

Drive Controller SD4x

POWERLINK Connection

A getting started guide for users





Copyright

Original instructions, Copyright © 2025 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	About this Manual.....	5
1.1	Abbreviations.....	5
2	Drive Control (CiA 402).....	7
2.1	Service Data Objects (SDO).....	7
2.2	Process Data Objects (PDO).....	8
2.3	State Machine.....	9
2.3.1	Commands.....	10
	Controlword (0x6040).....	10
2.3.2	States.....	11
	Statusword (0x6041).....	11
2.3.3	Option Codes.....	12
	Abort Connection Option Code (0x6007).....	12
	Quick Stop Option Code (0x605A).....	12
	Shutdown Option Code (0x605B).....	13
	Disable Operation Option Code (0x605C).....	13
	Halt Option Code (0x605D).....	13
	Fault Reaction Option Code (0x605E).....	13
2.4	Operating Modes.....	14
2.4.1	Cyclic Synchronous Velocity Mode (CSV).....	14
2.4.2	Profile Velocity Mode (PV).....	16
2.4.3	Velocity Mode (VL).....	16
2.4.3.1	Target and Actual Values in Velocity Mode.....	17
	vl Target Velocity (0x6042).....	17
	vl Velocity Demand (0x6043).....	18
	vl Velocity Actual Value (0x6044).....	18
	vl Velocity Min Max Amount (0x6046).....	18
	vl Velocity Acceleration (0x6048).....	19
	vl Velocity Deceleration (0x6049).....	19
	vl Velocity Quick Stop (0x604A).....	20
	vl Set-point Factor (0x604B).....	21
	vl Dimension Factor (0x604C).....	21
2.4.4	Torque Mode.....	22
2.4.5	Change Operating Mode.....	23
	Supported Drive Modes (0x6502).....	23
	Mode of Operation (0x6060).....	24
	Mode of Operation Display (0x6061).....	24
2.5	Target and Actual Values.....	24
2.5.1	General Target Values.....	25
	Target Velocity (0x60FF).....	25
	Target Torque (0x6071).....	25
	Interpolation Time Period (0x60C2).....	25
	Motion Profile Type (0x6086).....	26
2.5.2	Actual Values.....	26
	Position Actual Value (0x6064).....	26
	Velocity demand value (0x606B).....	27
	Velocity Actual Value (0x606C).....	27
	Torque Demand (0x6074).....	27
	Torque Actual Value (0x6077).....	27
	Current Actual Value (0x6078).....	27
	DC-Link Circuit Voltage (0x6079).....	27
2.5.3	Motor Parameters.....	28
	Motor Rated Current (0x6075).....	28
	Motor Rated Torque (0x6076).....	28
	Motor Type (0x6402).....	28
2.5.4	SI Units.....	28
	Position (0x60A8).....	29
	Velocity (0x60A9).....	29
	Acceleration (0x60AA).....	29
	Jerk (0x60AB).....	30
2.5.5	Scaling.....	30
	Polarity (0x607E).....	30
2.5.6	Offsets.....	30
	Velocity Offset (0x60B1).....	30
	Torque Offset (0x60B2).....	31



2.5.7	Limitations.....	31
	Max Torque (0x6072).....	31
	Max Current (0x6073).....	31
	Max Profile Velocity (0x607F).....	31
	Max Motor Speed (0x6080).....	32
	Positive Torque Limit (0x60E0).....	32
	Negative Torque Limit (0x60E1).....	32
2.5.7.7	Acceleration and Deceleration Limits.....	32
	Max Acceleration (0x60C5).....	32
	Max Deceleration (0x60C6).....	32
	Profile Acceleration (0x6083).....	33
	Profile Deceleration (0x6084).....	33
	Quick Stop Deceleration (0x6085).....	33
	Torque Slope (0x6087).....	33
2.5.8	Others.....	34
	Digital Inputs (0x60FD).....	34
	Digital Outputs (0x60FE).....	34
	Error Code (0x603F).....	35
2.6	Parameter Set Change.....	35
2.6.1	Parameter Set Change via Fieldbus.....	36
3	Control via POWERLINK.....	37
3.1	POWERLINK Network.....	37
3.2	Software Connection to the PLC.....	37
3.2.1	Create Project.....	38
3.2.2	Create Device Structure.....	42
3.2.3	Create ST Program.....	43
3.3	POWERLINK Network Management (NMT).....	44
4	Drive Parameterization and Diagnostics.....	46
4.1	Parameterization in drivemaster4.....	46
4.1.1	Address Offset for Node ID.....	47
4.2	Diagnosis in drivemaster4.....	49
4.2.1	Telegrams.....	49
4.2.2	Error Status.....	50
5	Reference.....	51
	Index.....	52

1 About this Manual

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

A basic description of the POWERLINK network can be found on the website of the company B&R Industrial Automation GmbH (www.br-automation.com).

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 POWERLINK 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 [section “Other documents”, page 51](#).

1.1 Abbreviations

ASnd	A synchronous S end (POWERLINK frame)
CAN	C ontroller A rea N etwork
CiA	C AN i n A utomation
CoE	C ANopen o ver E therCAT
CSV	c yclic s ynchronous v elocity mode
DC	D istributed C locks
MDevice	M ain D evice (formerly master)
NMT	n etwork m anagement
PDO	p rocess d ata o bject
PLC	p rogrammable l ogic c ontrol
PReq	P oll R equ e st (POWERLINK frame)
PRes	P oll R es p onse (POWERLINK frame)
PV	p rofile v elocity mode
RO	r ead o nly (object access)
RW	r ead w rite (object access)
Rx	receive
SDO	S ervice d ata o bject
SoA	S tart o f A synchronous (POWERLINK frame)
SoC	S tart o f C ycle (POWERLINK frame)
SubDevice	S ubordinate D evice (formerly slave)



Tx transmit
VL velocity mode
XDD XML device description file

2 Drive Control (CiA 402)

The drive controllers of the device family SD4x support the drive profile CiA 402. That means, the drive control described in this chapter is implemented according to the standards DS301 and DS402. The associated drive objects have the prefix “DS301” or “DS402”.

The drive profile CiA 402 defines objects, which are classified as [process data object \(PDO\)](#) 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 [controlword](#) and returns the current status via the [statusword](#). By means of the [option codes](#) 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 ([cyclic synchronous velocity mode](#), [profile velocity mode](#), [velocity mode](#) or [torque mode](#)). How to switch the operating mode is described under [change operating mode](#).

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 index
Access ⁽¹⁾	Data memory ⁽²⁾	Unit	Data type

⁽¹⁾ Possible values:

RW = read/write

RO = read only

⁽²⁾ Possible Values:

Volatile = data is lost after power cut-off

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

2.1 Service Data Objects (SDO)

Service data objects are exchanged acyclically. They are used for data that is exchanged infrequently and not in real time. During the boot phase of the fieldbus, for example, the fieldbus control (MDevice) and the fieldbus participant (SubDevice) negotiate the PDO mapping via SDOs.

The MDevice triggers each SDO access by sending a request telegram to the SubDevice. Since the request is acyclic, it may take several fieldbus cycles until the SubDevice responds.

The fieldbus control transmits SDO telegrams with subordinate priority. That means, if the total fieldbus capacity is used up by PDO telegrams, SDO communication is not possible anymore.

An SDO request includes 2 addresses:

- ▶ SubDevice address (e.g. ADS address or ID)
- ▶ object address (e.g. object index and object subindex)

One request can address individual objects or entire object blocks. The exact structure and sequence of an SDO request depends on the used fieldbus system and can vary.



2.2 Process Data Objects (PDO)

Process data objects transmit the user data that are exchanged cyclically and in real time.

During the boot phase of the fieldbus, the fieldbus control (MDevice) checks and negotiates, which process data are to be exchanged. This depends on whether a fieldbus participant (SubDevice) supports predefined or configurable PDOs. SD4x drives support configurable PDOs.

Not all data objects of a SubDevice can be processed in real time. Thus, not all data can be exchanged via PDOs. Usually, the fieldbus-specific description files (e.g. ESI, XML or EDS file) specify which data is to be exchanged via PDOs.

Note

Modern fieldbus systems come with sufficient bandwidth to transmit a lot of user data with high speeds. However, we still recommend to exchange data without real-time requirements via service data objects.

2.3 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.

