X20SM1444-1

Data sheet

1.20 (September 2024)

Translation of the original documentation

Publishing information

B&R Industrial Automation GmbH

B&R Strasse 1

5142 Eggelsberg

Austria

Telephone: +43 7748 6586-0

Fax: +43 7748 6586-26

office@br-automation.com

Disclaimer

Version history

B&R makes every effort to keep documents as current as possible. The most current versions are available for download on the B&R website (www.br-automation.com).

1 General information

1.1 Other applicable documents

For additional and supplementary information, see the following documents.

Other applicable documents

Document name	Title
MAX20	X20 System user's manual
MAEMV	Installations / EMV guide

1.2 Order data

Order number	Short description	Figure
	Motor controllers	
X20SM1444-1	X20 stepper motor module, module power supply 24 to 48 VDC ±25%, with current reduction function, 1 motor connection, 5 A continuous current, 10 A peak current, 5 V ABR, quick stop / enable, NetTime function, double-width module	
	Required accessories	
	Bus modules	
X20BM31	X20 bus module, for double-width modules, 24 VDC keyed, internal I/O power supply connected through	
	Terminal blocks	
X20TB12	X20 terminal block, 12-pin, 24 VDC keyed	

Table 1: X20SM1444-1 - Order data

1.3 Module description

The stepper motor module is used to control stepper motors with a nominal voltage of 24 to 48 VDC (±25%) at a motor current up to 5 A (10 A peak).

Functions:

- Integrated motor detection
- Currents independently configurable
- Load-dependent current control
- Stall detection
- Homing
- Limitations
- Motion generator
- Counters
- ABR interface and digital inputs
- Quick stop and enable
- Automatic shutdown
- NetTime Technology

Integrated motor detection

Due to the integrated motor detection, the connected motors can be identified by their coil characteristics and feedback in the form of an analog value can be generated. This makes it possible to detect not only wiring errors, but also incorrect motor types being used.

Currents independently configurable

With individual adjustment of the coil currents, the motor is only operated with the current it actually needs. This prevents unnecessary heating of the motor. This lower heating has a positive effect on the energy consumption, thermal load and thus on the service life of the complete system.

Load-dependent current control

The module contains a sensorless, load-dependent current control. Depending on the operating situation and load, the module controls the current downwards. This results in energy savings of up to 75%.

Stall detection

Stall detection is integrated to analyze the motor load. Motor stall detection is defined by a configurable threshold. This allows an overload or motor stall to be detected sufficiently accurately in many applications.

Homing

A large selection of homing procedures allows a wide range of applications for the module. In addition to procedures triggered by limit switches, stop-controlled and immediate homing procedures can also be defined.

For end stop-controlled homing, a delay time can also be defined for stall detection. Homing is only performed after the delay time has elapsed. This prevents unwanted homing due to accidental, brief stoppage of the motor.

Limitations

Software limitations and/or hardware limit switches allow precise control of the maximum range of movement. This prevents limit transgressions that could result in damage to the machine.

Motion generator

Movements can be generated directly by the module. Homing or exact positioning procedures can be implemented with minimal application effort using a specified target position and acceleration as well as several other parameters.

Counter functions

The exact position of the motor can be determined either by an AB(R) encoder or by internal calculations in the module. This allows a large number of motors to be used that are precisely adapted to the machine requirements.

Inputs/Outputs

The module is equipped with digital inputs that can additionally be used as homing inputs, limit switches or AB(R) encoder inputs.

Quick stop and enable

By using two digital inputs as quick stop and enable inputs, the motor can be quickly braked or set to free-wheel.

Automatic shutdown

The voltage of the module power supply and the module temperature are monitored. If a value overshoots the predefined limit value, the module is automatically shut down. The outputs are automatically started up again as soon as the value is within the limit value again.

In addition, SDC life sign monitoring can be used to monitor the communication between the controller and module. An interruption of the communication triggers an automatic shutdown of the motor by the module.

The module current, the voltage of the module power supply and the module temperature are monitored. If a value overshoots the predefined limit value, the module is automatically shut down.

NetTime timestamp of the position and trigger time

It is not just the position value that is important for highly dynamic positioning tasks, but also the exact time the position is measured. The module is equipped with a NetTime function for this that supplies a timestamp for the recorded position and trigger time with microsecond accuracy.

