest subindex su • 0x60FE:1 = Physical outputs • 0x60FE:2: Bit mask	0	
DS402_0X60FE_	_DIGITAL_OUTPU	ITS		ID: 0x60FE
RW	Volatile	Unit: – BYTEARRAY		

Bit 0 returns the status or controls the output of the motor holding brake. With SD4x drives, this bit corresponds to the function "(05) Motor holding brake" for digital outputs.

Bit 16 to 31 correspond to the outputs of the device. The used number of bits depends on the actual number of physical outputs provided by the SD4x drive.

Highest subindex supported (0x60FE:0)

In its subindex 0, this object contains the last subindex of the object (in this case 2).

Physical outputs (0x60FE:1)

Depending on the masking variable *bit mask* (0x60FE:2) this object provides either the current states of the digital outputs or controls individual outputs. Active bits are logical 1, non-active bits are logical 0.

Bit mask (0x60FE:2)

This object can transfer the control of individual digital outputs from the device to the object *physical outputs* (0x60FE:1). A logical 1 for a bit means that the output is controlled via the object; a logical 0 for a bit means that the status of the output is read.

Example:

- Output 1: parameterized function "(01) Ready for operation"
- Output 2: parameterized function "(03) Switched on"

The status of the physical output 1 is to be observed while the physical output 2 is to return the logical status 1 (active).

To achieve this, set the bits in the object **bit mask** as follows: bit 16 to logical 0 and bit 17 to logical 1. Then, the logical status of output 2 is defined by bit 17 in object **physical outputs**, bit 16 still corresponds to the logical status of the physical output 1. As long as bit 16 in the masking variable is logical 0, the control cannot change the status of output 1.



Error Code (0x603F)

This object returns the error code of the last error that occurred.

Error code		Last saved error code accoring to drive profile DS402. The value 0 means that no error is present in the device.		
DS402_0x603F_	ERROR_CODE			ID: 0x603F
RO	Volatile	Unit: –	U16	

The error codes hardware-dependent. For a detailed description of the error codes refer to the hardware documentation of your device.

4.4 Parameter Set Change

You can create up to 64 parameter sets for one SD4x drive. Switching to another parameter set is only possible, when the output stage is disabled. When the parameter set change was successful, the device is restarted with the selected parameter set.

You can switch the parameter set via the following control sources: *drivemaster4* software, digital inputs or fieldbus. The control source also determines the parameter set that is activated on device startup.

Selection source (0x2037)

This object determines the control source for selecting and switching the parameter set. Internally (within the *drivemaster4* software) you can find this object under index_0x0037.

The following file options are available:

O0: Fixed selection (EEPROM):

This option is the factory setting. The device ist started with the last selected parameter set. You can switch the parameter set via the *drivemaster4* software as well as via fieldbus system.

▶ (01) Digital inputs:

The device ist started with parameter set 0 and switches directly to the parameter set specified by the digital inputs. You can switch the parameter set via the digital inputs only.

(02) Bus system:

The device ist started with parameter set 0. You can switch the parameter set via the fieldbus system only.

The parameter set change via fieldbus is decribed in more detail below. For a description of the parameter set change via *drivemaster4* software or the digital inputs refer to the document "Drive Controller SD4x – Functions and Parameters".

4.4.1 Parameter Set Change via Fieldbus

The parameter set is switched via the object 0x202E (selection).

The following requirements must be met for a successful parameter set change via the bus system:

- 1. The object 0x2037 (selection source) is set to (02) Bus system or (00) Fixed selection (EEPROM).
- 2. The output stage is disabled.



Selection (0x202E)

The object contains the number of the currently selected parameter set. Internally (within the *drivemaster4* software) you can find this object under index_0x002E.

In order to use the object 0x202E for switching between existing parameter sets, only the respective parameter set number must be written into the object.

A parameter set change restarts the device and therefore the fieldbus connection is lost for a short time – i.e. no device control.



5 Diagnosis

5.1 Diagnosis in *drivemaster4*

In the user interface *drivemaster4*, the page "Diagnosis \rightarrow Fieldbus" displays the transmitted telegrams between drive and PLC in the tab "PROFINET IO".

In addition, the IP configuration as well as diagnostic data are displayed on the right side of the page.