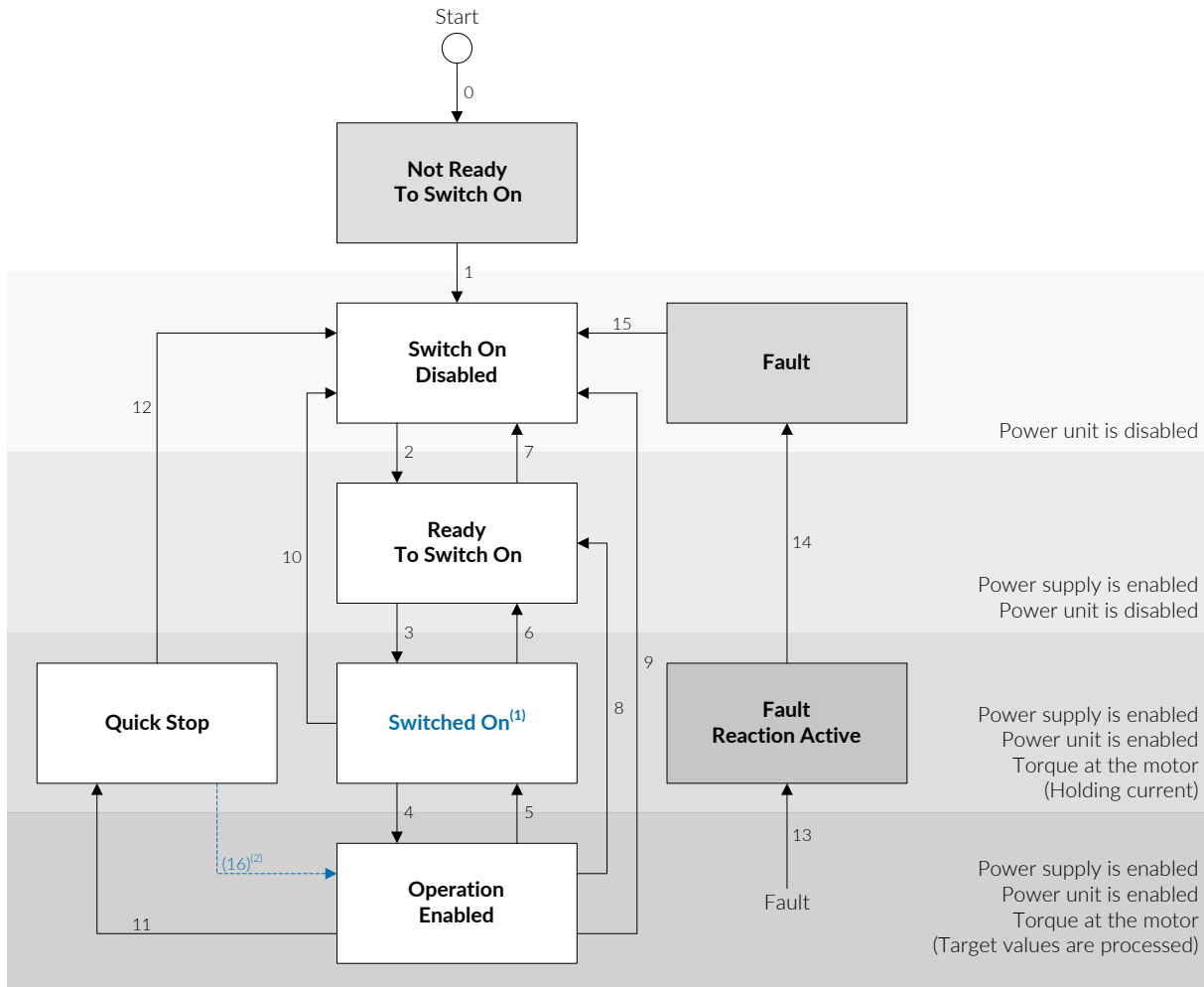


Fig. 1: Drive states and state transitions

- (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 [chapter 2.3.1 “Commands”, page 10](#). 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.



2.3.1 Commands

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

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

⁽¹⁾ 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.

2.3.2 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:

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

⁽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: <i>Drive setup tool (DRS) active*</i> Bit 9: Remote Bit 10: Target reached Bit 11: <i>Current limit reached*</i> Bit 12: Mode-specific 1 Bit 13: Mode-specific 2 Bit 14: <i>Safe Torque Off (STO)*</i> Bit 15: <i>Initialization finished*</i> <i>*Bit assignments shown in italics are SIEB & MEYER specific and may differ with other drive manufacturers.</i>		
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, the halt bit in the controlword and the current status:

- ▶ Operating modes: bit 10 is set as soon as the target value for speed, position or torque is reached.
Example for velocity modes (CSV/PV/VL): The target value is reached 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]”.
- ▶ 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.
- ▶ Halt bit in controlword (bit 8) is active: bit 10 is set as soon as the drive comes to standstill.
- ▶ Changing the operating mode: bit 10 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.

2.3.3 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 connection option code		Selection of the reaction to a fieldbus communication error: 00: No action 01: Fault signal 02: Disable voltage command 03: Quick stop command	
Object: DS402_0X6007_ABORT_CONNECTION_OPTION_CODE			ID: 0x6007
RW	Volatile	Unit: –	l16

Quick Stop Option Code (0x605A)

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

Quick stop option 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_QUICK_STOP_OPTION_CODE			ID: 0x605A

RW	Volatile	Unit: –	l16
----	----------	---------	-----

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 option code	Selection of the reaction to the command “shutdown”: –1: Slow down via short-circuiting the motor phases 00: Disable drive function (switch off the drive power output stage) 01: Slow down on slow down ramp, disable drive function		
DS402_0X605B_SHUTDOWN_OPTION_CODE			ID: 0x605B
RW	Volatile	Unit: –	l16

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 option code	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: 0x605C
RW	Volatile	Unit: –	l16

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	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: –	l16

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 option code	Selection of the reaction to a fault: 00: Disable drive function, motor is free to rotate 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_0X605E_FAULT_OPTION_CODE			ID: 0x605E
RW	Volatile	Unit: –	l16

The ramps are defined in the operating mode.

2.4 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.

2.4.1 Cyclic Synchronous Velocity Mode (CSV)

The cyclic synchronous velocity mode (CSV mode) enables an external trajectory calculation or position control by a higher-ranking control. Then, the drive follows the target values of the control. For smooth interpolation of the target values, the drive and control must be synchronized. In addition, the **interpolation time period** should match the fieldbus cycle time.

In order to control the drive in the CSV 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.

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

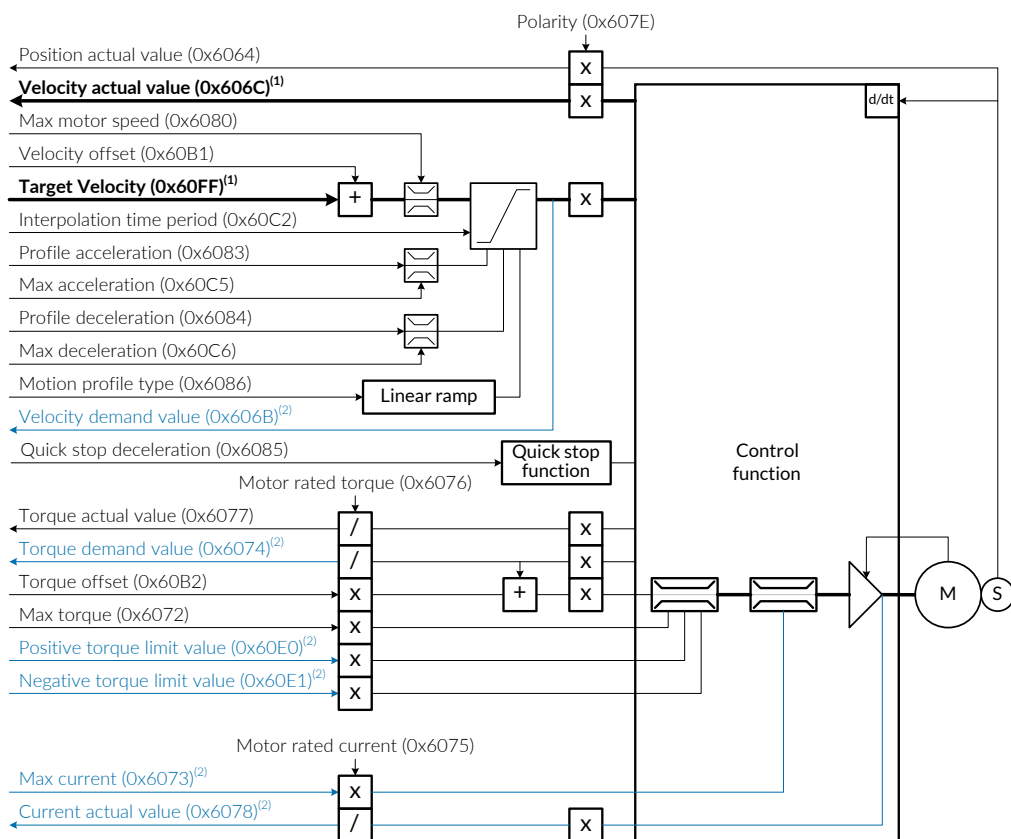


Fig. 2: Block diagram for CSV mode

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

The objects **quick stop deceleration** or **profile deceleration** are used when an option code defines a stop via quick stop ramp or deceleration ramp.

The object **torque offset** can be used for the torque feed forward control. In addition, the set torque, current and acceleration limits remain active.

The objects *interpolation time period* and *target velocity* automatically determine the change in speed until the next interpolated position of a trajectory calculation is reached.

CSV-specific bits in the statusword

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

- ▶ Bit 12 (drive follows target value) is active as long as the drive follows the trajectory from the control.

2.4.2 Profile Velocity Mode (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:

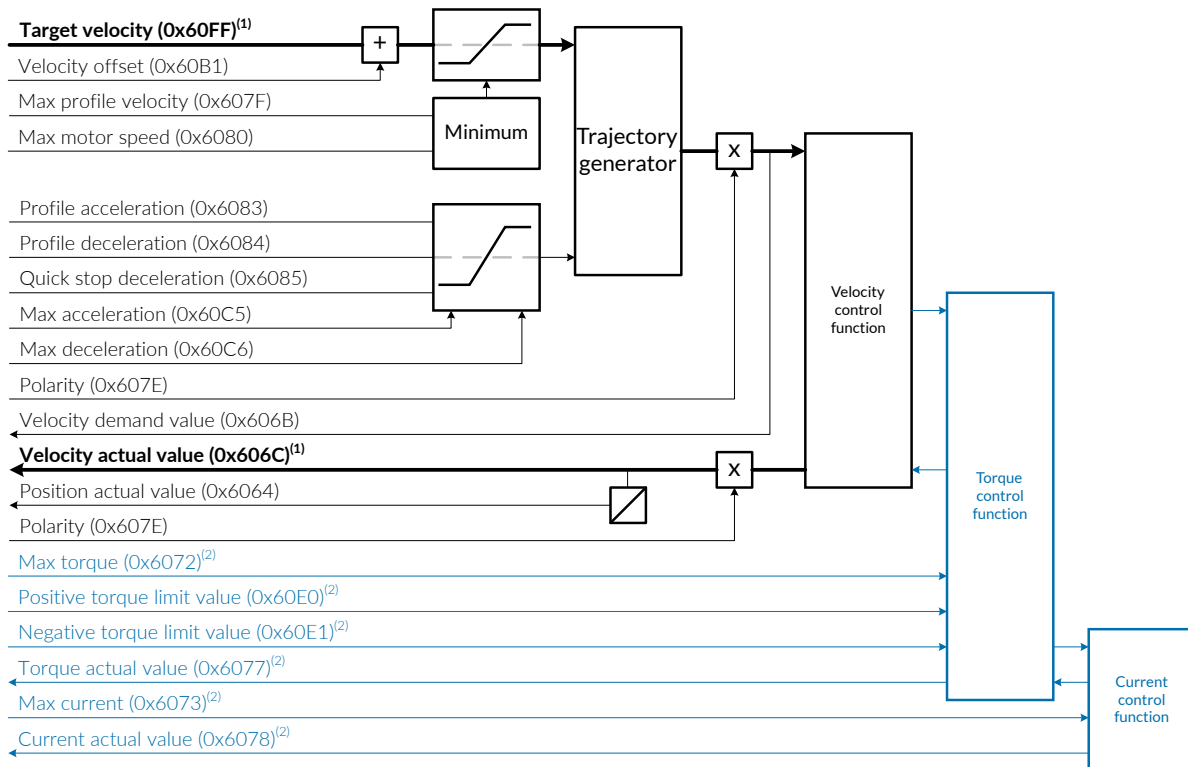


Fig. 3: Block diagram for profile velocity mode (PV)

- (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 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 → Messages → Speed zero [M15]”.
- ▶ Bit 13 (maximum slip) is active when the drive has reached the parameterized slip.