2 Technical description

2.1 Technical data

Order number	**X20SM1444-1**
Short description	
I/O module	1 full bridge for controlling stepper motors
General information	
B&R ID code	0xF3ED
Status indicators	I/O function per channel, operating state, module status
Diagnostics	
Module run/error	Yes, using LED status indicator and software
Output	Yes, using LED status indicator and software
I/O power supply	Yes, using software
Power consumption	
Bus	0.1 W
Internal I/O	-
External I/O [1]	At 18 VDC 0.54 W At 24 VDC 0.6 W At 40 VDC 0.76 W At 60 VDC 0.97 W
Additional power dissipation caused by actuators (resistive) [W] [2]	At 1 A 0.31 W At 2 A 0.574 W At 3 A 0.848 W At 4 A 1.1 W At 5 A 1.4 W
Input current limiting	Max. 12.5 A
Type of signal lines	Shielded lines must be used for all signal lines.
Certifications	
CE	Yes
UKCA	Yes
Motor bridge - Power unit	
Quantity	1
Type	2-phase bipolar stepper motor (full bridge)
Nominal voltage	24 to 48 VDC ±25% [3]
Nominal current	5 A
Maximum current	10 A for 1 s [4]
DC bus capacitance	300 µF
Step resolution	Max. 256 microsteps per step
Module power supply	
Supply	External
Fuse	Required line fuse: Max. 10 A, slow-blow
Output protection	Reverse polarity protection on supply voltage
Digital inputs	
Quantity	2
Nominal voltage	24 VDC
Input circuit	Sink
Input filter	
Hardware	<5 µs
Software	-
Connection type	1-wire connections
Input resistance	Typ. 8.2 kΩ
Additional functions	Use as R+ and R- for ABR interface, quick stop and enable function [5]
Open-circuit and short-circuit detection	No
Switching threshold	
Low	<7.5 VDC
High	>11.5 VDC
ABR incremental encoder	
Quantity	1
Encoder inputs	5 V, symmetrical
Counter size	16-bit
Input frequency	Max. 125 kHz
Evaluation	4x
Open-circuit detection	Yes, differential detection
Electrical properties	
Electrical isolation	Channel isolated from bus and channel isolated from I/O power supply

Table 2: X20SM1444-1 - Technical data

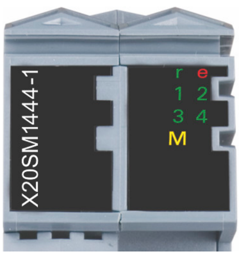
Order number	X20SM1444-1
Operating conditions	
Mounting orientation	
Horizontal	Yes
Vertical	Yes
Installation elevation above sea level	
0 to 2000 m	No limitation
>2000 m	Reduction of ambient temperature by 0.5°C per 100 m
Degree of protection per EN 60529	IP20
Ambient conditions	
Temperature	
Operation	
Horizontal mounting orientation	-20 to +50°C
Vertical mounting orientation	-20 to +40°C
Derating	See section "Derating".
Storage	-25 to +80°C
Transport	-25 to +80°C
Relative humidity	
Operation	5 to 95%, non-condensing
Storage	5 to 95%, non-condensing
Transport	5 to 95%, non-condensing
Mechanical properties	
Note	Order 1x terminal block X20TB12 separately. Order 1x bus module X20BM31 separately.
Pitch	$25^{+0.2}$ mm

Table 2: X20SM1444-1 - Technical data

1) Depends on the supplied voltage
2) Depends on the strength of the motor current
3) The tolerance value is composed of the voltage tolerances and permissible total AC voltage component with a peak value of 5% of the rated voltage.
4) See section "Let-through energy I2T".
5) See section "ABR interface and digital inputs".

2.2 LED status indicators

For a description of the various operating modes, see section "Additional information - Diagnostic LEDs" in the X20 System user's manual.

Figure	LED	Color	Status	Description
	r	Green	Off	No power to module
			Single flash	Mode RESET
			Double flash	Mode BOOT (during firmware update)[1]
			Blinking	Mode PREOPERATIONAL
			On	Mode RUN
	e	Red	Off	Module not supplied with power or everything OK
			On	Error or reset state
	e + r	Solid red / Single green flash		Invalid firmware
	1	Green	On/Off	Input state of encoder input A
			Blinking	Open circuit or not connected
	2	Green	On/Off	Input state of encoder input B
			Blinking	Open circuit or not connected
	3	Green	On/Off	Input state of encoder input R or digital input 1
			Blinking	Open circuit or not connected [2]
	4	Green	On/Off	Input state of digital input 2
	M	Orange	On	The motor is active.

1) Depending on the configuration, a firmware update can take up to several minutes.
2) Only when used as encoder input R

2.3 Pinout

To guarantee full motor power, voltage drops that could result from the cable length and the electrical connections must also be taken into account when selecting the attachment cable.

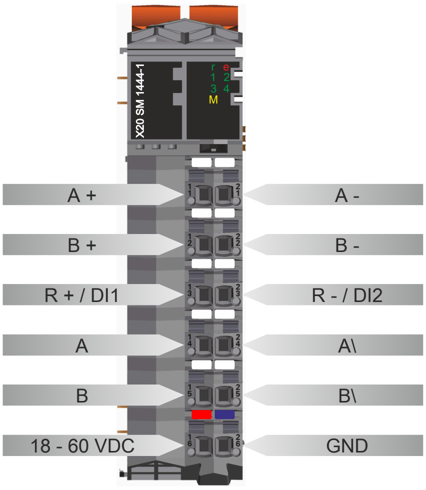
Shielded cables must be used for all signal lines.