5.1.1 Telegrams

The following figure shows the logged telegrams on the PROFINET IO diagnosis page.

ame	Counter	Time stamp	Msgs/s	Dir	Cmd	Length	Status
+ RPDO 0	Counter	mile stamp	141393/3	UII	Cind	Length	Status
	aa 00 00 00 00						
Controlword	uu 00 00 00 00	6	0	006h			
Target veloc:	ity	43737	0000a	ad9h			
Max torque		0	0	000h			
➡ TPDO 0	160541	229864.74	1100	Tx		12	
✓ Data : 40 92 00	00 00 00 00 00 00 00	00 00					
Statusword		37440	9	240h			
Velocity act	ual value	0	00000	000h			
Position act	ual value	0	00000				
Error code		0	0	000h			
<table-cell-rows> Unsupported</table-cell-rows>	5	10950.49	0	Rx	lf2eh	8	0h
 Unsupported 	5	10950.50	0	Tx	lf2fh	0	c0000202h
<table-cell-rows> Link_Status_Char</table-cell-rows>	nge 2	4953.46	0	Rx	2f8ah	32	0h
 Link_Status_Char 	nge 2	4953.54	0	Tx	2f8bh	0	0h
<table-cell-rows> Parameter_End</table-cell-rows>	1	10938.50	0	Rx	lf0eh	12	0h
 Parameter_End 	1	10938.52	0	Tx	lf0fh	8	0h
<table-cell-rows> Save_IP_Addr</table-cell-rows>	1	7926.53	0	Rx	lfb8h	13	0h
➡ Save_IP_Addr	1	7926.59	0	Тх	lfb9h	0	0h
Clear							From control 🜩 To co

Fig. 30: PROFINET IO bus activities

Station name

The top line indicates the PROFINET IO station name. The station name is stored in the following object:

Object 155 (0x009B) Station name

The station name consists of the following bytes:

- length indication of the current station name: 2 byte
- station name in ASCII format: max. 240 byte

Example:

- Data length of station name: 4 byte
- Station name: sd4x



Representation in the object browser of the drivemaster4 software:

Object PROFINET IO - Station name									
	00	01	02	03	04	05	06	07	08
0000	04	00	73	64	34	78	00	00	00

Fig. 31: Station name

Output window

You can unfold the lines of the cyclic telegrams RPDO x and TPDO x to see their individual parameterized objects.

The acyclic telegrams are displayed in pairs as request and response telegram. User data are not displayed because more than 1000 bytes could be transmitted here and therefore the readability would be lost.

The following information is displayed in the output window for the telegrams:

Name	telegram name			
Counter	counter of telegram frequency			
Timestamp	drive-internal time stamp			
Msgs/s	number of sent telegrams per second			
Dir	telegram direction from the point of view of the drive:			
	 Rx: telegram was received by the drive. 			
	 Tx: telegram was sent by the drive. 			
Additional information for acyclic telegrams:				
cmd	command number of acyclic telegram			
Length	length of telegram in bytes			

Status error status for the execution of the command

Clear

You can clear the output window by click on the button "Clear". Afterwards, logging the detected messages is restarted.

Communication state

This field displays the communication status of the bus system. The following states are possible: Unknown, Offline, Stop, Idle and Operate.



5.1.2 Diagnostic data

The following figure shows the IP configuration for the Ethernet communication and the diagnostic data on the right side of the PROFINET IO diagnosis page. The indicated values are typical values for the status OPERATE.

IP Configuration	1
IP address	192.168.0.1
Network mask	255.255.0.0
Gateway address	192.168.0.1
Diagnostic data	
PNS state	0x00000fb
Last error	0x0000000
Link state	1
Config state	3
Communication state	4
Communication error	0x0000000
Line delay port 1	7 ns
Line delay port 2	0 ns

Fig. 32: IP configuration and diagnostic data

The data are stored in the following objects.

Object 154 (0x009A) diagnostic data

The diagnostic data consist of 8 values with 4 bytes each:

Variable	able Value Description		Example value
PNS state	Bit	Status of the protocol stack in NetX90	0x00000FB
	0	Device information is set	1
	1	PROFINET stack is started	1
	2	Reserved	0
	3	At least one API is present	1
	4	Module 0 is available	1
	5	Module 0 and submodule 1 are available	1
	6	Network communication is permitted	1
	7	Network communication is activated	1
	8	Configuration is locked	0
	9	Fatal error occurred	0
	10	Diagnostic data with fatal error available	0
	11	Data set requiring maintenance exists	0
	12	Maintenance request data set exists	0
Last error		Last error is displayed	0x0000000
Link state	0	No information available	0x0000001
	1	Physical link works correctly	
	2	Physical link works with low speed	
	3	No physical link available	