2.4.3 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 **vl dimension factor**. In addition, the object **vl 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 [Profile Velocity Mode \(PV\)](#), [page 16](#), 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 **vl target velocity**. It is recommended to use the **vl velocity actual value** as return value.

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

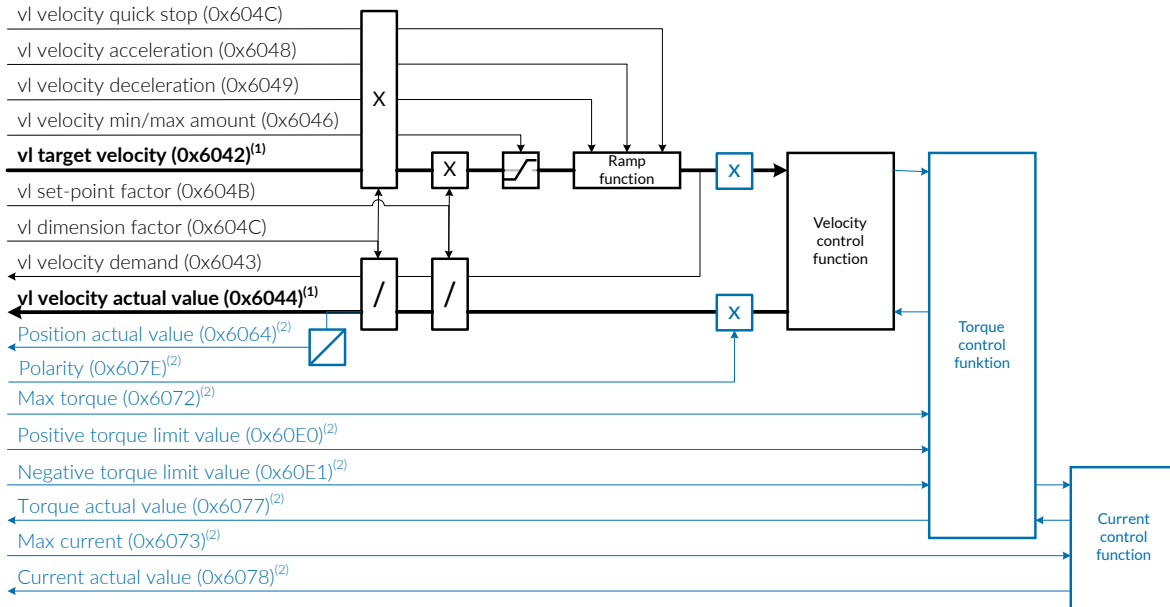


Fig. 4: Block diagram for velocity mode VL

- (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 bit 12 as follows:

- ▶ 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 → Messages → Speed zero [M15]”.

2.4.3.1 Target and Actual Values in Velocity Mode

vl Target Velocity (0x6042)

This object contains the target speed in velocity mode.

vl target velocity	Target speed in velocity mode When the objects vl set-point factor (0x604B) and vl dimension factor (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_VELOCITY	ID: 0x6042

RW	Volatile	Unit: rpm (default)	l16
----	----------	---------------------	-----

vl Velocity Demand (0x6043)

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

vl velocity demand	Speed setpoint in velocity mode The value indicates the speed setpoint after the ramp and limiting functions. The unit and direction correspond to object vl target velocity (0x6042).		
DS402_0x6043_VL_VELOCITY_DEMAND			ID: 0x6043
RO	Volatile	Unit: rpm (default)	l16

vl Velocity Actual Value (0x6044)

This object returns the speed actual value in velocity mode.

vl velocity actual value	Actual speed in velocity mode The unit and direction correspond to object vl target velocity (0x6042).		
DS402_0x6044_VL_VELOCITY_ACTUAL_VALUE			ID: 0x6044
RO	Volatile	Unit: rpm (default)	l16

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 max amount	Speed limitation in velocity mode Sub-IDs: <ul style="list-style-type: none"> ▶ 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 (default), this object indicates the speed limits in rpm.		
DS402_0x6046_VL_VELOCITY_MIN_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 **vl velocity min max amount**. The dotted line shows the speed setpoint and the blue, continuous line shows the resulting target speed.

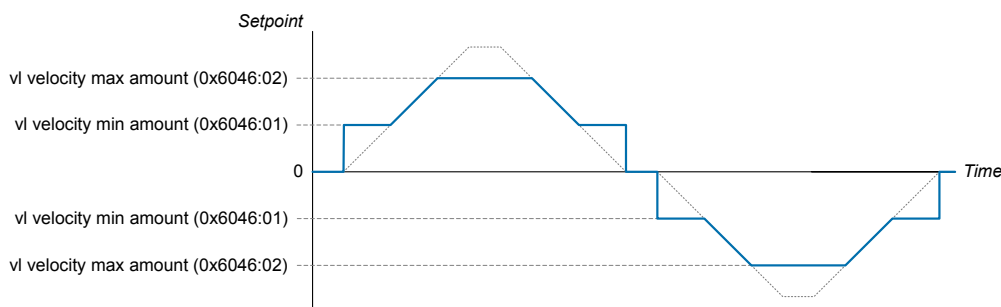


Fig. 5: Speed limitation in velocity mode

vl Velocity Acceleration (0x6048)

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

vl velocity acceleration		Acceleration ramp in velocity mode Sub-IDs: <ul style="list-style-type: none"> ▶ 0x6048:00 = Highest subindex supported (2) ▶ 0x6048:01 = Delta speed If the object vl dimension factor (0x604C) is set to the value 1 (default), this object indicates the speed in rpm. ▶ 0x6048:02 = Delta time 	
DS402_0x6048_VL_VELOCITY_ACCELERATION		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.

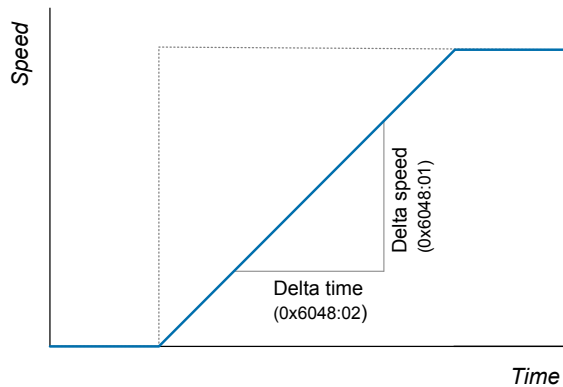


Fig. 6: Limitation of acceleration in velocity mode

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 deceleration		Deceleration ramp in velocity mode Sub-IDs: <ul style="list-style-type: none"> ▶ 0x6049:00 = Highest subindex supported (2) ▶ 0x6049:01 = Delta speed If the object vl dimension factor (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_DECELERATION		ID: 0x6049	
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 deceleration via the object **vl velocity deceleration**. The dotted line shows the speed setpoint and the blue, continuous line shows the resulting target speed.

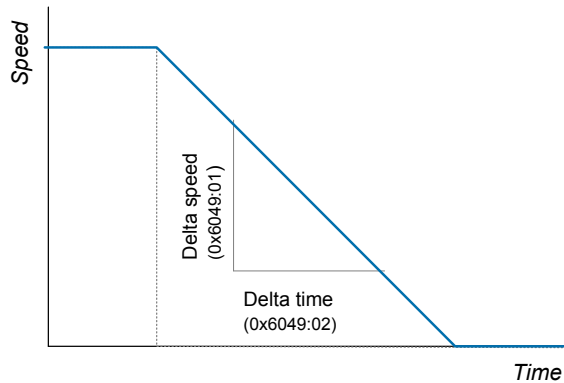


Fig. 7: Limitation of deceleration in velocity mode

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.

vl velocity quick stop		Quick stop ramp in velocity mode Sub-IDs: <ul style="list-style-type: none"> ▶ 0x604A:00 = Highest subindex supported (2) ▶ 0x604A:01 = Delta speed If the object vl dimension factor (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_QUICK_STOP		ID: 0x604A	
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 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.

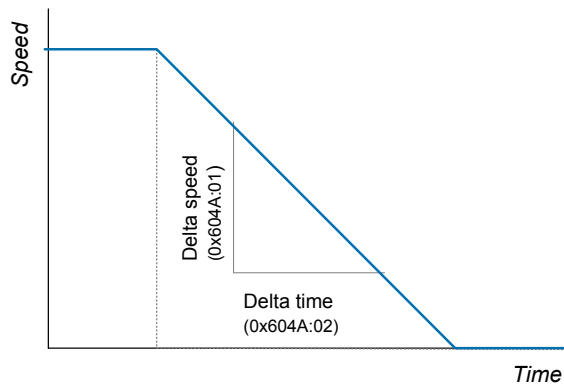


Fig. 8: Limitation of deceleration in the event of a quick stop in velocity mode

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.

vl set-point factor		Scaling factor for the setpoint and the actual value in velocity mode Sub-IDs: <ul style="list-style-type: none"> ▶ 0x604B:00 = Highest subindex supported (2) ▶ 0x604B:01 = Numerator The numerator has no unit. The value 0 is not permitted, in this case the last valid value is retained. ▶ 0x604B:02 = Denominator The denominator has no unit. The value 0 is not permitted, in this case the last valid value is retained. 	
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.

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

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 factor		Scaling factor for the speed unit in velocity mode Sub-IDs: <ul style="list-style-type: none"> ▶ 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_ACCERLERATION_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).