Warning!

The terminal block is not permitted to be connected or disconnected during operation.

Information:

Shielded motor cables must be used in order to meet the limit values per standard EN 55011 (emissions).

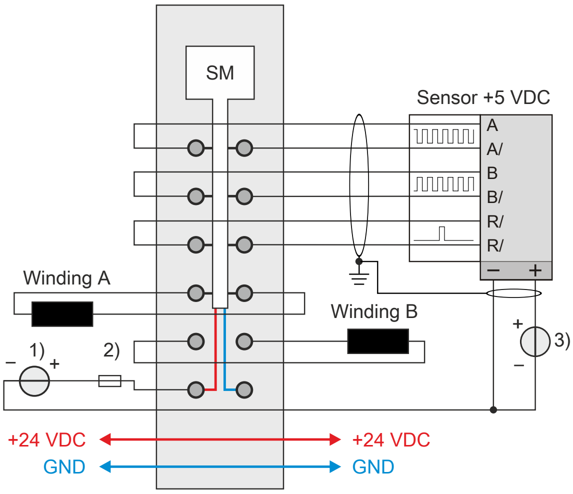


2.4 Connection example

Information:

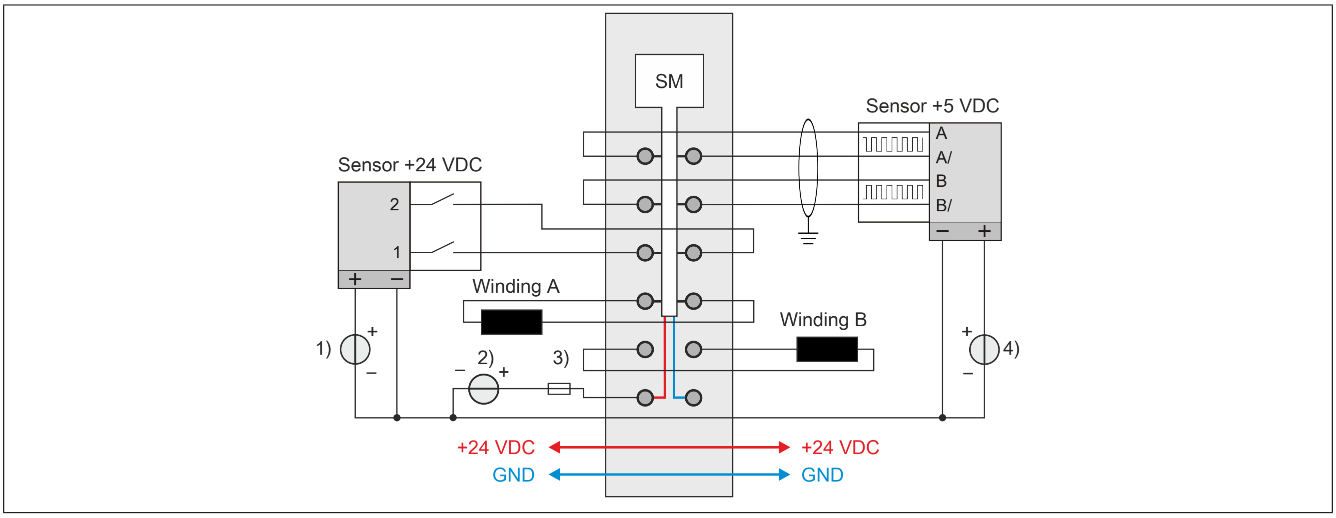
This module can only be operated if supplied with power via the terminal block.