Variable	ble Value Description		Example value
Config state	0	Not configured	0x0000003
	1	Configured via DBM files]
	2	Error during configuration via DBM files	
	3	Configured via application	
	4	Configuration via application in progress	
	5	Error during configuration via application]
	6	Configured via warm boot parameters]
	7	Configuration via warm boot parameters in progress]
	8	Error during configuration via warm boot parameters	
Communication	0	UNKNOWN (black)	0x0000004
state	1	OFFLINE (gray)	
	2	STOP (red)	
	3	IDLE (yellow)]
	4	OPERATE (green)]
Communication er- ror		Error status in communication channel is displayed	0x0000000
Line delay port 1		Signal delay [ns] for port 1 is displayed	0x0000007
Line delay port 2		Signal delay [ns] for port 2 is displayed	0x0000000

Representation in the object browser of the *drivemaster4* software:

Object PROFINET IO - Diagnostic data																
	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
0000	fb	00	00	00	00	00	00	00	01	00	00	00	03	00	00	00
0010	04	00	00	00	00	00	00	00	07	00	00	00	00	00	00	00

Fig. 33: Diagnostic data

Object 156 (0x009C) IP address

The IP address consist of 3 values with 4 bytes each:

Description	Example value				
IP address	C0:A8:00:01				
Network mask	FF:FF:00:00				
Gateway	C0:A8:00:01				

Representation in the object browser of the *drivemaster4* software:

Object PROFINET IO - IP address												
	00	01	02	03	04	05	06	07	08	09	0a	0b
0000	01	00	a8	c0	00	00	ff	ff	01	00	a8	c 0

Fig. 34: IP address

5.2 Diagnosis in PLC

No connection with drive

If the connection to a device cannot be established, check that the station name, the IP address and the MAC address match the data in the drive. If not, set the desired data in the *drivemaster4* software and write them to the drive.



6 Reference

Other documents

The following documents provide more information on this topic:

Supplier	Document
SIEB & MEYER AG	 drivemaster4 - User Manual Drive System SD4 – Ethernet Configuration
PROFIBUS Nutzerorganisation e.V. (P- NO)	As registered user you can download documents for PROFINET from the website of the organization.
CiA e.V.	 CiA 301, version 4.2.0. CANopen application layer and communication profile CiA 402, version 5.0.0. CANopen device profile drives and motion control CiA 890, version 5.0.0. Presentation of SI units and prefixes

Websites

The following websites provide more information on this topic:

Supplier	Document
SIEB & MEYER AG	www.sieb-meyer.com
PROFIBUS Nutzerorganisation e.V. (P-NO)	www.profibus.com
CiA e.V.	www.can-cia.org
Siemens AG	www.siemens.com



Index

A

Abbreviations <u>5</u> Acyclic service data <u>17</u> program example <u>19</u>

С

Communication status <u>15</u> Control commands of drive <u>26</u> Controlword <u>26</u>

D

Diagnosis drivemaster4 <u>52</u> PLC <u>55</u>

Ν

Network <u>8</u> NMT status <u>15</u>

0

Objects abort connection option code 28 acceleration 44 controlword 26 current actual value 42 DC link circuit voltage 42 digital inputs 48 digital outputs 49 disable operation option code 29 error code 50 fault reaction option code 29 halt option code 29 jerk 44 max acceleration 47 max current 46 max deceleration 47 max motor speed 46 max profile velocity 46

max torque 46 mode of operation 39 mode of operation display 40 motor rated current 42 motor rated torque 42 motor type <u>43</u> negative torque limit 47 polarity 45 position 43 position actual value 41 positive torque limit 46 profile acceleration 47 profile deceleration 47 quick stop deceleration 48 quick stop option code 28 shutdown option code 29 statusword 27 supported drive modes 39 target torque 41 target velocity 40 torque actual value 42 torque demand 41 torque offset 45 torque slope 48 velocity 44 velocity actual value 41 velocity demand value 41 velocity offset 45 vl dimension factor 37 vl set-point factor 36 vl target velocity 33 vl velocity acceleration 34 vl velocity actual value 33 vl velocity deceleration 35 vl velocity demand 33 vl velocity min max amount 33 vl velocity quick stop 35 Operating mode change 38 Operating modes 30



Option codes 28

Ρ

Parameter set change <u>50</u> PLC connection <u>8</u> Profile torque mode <u>38</u> Profile velocity mode <u>31</u>

S

State machine <u>25</u> State transitions of the drive <u>25</u> States <u>27</u> Statusword <u>27</u>

Т

TIA Portal <u>8</u> Torque mode <u>38</u>

V

Velocity mode <u>32</u> Velocity mode PV <u>31</u> VL <u>32</u>