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

When the scaling factor is changed, the ramp and limiting functions are automatically scaled to the new unit. This applies to the following objects: vl velocity min amount (0x6046:01), vl velocity max amount (0x6046:02), vl velocity acceleration (0x6048:01), vl velocity deceleration (0x6049:01) and vl 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.

2.4.4 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:

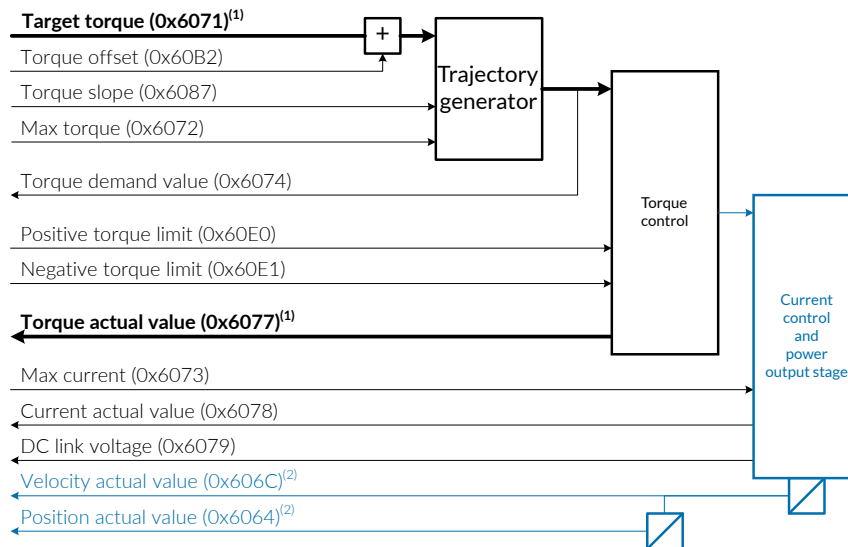


Fig. 9: Block diagram for torque mode

- (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.

2.4.5 Change Operating Mode

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

Note

As an alternative, SD4x drives allow switching the complete parameter set (see [chapter 2.6 “Parameter Set Change”, page 35](#)). You should consider this option when several parameters must be changed in addition to the operating mode.

⚠ WARNING



Risk of injury due to unintentional motor start

→ 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 display** (0x6061).

After the operating mode was successfully changed, bit 10 (target reached) in the **statusword** (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 modes	<p>Available operating modes:</p> <ul style="list-style-type: none"> 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 mode with commutation angle Bit 11: ppaf – profile position mode with additional feedback Bit 12: pvaf – profile velocity mode with additional feedback Bit 13: cspaf – cyclic sync position mode with additional feedback Bit 14: csvaf – cyclic sync velocity mode with additional feedback Bit 15: r – reserved
------------------------------	---



DS402_0X6502_SUPPORTED_DRIVE_MODES			ID: 0x6502
RO	Volatile	Unit: –	U32

Mode of Operation (0x6060)

Via this object you can change the operating mode.

Mode of operation	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 commutation angle 12: Profile position mode with additional feedback 13: Profile velocity mode with additional feedback 14: Homing mode with additional feedback 15: Cyclic sync position mode with additional feedback 16: Cyclic sync velocity mode with additional feedback		
DS402_0x6060_MODE_OF_OPERATION			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 mode 08: Cyclic sync position mode 09: Cyclic sync velocity mode 10: Cyclic sync torque mode 11: Cyclic sync torque mode with commutation angle 12: Profile position mode with additional feedback 13: Profile velocity mode with additional feedback 14: Homing mode with additional feedback 15: Cyclic sync position mode with additional feedback 16: Cyclic sync velocity mode with additional feedback		
DS402_0X6061_MODE_OF_OPERATION_DISPLAY			ID: 0x6061
RO	Volatile	Unit: –	U8

2.5 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 (v1) uses mostly its own target, actual, scaling and limiting values. These are directly described below the operating mode, see [page 17](#).

2.5.1 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
RW	Volatile	Unit: –	I32

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 motor rated torque (0x6076).	
DS402_0x6071_TARGET_TORQUE			ID: 0x6071
RW	Volatile	Unit: ‰	I16

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).

Interpolation Time Period (0x60C2)

This object defines the time period between two updates of the target value.

Interpolation time period		Time period for interpolation Sub-IDs: <ul style="list-style-type: none"> ▶ 0x60C2:00 [U8] = Highest subindex supported (2) ▶ 0x60C2:01 [U8] = Interpolation time period value (1) ▶ 0x60C2:02 [I8] = Interpolation time index (-3) 	
DS402_0x60C2_INTERPOLATION_TIME_PERIOD			ID: 0x60C2
RW	Volatile	Unit: –	BYTEARRAY

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

The value for the interpolation time period <subindex1> is indicated in the unit 10 to the power of <subindex2> seconds (default = 1×10^{-3} s = 1 ms). If the resulting time T_{itp} is shorter than the bus cycle time, the target value is reached at the time of T_{itp} (if not otherwise limited). Operating modes with target value buffer: after the interpolation time has elapsed, the next value from the buffer is used.

For cyclic operating modes, you should set this object to the current fieldbus cycle time. Otherwise, the interpolation time deviates from the fieldbus cycle time, which causes target value jumps. If interpolation is finished before a new target value is available via fieldbus, the final value of the interpolation is maintained until the next update (see figure to the right).

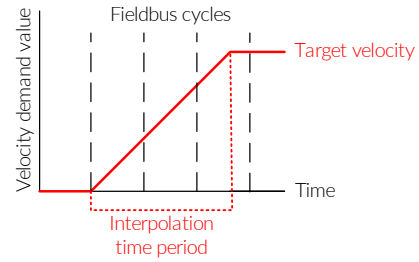


Fig. 10: Interpolation ≠ fieldbus cycle time

The following figures show examples of the interpolation (velocity demand value in blue) of a target value (in red):

Interpolation time period is set correctly:

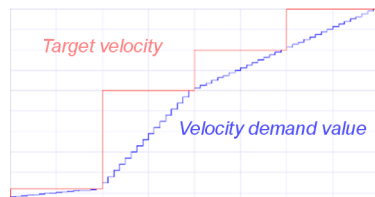


Fig. 11: No target value jumps

Interpolation time period is set incorrectly:

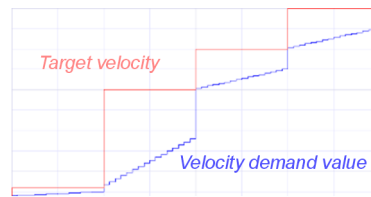


Fig. 12: Target value jumps

Motion Profile Type (0x6086)

This object specifies the shapes of the acceleration and deceleration ramps.

Motion profile type		Motion profile types: 0: Linear ramp 1: Sin ² ramp 2: Jerk-free ramp 3: Jerk-limited ramp	
DS402_0X6086_MOTION_PROFILE_TYPE			ID: 0x6086
RW	Volatile	Unit: –	I16

At present, SD4x drives support the linear ramp shape only. Therefore, this object has the fix value 0.

2.5.2 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
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 demand value		Speed setpoint after ramp and limiting functions The unit and direction correspond to object target velocity (0x60FF).	
DS402_0x606B_VELOCITY_DEMAND_VALUE			ID: 0x606B
RO	Volatile	Unit: –	I32

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 target velocity (0x60FF).	
DS402_0x606C_VELOCITY_ACTUAL_VALUE			ID: 0x606C
RO	Volatile	Unit: –	I32

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 target torque (0x6071).	
DS402_0x6074_TORQUE_DEMAND			ID: 0x6074
RO	Volatile	Unit: ‰	I16

Torque Actual Value (0x6077)

This object returns the actual torque.

Torque actual value		Torque actual value The unit and direction correspond to object target torque (0x6071).	
DS402_0x6077_TORQUE_ACTUAL_VALUE			ID: 0x6077
RO	Volatile	Unit: ‰	I16

Current Actual Value (0x6078)

This object returns the actual current.

Current actual value		Actual current value The value is indicated in parts per thousand (1000 = 100 %) of the rated current, object motor rated current (0x6075).	
DS402_0x6078_CURRENT_ACTUAL_VALUE			ID: 0x6078
RO	Volatile	Unit: ‰	I16

DC-Link Circuit Voltage (0x6079)

This object returns the currently applied DC link voltage.

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

2.5.3 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_MOTOR_RATED_CURRENT			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 torque		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 motor 0x0003: PM synchronous motor 0x0004: AC synchronous sinewave wound field 0x0005: AC synchronous reluctance switched 0x0006: AC asynchronous induction polyphase wound rotor 0x0007: AC asynchronous induction squirrel cage 0x0008: AC synchronous step 0x0009: Micro-step stepper motor 0x000A: AC synchronous sinusoidal PM 0x000B: AC synchronous brushless PM trapezoidal 0x000C: AC synchronous reluctance sync 0x000D: DC commutator PM 0x000E: DC commutator wound field series 0x000F: DC commutator wound field shunt 0x0010: DC commutator wound field compound 0x0011 to 0x7FFF: No motor type assigned	
DS402_0x6402_MOTORTYPE			ID: 0x6402
RO	Volatile	Unit: increment	U16

2.5.4 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 [chapter \(p. 17\)](#)).

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 size 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 $10^{-3} \times \text{rpm}$ 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.

2.5.5 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	6	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.

2.5.6 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 target velocity (0x60FF).
------------------------	--

DS402_0X60B1_VELOCITY_OFFSET			ID: 0x60B1
RW	Volatile	Unit: –	I32

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 [chapter. \(p. 16\)](#)) 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 thousand (1000 = 100 %) of the rated torque, object motor rated torque (0x6076).	
DS402_0X60B2_TORQUE_OFFSET			ID: 0x60B2
RW	Volatile	Unit: ‰	I16

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**.