Pinout in standard mode



1) 18 to 60 VDC supply
2) Fuse, 10 A slow-blow
3) 5 VDC supply

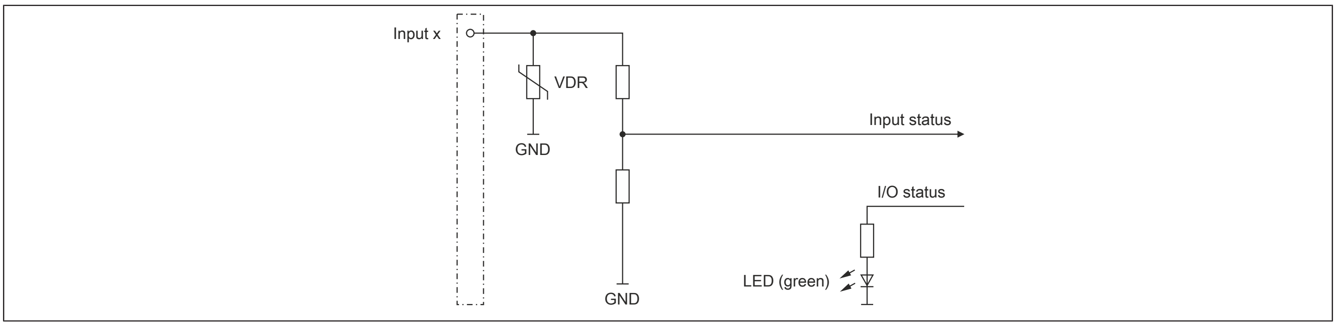
Pinout in extended mode



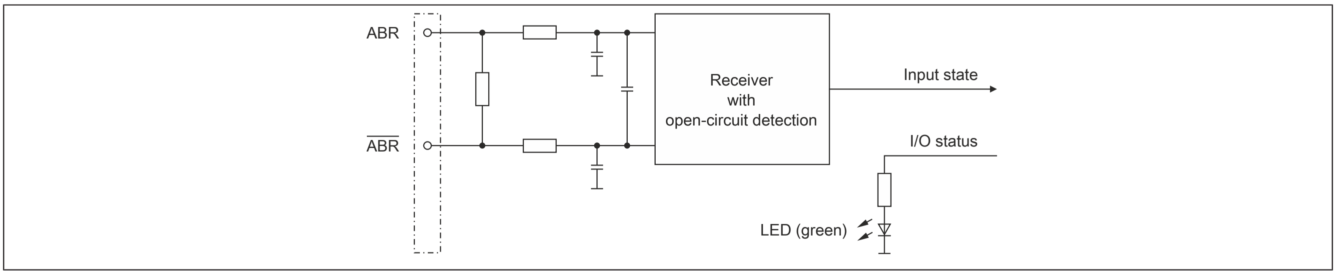
1) 24 VDC supply
2) 18 to 60 VDC supply
3) Fuse, 10 A slow-blow
4) 5 VDC supply

2.5 Input circuit diagram

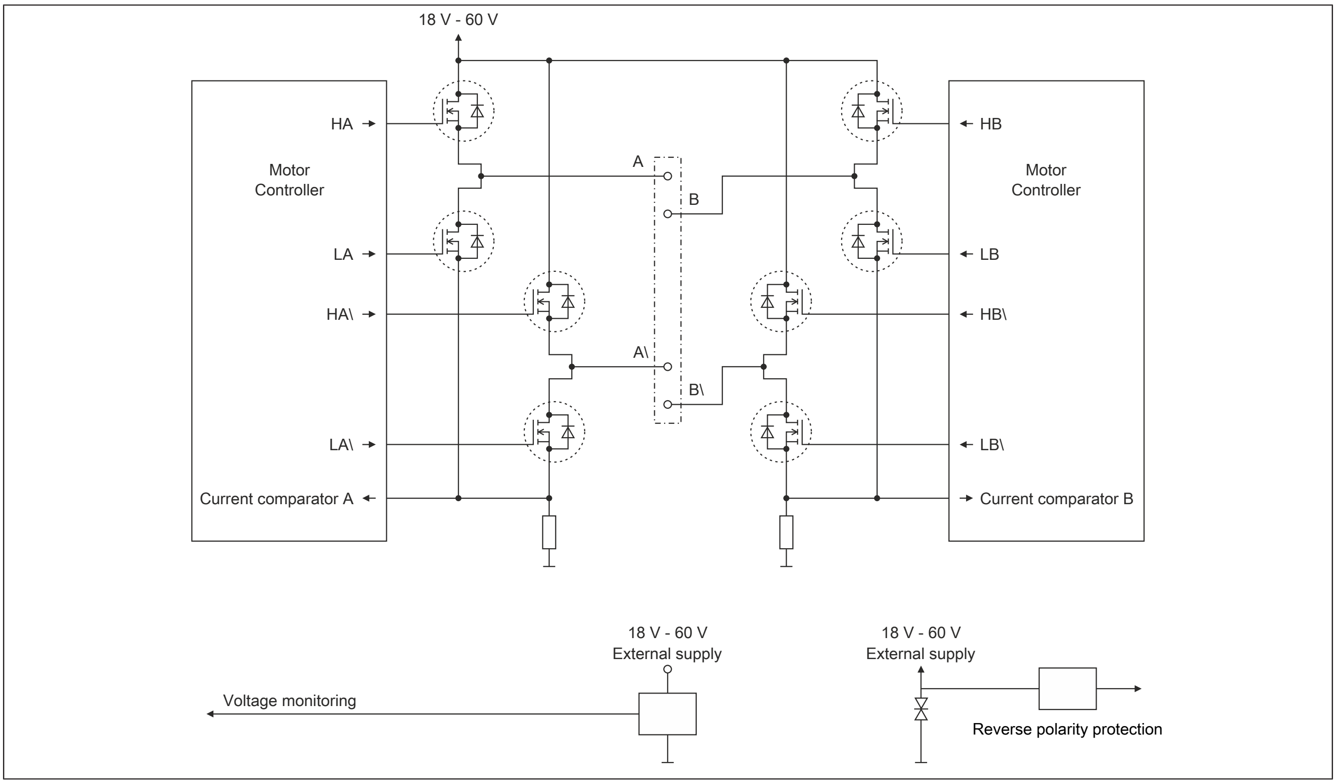
Digital inputs



Counter inputs



2.6 Output circuit diagram



2.7 Power supply unit dimensioning

The motor's current consumption depends on the defined motor currents, the available power and the actual motor being used.