2.5.7 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 motor rated torque (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 motor rated current (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 velocity		Speed limit for the motion profile The value applies to both directions. The unit is the same as in object target velocity (0x60FF).	
DS402_0x607F_MAX_PROFILE_VELOCITY			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 speed		Maximum speed specified by the motor The value applies to both directions. The unit is the same as in object target velocity (0x60FF).	
DS402_0x6080_MAX_MOTOR_SPEED			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 motor rated torque (0x6076).	
DS402_0x60E0_POSITIVE_TORQUE_LIMIT_VALUE			ID: 0x60E0
RW	Volatile	Unit: ‰	I16

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 = 100 %) of the rated torque, object motor rated torque (0x6076).	
DS402_0x60E1_NEGATIVE_TORQUE_LIMIT_VALUE			ID: 0x60E1
RW	Volatile	Unit: ‰	I16

2.5.7.7 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 acceleration		Parameterized maximum acceleration of the system The unit is set in object acceleration (0x60AA).	
DS403_0x60C5_MAX_ACCELERATION			ID: 0x60C5
RW	Volatile	Unit rotary motor: 1/s ² Unit linear motor: mm/s ²	U32

Max Deceleration (0x60C6)

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 deceleration		Parameterized maximum deceleration of the system The unit is set in object acceleration (0x60AA).	
DS403_0x60C6_MAX_DECELERATION			ID: 0x60C6

RW	Volatile	Unit rotary motor: 1/s ² Unit linear motor: mm/s ²	U32
----	----------	---	-----

Profile Acceleration (0x6083)

Via this object you can set the acceleration.

Profile acceleration		Acceleration ramp The unit is set in object acceleration (0x60AA).	
DS402_0x6083_PROFILE_ACCELERATION			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_DECELERATION			ID: 0x6084
RW	Volatile	Unit rotary motor: 1/s ² Unit linear motor: mm/s ²	U32

Quick Stop Deceleration (0x6085)

Via this object you can set the deceleration in the event of a quick stop. In the [Option Codes. \(p. 12\)](#) 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_DECELERATION			ID: 0x6085
RW	Volatile	Unit rotary motor: 1/s ² Unit linear motor: mm/s ²	U32

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 motor rated torque (0x6076).	
DS402_0x6087_TORQUE_SLOPE			ID: 0x6087
RW	Volatile	Unit: ‰/s	U32



2.5.8 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 - configurable	
DS402_0X60FD_DIGITAL_INPUTS			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 it 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 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-bit 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 - configurable Sub-IDs: ▶ 0x60FE:0 = Highest subindex supported (2) ▶ 0x60FE:1 = Physical outputs ▶ 0x60FE:2 = Bit mask	
DS402_0X60FE_DIGITAL_OUTPUTS			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 according 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.

2.6 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.

Note

If the PDO mapping is set by fieldbus control, this must be repeated after a parameter set change. Usually, the device must complete the fieldbus initialization phase again (e.g. NMT status change from Pre-Operational to Operational).

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:

- ▶ **00: Fixed selection (EEPROM):**
This option is the factory setting. The device is 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 is 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 is started with parameter set 0. You can switch the parameter set via the fieldbus system only.

The parameter set change via fieldbus is described 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".

2.6.1 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.

3 Control via POWERLINK

The following chapters describe the addressing of the drives in the POWERLINK network and the connection to the PLC using the B&R software “Automation Studio” as an example.

3.1 POWERLINK Network

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

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

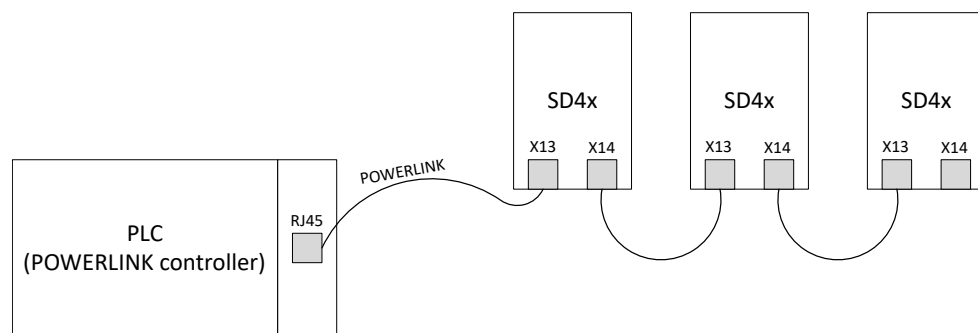


Fig. 13: POWERLINK network

For information on the connector pin assignment of the POWERLINK 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 via the application development system “Automation Studio” by B&R Industrial Automation GmbH.

1. Copy the XDD file provided by SIEB & MEYER to the directory {drive}:\B_R\workspace\xdd.

– e.g.: 000003DD_SD4x_20220224.xdd

The XDD file describes the communication parameters and the object dictionary of the drive.

You can download the XDD file from the SIEB & MEYER website www.sieb-meyer.com. The XDD file version may change with a new firmware.

2. Start the application development system “Automation Studio” and select the menu “Tools → Manage 3rd-Party Devices...”.

A window for managing the device description files appears.

3. Use the import function to include the XDD file.

- ✓ During the import, the program checks the validity of the data. After a successful import, the device data are available in the hardware catalog.

3.2.1 Create Project

Create new project

1. Select the menu “File → New Project...”.
→ The project wizard appears.
2. Enter a name for the project, e.g. Project_1, and click “Next”.
3. Select a name for the hardware configuration. You can also use the default name “Config1”.

With an offline project, select the option “Define a new hardware configuration manually”.

Click “Next”.

4. Select the desired CPU of the PLC. The default CPU is “X20cCP1584”.

With an offline project, set the option “Activate Simulation”.

Click “Finish” to confirm your input.

→ The project is created.

A selection window with the entries “Physical View”, “Logical View” and “Configuration View” is opened. In addition, the window “Hardware.hwl (System Designer)” appears.

5. In the “Configuration View”, select the according CPU. Right-click on the CPU and select the entry “Properties...” in the context menu.
6. In the Properties window, select the tab “I/O”. Activate the option “Disable module supervising for newly added modules” there.

Click “OK” to confirm your input and close the window.

Establish online connection

7. Select the menu “Online → Settings”.

The window “Online Settings” appears:

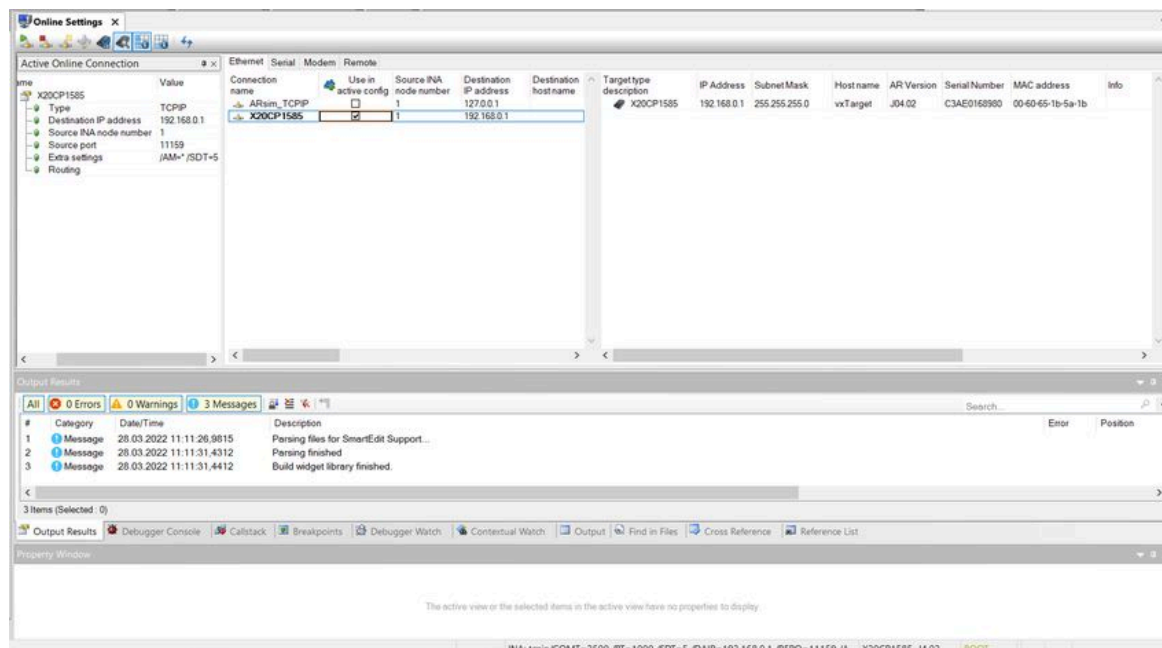


Fig. 14: Online Settings

If the IP address of the PLC is not valid or the IP address is not located in the current subnetwork, the IP address is displayed in red:

Connection name	Use in active config	Source INA node number	Destination IP address	Target type description	IP Address	Subnet Mask	Hostname	AR Version	Serial Number	MAC address	Info
ARsim_TCPIP	<input type="checkbox"/>	1	127.0.0.1	X20CP1585	0.0.0	0.0.0	vxTarget	J04.02	C3AE0168900	00-60-65-1b-5a-1b	IP address or subnet does not match
X20CP1585	<input checked="" type="checkbox"/>	1	192.168.0.1								

Fig. 15: IP address in red

In this case, change the IP address as follows:

- a. Right-click on the CPU and select the entry “Set IP Parameters” in the context menu.
- b. Enter the IP address and the subnet mask in the following window. In addition, you must activate the checkbox “Apply IP Parameters to:” so that the data are permanently changed.

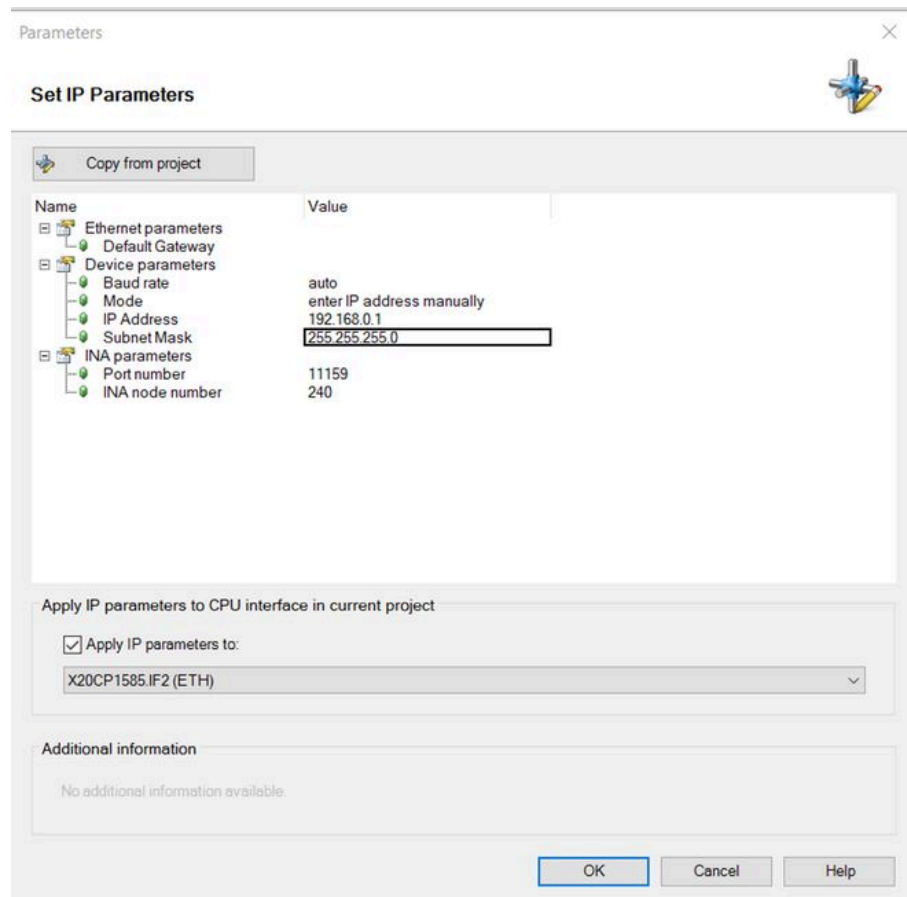


Fig. 16: Set IP address

- c. Click “OK” to confirm the IP parameters and to return to the window “Online Settings”.
8. Right-click on the CPU and select the entry “Connect...” in the context menu.
- After a short time, the connection to the PLC is established. The bottom line of the program displays the current status.

First software transfer

If you want to upload a program to the PLC via Compact Flash for the first time, you must write the Compact Flash card offline.

9. Remove the Compact Flash card from the PLC and insert the card in the card reader of the PC.

Note

For a detailed description on the initial setup of an X20CP1585 refer to the Automation Studio help under “Getting Started → Creating programs with Automation Studio → First project for target systems with Compact Flash”.

10. Install the project on the Compact Flash card via offline installation:

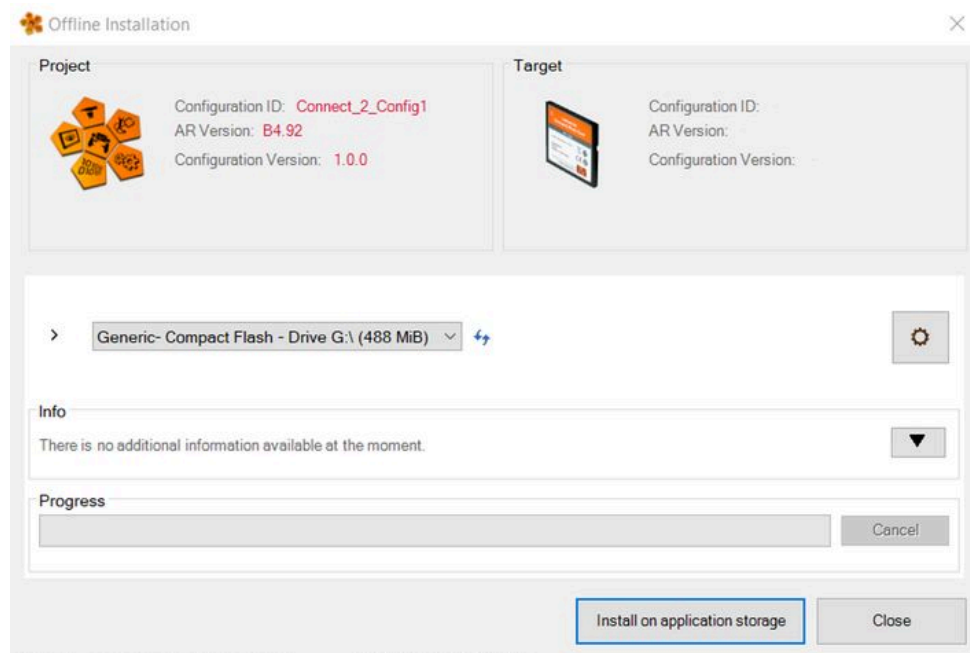


Fig. 17: Compact Flash: project installation

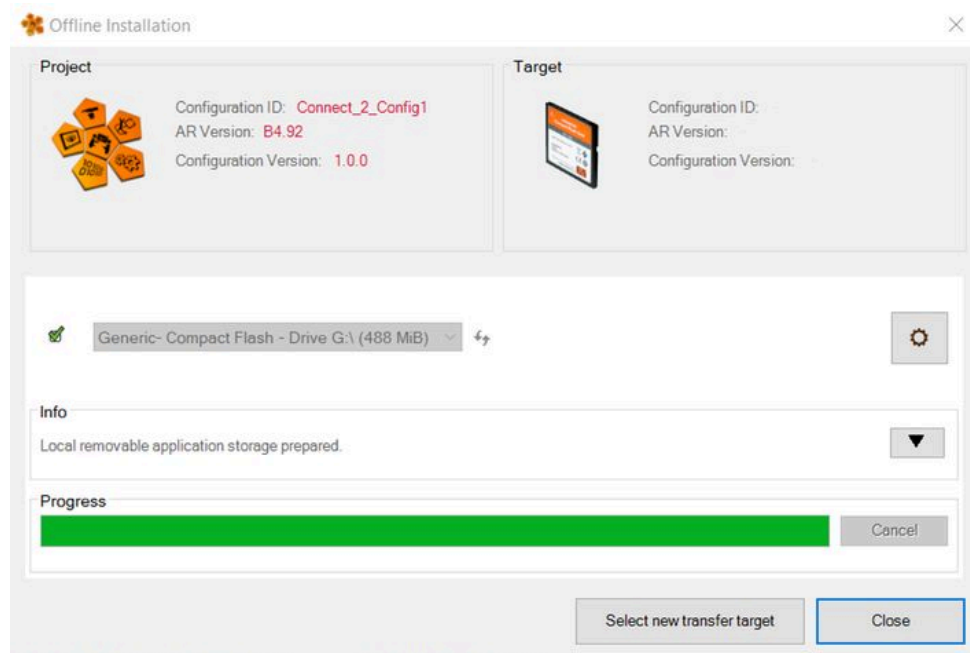






Fig. 18: Compact Flash: project installation finished

11. Reinsert the Compact Flash card in the PLC.

→ After booting the PLC, the LEDs have the following states:

LED	Status
R/E	 Green
S/E	 Green
PLK	 Green flashing
ETH	 Green flashing

Next software transfer

If you change the project, you can use the transfer function to upload the new project to the PLC.

12. Select the menu “Project → Project Installation → Transfer To Target”. The project is translated and the transfer window appears.

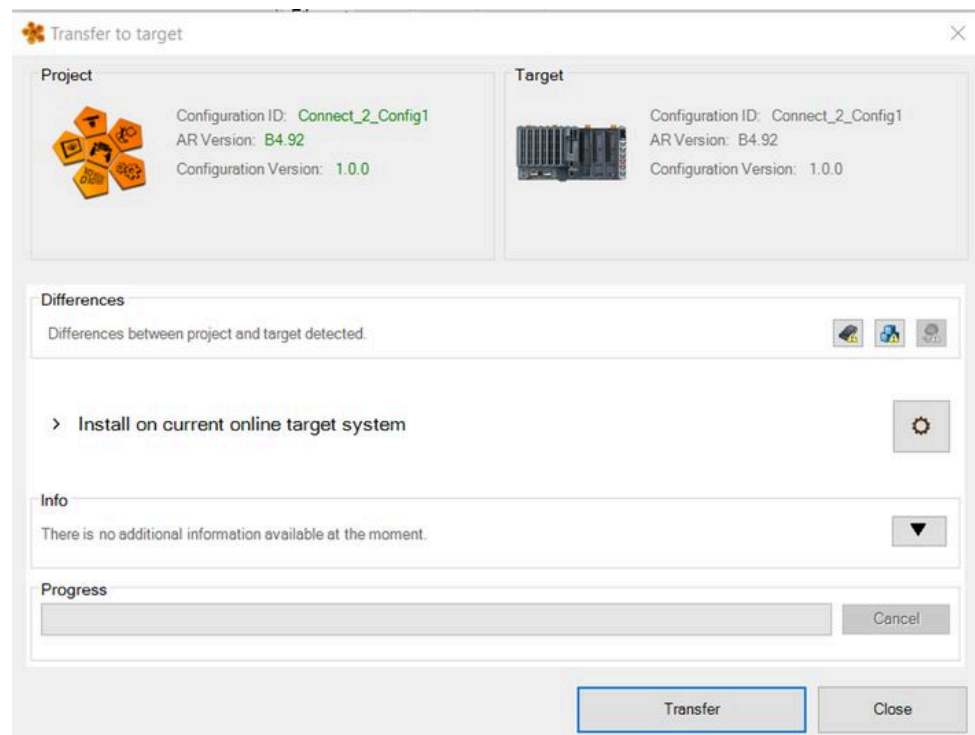


Fig. 19: Software transfer

13. Click “Transfer” to upload the project to the PLC.

3.2.2 Create Device Structure

Select devices

The window “Hardware.hwl (System Designer)” is used for displaying the individual devices and connecting them.

1. Double-click on a device in the hardware catalog to position it in the window “Hardware.hwl (System Designer)”. Then, the device is connected to the CPU.

Note

The hardware catalog provides different filters to select a particular device group, for example:

3rd Party Devices: lists all devices by third party suppliers

POWERLINK: lists all devices for POWERLINK

You can also combine the filters to further reduce the device list.

2. If required, position more devices from the hardware catalog in the window “Hardware.hwl (System Designer)” by double-clicking them. The devices are automatically connected with the previously marked device.

Set node number

3. Select the window “Physical View”.
4. Right-click on the desired device and select “Node Number → Change Node Number”.