Example	
Order number of the motor	80MPD5.300S000-01
Configured current in the motor module	3 A
Supply voltage of the motor module	48 VDC
Motor load	1 Nm

Table 3: Power supply unit dimensioning example - Basic data

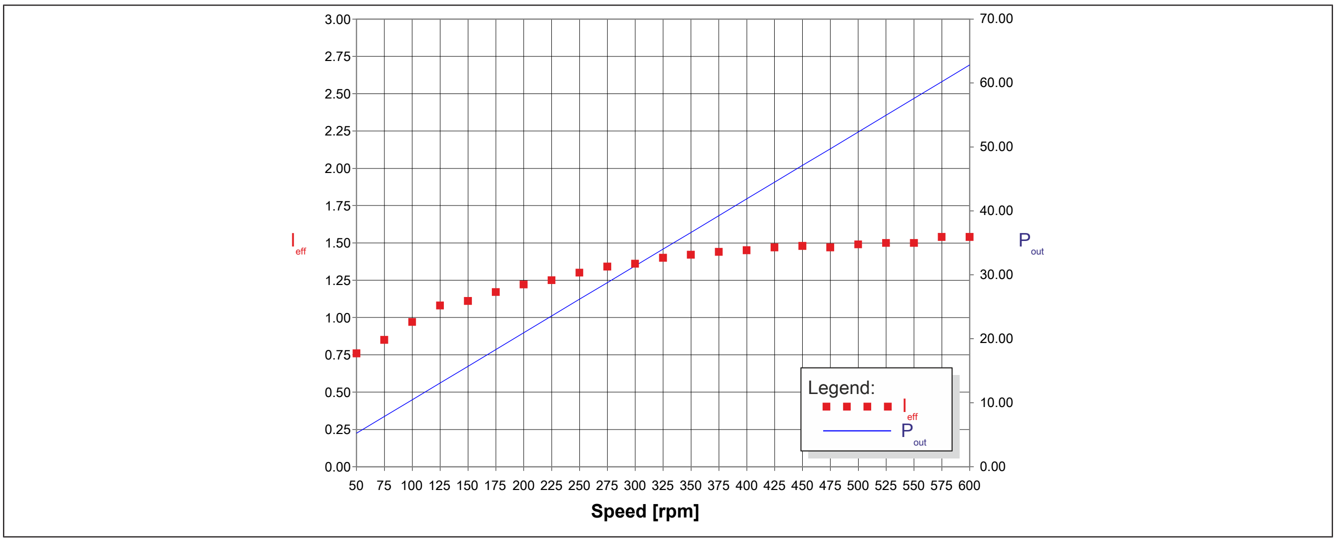


Figure 1: Power supply unit dimensioning example - Power/Speed relationship

The example is based on a constant load throughout the entire speed range.

An increase in the motor load causes an increase in the effective current of the I/O power supply.

2.8 Fuse protection

Line protection must be provided in the power supply line via a circuit breaker or fuses. In general, dimensioning the supply line and overcurrent protection depends on the structure of the power supply (modules can be connected individually or in groups).

Information:

The effective current for the power supply depends on the load but is always less than the motor current. Make sure the maximum nominal current of 10 A is not exceeded on the power supply terminals of the power unit.

When choosing a suitable fuse, the user must also account for characteristics such as aging effects, temperature derating, overcurrent capacity and the definition of the rated current, which can vary by manufacturer and type. In addition, the fuse that is selected must also be able to handle application-specific characteristics (e.g. overcurrent that occurs in acceleration cycles).

The cross section of the mains power input and the rated current of the used fuse are chosen according to the current-carrying capacity such that the permissible current-carrying capacity of the selected cable cross section (depending on wiring, see table) is greater than or equal to the current load in the mains power input. The rated current of the fuse protection must be less than or equal to the permissible current-carrying capacity of the selected cable cross section (depending on the how it is installed, see table):

$$I_{Mains} \leq I_b \leq I_Z$$

Mains ≤ Fuse ≤ Line/Cable

Line cross section [mm²]	Current-carrying capacity of cable cross section I_Z / rated current of fuse I_b [A] depending on the to type of wiring at an ambient air temperature of 40°C per EN 60204-1			
	B1	B2	C	E
1.5	13.5 / 13	13.1 / 10	15.2 / 13	16.1 / 16
2.5	18.3 / 16	16.5 / 16	21 / 20	22 / 20

Table 4: Cable cross section of the mains power input depending on the type of wiring

The tripping current of the fuse is not permitted to exceed the rated current of the fuse I_b.

Type of wiring	Description
B1	Wires in conduit or cable duct
B2	Cables in conduit or cable duct
C	Cables or lines on walls
E	Cables or lines on open-ended cable tray

Table 5: Type of wiring used for the mains power input

2.9 Let-through energy I2T

The module is designed for a continuous current of 5 A. A higher current can be temporarily drawn, however. The following points must be observed:

- The module is designed for a let-through energy of 825 A^2s for a period of 30 seconds.