A small window opens next to the device in the tree. There, you can set the node number. :

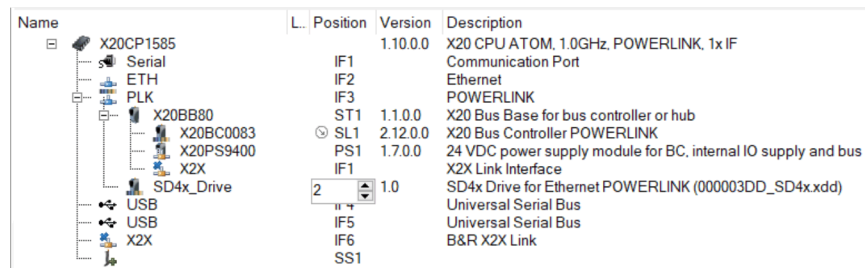


Fig. 20: Node number

Determine cyclic data

5. Select the window “Physical View”.

- Right-click on the desired device and select “Configuration”.

The configuration window appears. Under the point “Channels” all PDO-capable objects are listed. You can unfold the objects:

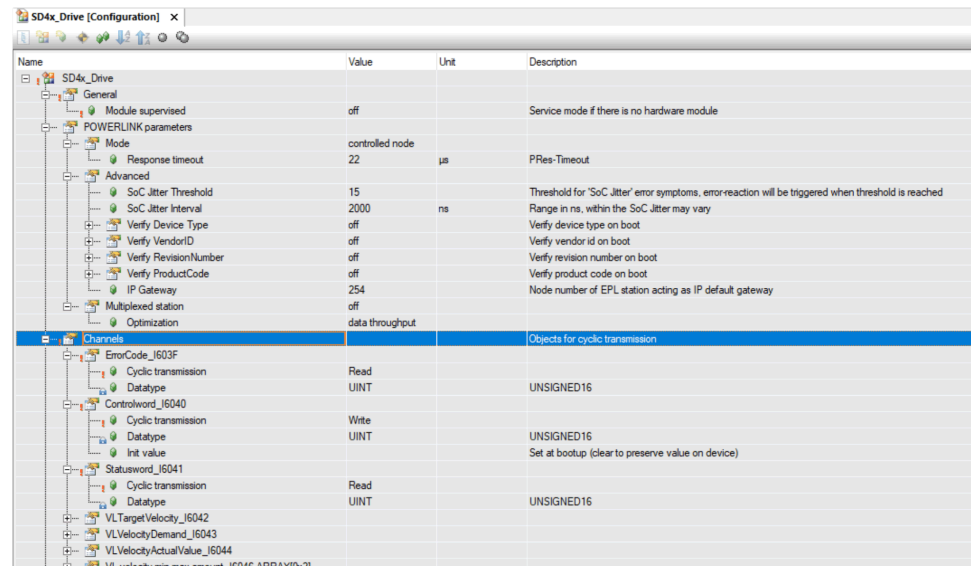


Fig. 21: Configuration

- Under the point “Cyclic Transmission” you can determine, whether the object is transmitted cyclically or not (Read, Write, None).

3.2.3 Create ST Program

- Select the window “Logical View”.
- In the “Object Catalog”, double-click on the object “ST Program”.
 - The program object is added in the tree.

Add library

- In the window “Logical View”, right-click on the menu item “Libraries” and select “Add Object ...” in the context menu.
- In the “Object Catalog” under “Programmable Object Units”, click “Library”.
 - A list of libraries opens.
- Select the entry “B&R Libraries”.
 - A window listing the B&R libraries appears.
- Select the desired library and click “Finish” to confirm your selection.
 - The library is added in the tree.

Create process variables

- In the menu tree, select the point “Global.var” or “Program → Variables.var” or similar.
 - A window appears, in which you can create the required variable for the process.

Mapping of I/O variables

- Select the window “Physical View”.
- Double-click on the device to open the window “I/O Mapping” with the PDO variables.

Types of communication:

- (1) No communication
- (2) Listen only
- (3) Legacy Ethernet (IP and others)
- (4) POWERLINK reduced cycle (SoA, Asnd)
- (5) POWERLINK cycle (SoC, PReq, PRes, SoA, Asnd)

NMT status	Value	Description
NMT_GS_INITIALISATION	0x09	This status controls the initialization and the reset mode. It is divided into 4 other substates.
NMT_CS_NOT_ACTIVE	0x1C	This is a non-permanent status that enables the starting drive to recognize the current network status. For this purpose, the drive observes the current network traffic. The drive is not authorized to send frames autonomously.
NMT_CS_PRE_OPERATIONAL_1	0x1D	In this status, the drive downloads its configuration data from the MDevice. The transition to the next status is triggered by the receipt of a SoC frame. The drive is not authorized to send frames autonomously. There is no PDO communication.
NMT_CS_PRE_OPERATIONAL_2	0x5D	In this status, the drive waits for the configuration to be completed. The received PDO data may be invalid. Therefore, the drive ignores them. The drive is not authorized to send frames autonomously. The transition to the status NMT_CS_READY_TO_OPERATE is triggered by an NMTEnableReadyToOperate command, when the application is ready for operation.
NMT_CS_READY_TO_OPERATE	0x6D	With this status, the drive informs the MDevice of its readiness for operation. The drive participates in the cyclic data exchange. The drive is not authorized to send frames autonomously. The transition to the status NMT_CS_OPERATIONAL is triggered by the receipt of the NMT command "NMTStartNode".
NMT_CS_OPERATIONAL	0xFD	This is the normal operational status of the drive. The drive participates in the cyclic data exchange. The drive is not authorized to send frames autonomously. The drive interprets the received PDO data. All transmitted data must comply with the requirements defined by the PDO mapping.
NMT_CS_STOPPED	0x4D	In this status, the drive is mostly passive. This status is used for controlled shutdown of the drive while the system is still running. The drive does not participate in the cyclic data exchange but still observes SoA frames. The drive is not authorized to send frames autonomously.
NMT_CS_BASIC_ETHERNET	0x1E	In this status the drive may communicate via Legacy Ethernet (standard Ethernet). There is no POWERLINK-specific network traffic control. The drive can send frames autonomously. The drive recognizes SoC, Preq, Pres and SoA frames. On receipt of such a frame, the drive immediately stops all autonomous transmission and switches to the status NMT_CS_PRE_OPERATIONAL_1.

4 Drive Parameterization and Diagnostics

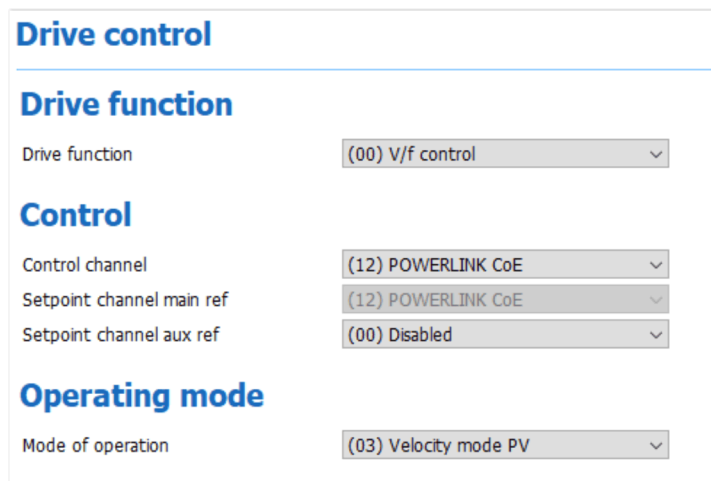
The following chapters describe the parameterization and diagnostics of an SD4x drive with the software *drivemaster4*.

4.1 Parameterization in *drivemaster4*

Before you can operate the SD4x drive in the POWERLINK network, you must set the POWERLINK 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 “POWERLINK CoE”. The reference value channel (main ref) is automatically set the same.



Drive control	
Drive function	
Drive function	(00) V/f control
Control	
Control channel	(12) POWERLINK CoE
Setpoint channel main ref	(12) POWERLINK CoE
Setpoint channel aux ref	(00) Disabled
Operating mode	
Mode of operation	(03) Velocity mode PV

Fig. 23: POWERLINK 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 “POWERLINK” and set the communication parameters for the fieldbus.

In order to communicate via POWERLINK with the drive, you must set the parameters “Node ID select” and “Node ID” correctly:

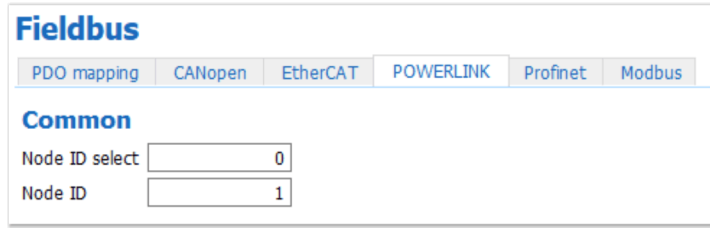


Fig. 24: Fieldbus parameterization in drivemaster4

Node ID select:

- value = 0: definition in parameter “Node ID”
- value > 0: reserved

Node ID:

The parameter indicates the bus address of the drive. The node ID value is in the range of 1 (default) to 239.

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 POWERLINK.

4.1.1 Address Offset for Node ID

This function is available from firmware version Vx.7.x onwards.

The node ID address offset allows changing the POWERLINK node ID via the digital inputs. This function is executed only during the boot phase of the SD4x drive.

Parameterization of the digital inputs

The *drivemaster4* software provides the function (23) “Address offset (fieldbus)” for the digital inputs. You can select this function for several inputs. Use the parameter **DINx bit number** to select the bit of the node ID that is to be changed:

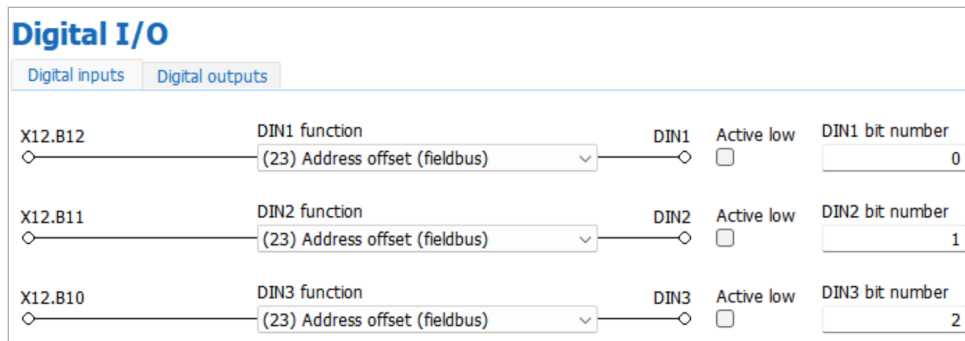


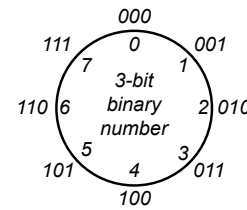
Fig. 25: Address offset function in digital inputs

The node ID is a number between 0 and 255 and therefore consists of 8 bits.