Information:

Exceeding the maximum let-through energy can result in damage to the module.

- If maximum current I_{boost} is drawn for maximum time t_{boost}, the continuous current of 5 A can be drawn for the remaining time of 30 seconds.
- If maximum current I_{boost} is drawn for more than maximum time t_{boost}, the current is not permitted to exceed the calculated residual current for the remaining time of 30 seconds (see example below).
- At the end of a 30 second period with increased current consumption, 5 A continuous current or a higher current can be drawn again.

Calculating the residual current

$$I_{boost}{}^{2} * t_{boost} + I_{residual}{}^{2} * (30 - t_{boost}) \leq 825 A^2 s$$

$$I_{residual} = \sqrt{\frac{825\,A^2\,s - I_{boost}{}^{2} * t_{boost}}{30\,s - t_{boost}}}$$

Example

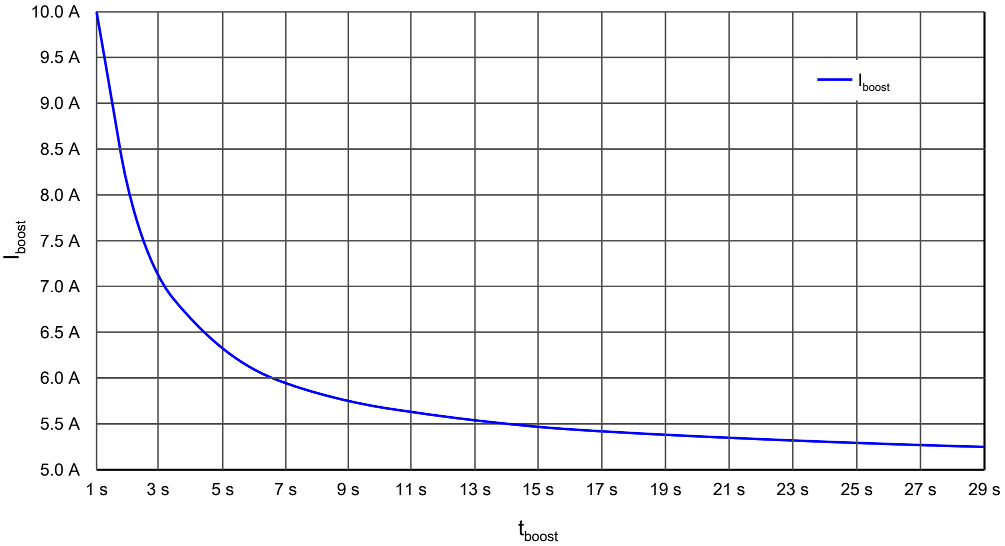
A boost current of 8 A is needed for a duration of 3 seconds. According to the formula, the residual current of 4.84 A is not permitted to be exceeded for the remaining 27 seconds.

$$I_{residual} = \sqrt{\frac{825\,A^2 s - 8\,A^2 * 3\,s}{30\,s - 3\,s}} = 4.84\,A$$

I_{boost} values if I2T = 825 A²s and I_{rest} = 5 A

t_{boost}	I_{boost}	$t_{residual}$		t_{boost}	I_{boost}	$t_{residual}$
s	A	s		s	A	s
1	10.00	29		16	5.45	14
2	7.91	28		17	5.42	13
3	7.07	27		18	5.40	12
4	6.61	26		19	5.38	11
5	6.32	25		20	5.36	10
6	6.12	25		21	5.35	9
7	5.98	23		22	5.33	8
9	5.77	21		23	5.32	7
10	5.70	20		24	5.30	6
11	5.64	19		25	5.29	5
12	5.59	18		26	5.28	4
13	5.55	17		27	5.27	3
14	5.51	16		28	5.26	2
15	5.48	15		29	5.25	1
				30	5.24	0

These values correspond to the following curve for let-through current I2T:

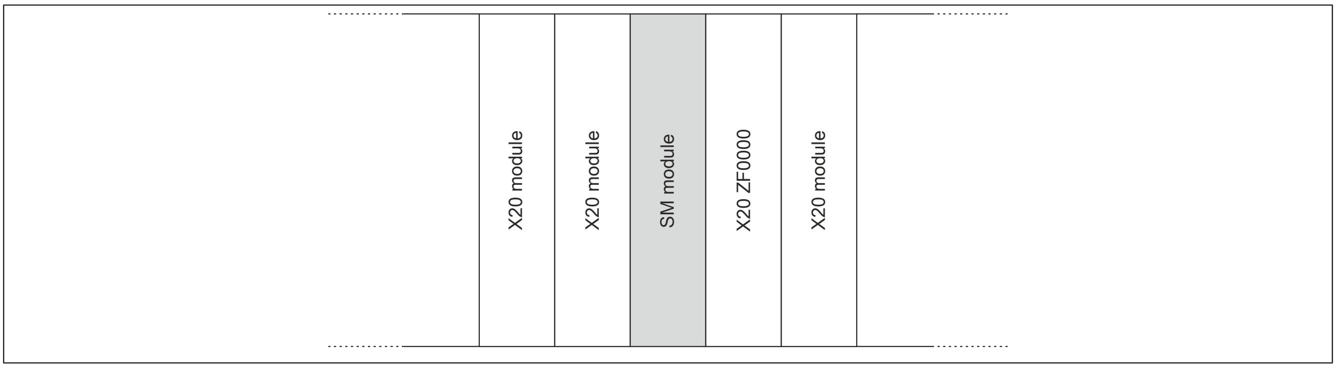


2.10 Derating

For an example of calculating the power dissipation of I/O modules, see section "Mechanical and electrical configuration - Power dissipation of I/O modules" in the X20 user's manual.

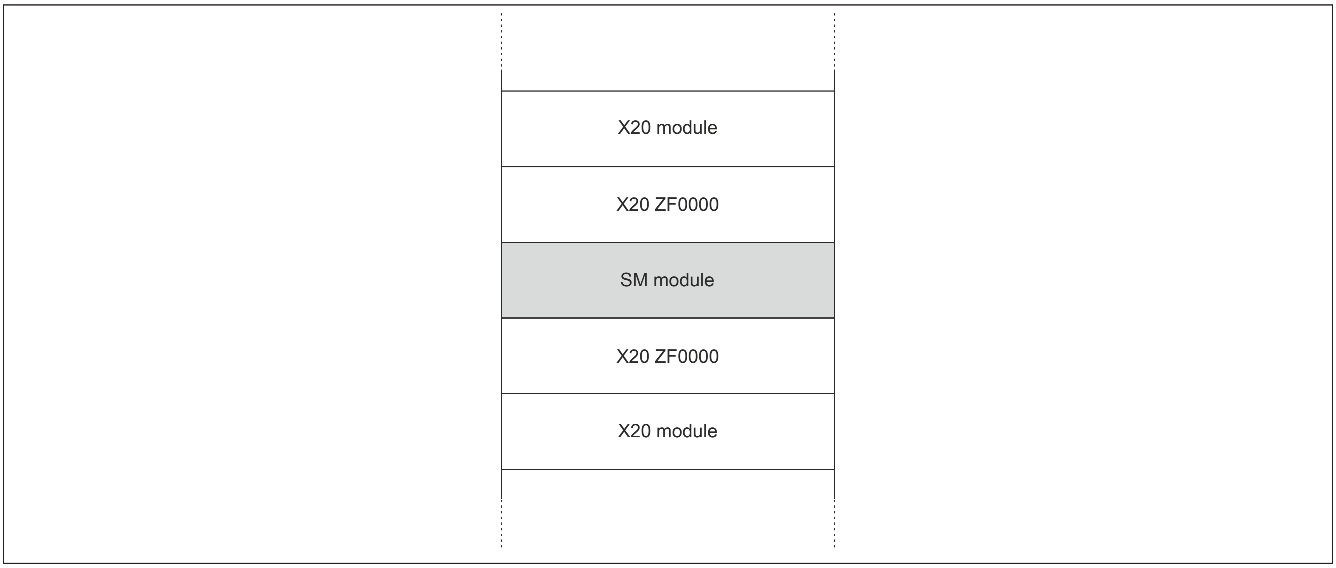
Horizontal mounting orientation

When operating the module up to 50°C in the horizontal mounting orientation, a dummy module must be connected on the right side.



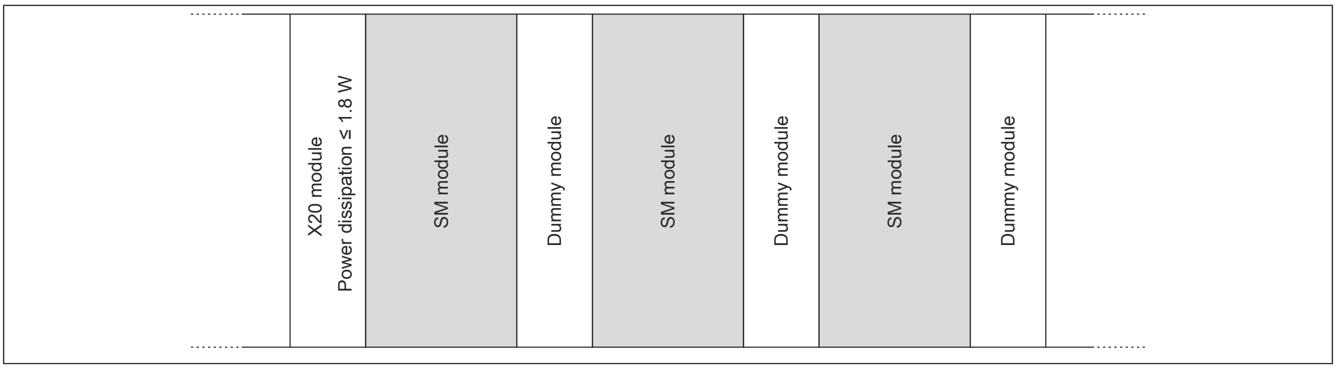
Vertical mounting orientation

When operating the module up to 40°C in the vertical mounting orientation, a dummy module must be connected on both sides.