The following truth table shows the resulting address offsets when 3 bits of the node ID are changed via the digital inputs:

Status			Address offset
DIN3	DIN2	DIN1	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7



Note

The software does not check for incorrect parameter settings. If, for example, the bit number is set to 0 for two inputs, the two inputs are ORed to bit 0 of the address offset. That means, the address offset is 1 when one or the other input is set to TRUE.

Calculation of the POWERLINK node ID

The address offset configured in the digital inputs is added to the parameterized node ID (see parameter page “Fieldbus”, tab “POWERLINK”):

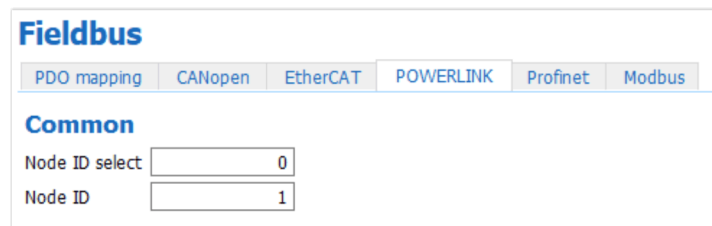


Fig. 26: Parameterized node ID

Note

The parameter **Node ID select** is not used at present.

Example: When the evaluation of the digital input results in an address offset of 2 and the parameterized node ID is 1, the resulting node ID is 3.

The resulting node ID is written to the multiprotocol chip NetX90 of the SD4x drive during the boot phase. If you change the parameterized node ID or the bits of the address offset, you must reset the device (e.g. disconnect and reconnect the logic supply or via the software *drivemaster4*) so that the SD4x drive is rebooted.

Note

Reaction to incorrect parameterization: if the parameterized node ID plus address offset results in a value outside the valid range (1 to 239), the SD4x drive sets the node ID to 1.

The object **Node ID actual** contains the resulting POWERLINK node ID (see object browser in *drivemaster4* software):

Name	Short name	Object	ID	Sub-ID	Value	Unit
Communication		COMMUNICATION				
POWERLINK		GROUP				
POWERLINK - Node ID	Node ID	POWERLINK_NODE_ID	0x0096		1	
POWERLINK - Node ID select	Node ID select	POWERLINK_NODE_ID_SELECT	0x0097		0	
POWERLINK - Node ID actual	Node ID actual	POWERLINK_NODE_ID_ACTUAL	0x018c		3	

Fig. 27: Node ID actual in object browser

4.2 Diagnosis in *drivemaster4*

In the user interface *drivemaster4*, the page “Diagnosis → Fieldbus” displays the transmitted telegrams between drive and PLC in the tab “POWERLINK”.

4.2.1 Telegrams

The following figure shows the logged telegrams on the POWERLINK diagnosis page.

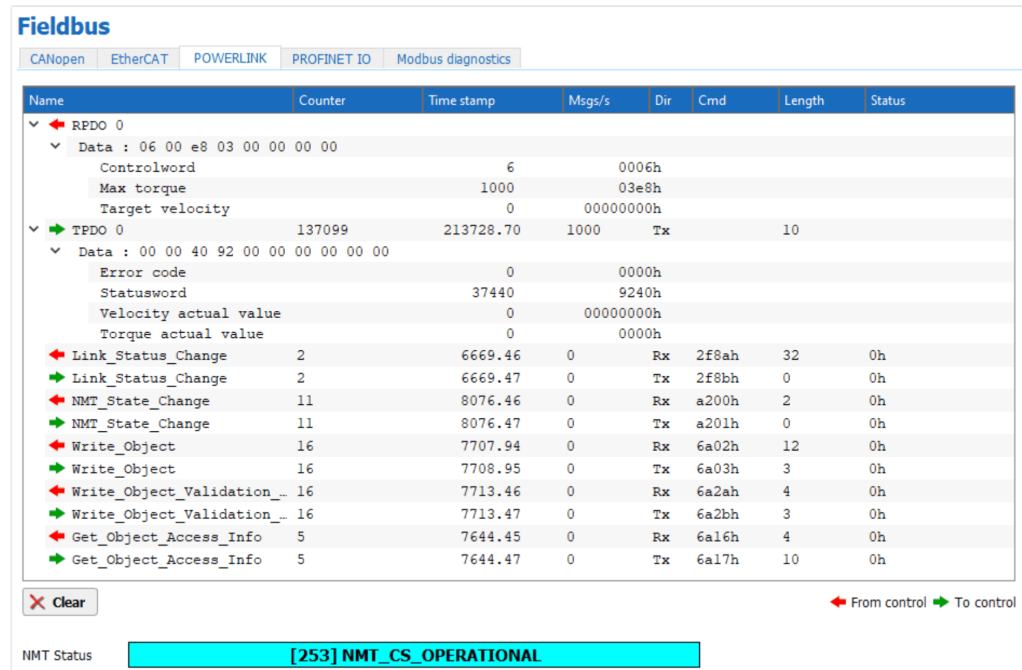


Fig. 28: POWERLINK bus activities

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.

NMT status

This field displays the NMT status (POWERLINK communication status). The following states are possible:

- ▶ NMT_GS_OFF
- ▶ NMT_GS_INITIALISATION
- ▶ NMT_CS_NOT_ACTIVE
- ▶ NMT_CS_PRE_OPERATIONAL_1
- ▶ NMT_CS_PRE_OPERATIONAL_2
- ▶ NMT_CS_READY_TO_OPERATE
- ▶ NMT_CS_OPERATIONAL
- ▶ NMT_CS_STOPPED
- ▶ NMT_CS_BASIC_ETHERNET

4.2.2 Error Status

In addition, you will find status information on the right side of the POWERLINK diagnosis page:

Status

No entry

Index	0x0000
Entry type	0x0000
Status code	0x0000

Error

No entry

Entry type	0x0000
Error code	0x0000

For more information on a displayed status or error code refer to the POWERLINK documentation.

5 Reference

Other documents

The following documents provide more information on this topic:

Supplier	Document
SIEB & MEYER AG	<ul style="list-style-type: none"> ▶ drivemaster4 - User Manual ▶ Drive System SD4 – Ethernet Configuration
Ethernet POWERLINK Standardization Group ⁽¹⁾	<ul style="list-style-type: none"> ▶ EPSG WDP 301, Communication Profile Specification ▶ EPSG DS 311, XML Device Description (2016)
CiA e.V.	<ul style="list-style-type: none"> ▶ 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

⁽¹⁾ The Ethernet POWERLINK Standardization Group dissolved in March 2023. Currently, the company B&R revises the documents. When new documents are available, they will be added here.

Websites

The following websites provide more information on this topic:

Supplier	Document
SIEB & MEYER AG	www.sieb-meyer.com
B&R Industrial Automation GmbH	www.br-automation.com
CiA e.V.	www.can-cia.org



Index

A

- Abbreviations [5](#)
- Address offset (fieldbus) [47](#)
- Address Offset for Node ID [47](#)
- Automation Studio [37](#)

C

- Control commands of drive [10](#)
- Controlword [10](#)
- CSV mode [14](#)
- Cyclic synchronous velocity mode [14](#)

D

- Diagnosis
 - drivemaster4 [49](#)

N

- Network [37](#)
- NMT state machine [44](#)

O

- Objects
 - abort connection option code [12](#)
 - acceleration [29](#)
 - controlword [10](#)
 - current actual value [27](#)
 - DC link circuit voltage [27](#)
 - digital inputs [34](#)
 - digital outputs [34](#)
 - disable operation option code [13](#)
 - error code [35](#)
 - fault reaction option code [13](#)
 - halt option code [13](#)
 - interpolation time period [25](#)
 - jerk [30](#)
 - max acceleration [32](#)
 - max current [31](#)
 - max deceleration [32](#)
 - max motor speed [32](#)
 - max profile velocity [31](#)
 - max torque [31](#)
 - mode of operation [24](#)
 - mode of operation display [24](#)
 - motion profile type [26](#)
 - motor rated current [28](#)
 - motor rated torque [28](#)
 - motor type [28](#)
 - negative torque limit [32](#)
 - polarity [30](#)
 - position [29](#)
 - position actual value [26](#)
 - positive torque limit [32](#)
 - profile acceleration [33](#)
 - profile deceleration [33](#)
 - quick stop deceleration [33](#)
 - quick stop option code [12](#)
 - shutdown option code [13](#)
 - statusword [11](#)
 - supported drive modes [23](#)
 - target torque [25](#)
 - target velocity [25](#)
 - torque actual value [27](#)
 - torque demand [27](#)
 - torque offset [31](#)
 - torque slope [33](#)
 - velocity [29](#)
 - velocity actual value [27](#)
 - velocity demand value [27](#)
 - velocity offset [30](#)
 - vl dimension factor [21](#)
 - vl set-point factor [21](#)
 - vl target velocity [17](#)
 - vl velocity acceleration [19](#)
 - vl velocity actual value [18](#)
 - vl velocity deceleration [19](#)
 - vl velocity demand [18](#)
 - vl velocity min max amount [18](#)

vl velocity quick stop [20](#)

Operating mode change [23](#)

Operating modes [14](#)

Option codes [12](#)

P

Parameter set change [35](#)

PDO [8](#)

PLC connection [37](#)

Profile torque mode [22](#)

Profile velocity mode [16](#)

S

SDO [7](#)

State machine [9](#)

State transitions of the drive [9](#)

States [11](#)

Statusword [11](#)

T

Torque mode [22](#)

V

Velocity mode [16](#)

Velocity mode PV [16](#)

VL [16](#)