Hardware configuration for multiple SM modules

If three or more SM modules are operated in a cluster, a dummy module must be inserted between the SM modules. There is no derating in this configuration.



3 Function description

3.1 Integrated motor detection

The stepper motor module can identify the connected motor by the coil characteristics. This makes it possible to detect not only wiring errors, but also incorrect motor types being used mistakenly.

After successful measurement, the time required to apply a current increase of ΔI = 1 A to a motor winding is returned in microseconds.

This depends on:

- Operating voltage
- Inductance and resistance of the motor winding

<table>
<tr><th colspan="3">Measurement procedure</th></tr>
<tr><td rowspan="3">1)</td><td colspan="2">To achieve reproducible results, the measurement must be made under the following defined conditions:</td></tr>
<tr><td>a)</td><td>Motor is at standstill.</td></tr>
<tr><td>b)</td><td>The motor must be in a half-step position (phase A fully powered, phase B not powered). This means the internal position counter on the SM module must have a value that fulfills the following conditions:
• Full steps are divisible by 4.
• Microsteps = 0</td></tr>
<tr><td>2)</td><td colspan="2">Condition 1b) is fulfilled after a the SM module is reset or switched on. Immediately afterwards, when the holding current is applied to the motor for the first time (at standstill), the duration for applying the current is measured. This is therefore a suitable time to read the motor identification register in the application.</td></tr>
<tr><td>3)</td><td colspan="2">The current range from approximately 1/3 of the nominal current up to the nominal current is used as operating range for determining the motor identifier.</td></tr>
</table>

Information:

The registers are described in "Motor identification" on page 52.

3.2 Currents independently configurable

Due to the individual adjustment of the coil currents, the motor is only operated with the current it actually needs. The required motor voltage is automatically set via the constant current control of the module and the provided winding resistance of one phase of the motor.

This simplifies the selection of the available motors and prevents unnecessary heating. Because this affects energy consumption and thermal load, the effects are positive on the service life of the complete system. Complete flexibility is achieved through the use of independently adjustable holding, maximum and nominal current values. The current for microsteps is automatically adjusted to the configured current values.

The holding current, nominal current and maximum current registers are used to configure the desired motor current. The nominal motor current is entered in the nominal current register according to the motor's data sheet.

Reasonable values:

- Holding current < Nominal current < Maximum current

Information:

The maximum current must always be configured greater than or equal to the nominal current.

Register	Description
Nominal current	Current during operation at constant speed
Maximum current	Current during short acceleration phases if a higher motor torque is required. In mode "Homing", the nominal current is always used instead of the maximum current, even in acceleration phases.
Holding current	Current consumption for phases in which less torque is required (e.g. at standstill). This reduces the amount of heat generated by the motor.

When the current changes to a weaker value (e.g. when transitioning from the acceleration phase to the constant speed mode), the stronger current is maintained for an additional 100 ms. The following priority applies regardless of the values actually set: Maximum current before nominal current before holding current.

Information:

The registers are described in "Configuring currents (function model 0)" on page 46 and "Configuring currents (function models 3 and 254)" on page 46.

3.3 Load-dependent current control

Load-dependent current control uses the motor load value to reduce the current for the motor when the motor is only slightly loaded. In addition to saving energy, this also reduces the noise level of the motor. The motor is also not heated as much. If a heavier load is again placed on the motor, the motor current is increased and thus more torque can be provided.

To set the load-dependent current control, the upper and lower limits of the motor load must be set so that they can be increased or decreased as needed.

Due to the dependence of the motor load value on motor-specific properties and application-specific load and speed requirements, the operating conditions should be matched to the actual application.

For the configuration, see Stall detection.

Information:

The registers are described in "Measuring the motor load" on page 52 and "Configuring current control" on page 42 .

3.4 Stall detection

To reliably detect a standstill of the motor, the stall threshold must be determined. The stall threshold should be a value within the operating limits and slightly higher than the minimum value before an actual motor standstill occurs.

- The SGT correction values are used to compensate for an offset that results from the back EMF of the motor. For this purpose, the maximum load that the motor can run without blocking must be determined. Ideally, the motor load value should drop to 0 before a step loss due to overload causes the motor to stop.
  If the SGT correction value is set so that a value of 0 is displayed at maximum motor load, then the stall is precisely detected and the stall error bit is set correctly.
  For some motors, however, stall detection is very difficult to use or cannot be used at all. With these motors, no settings can be determined where the motor stops without slipping, e.g. due to electrical parameters or too little magnetic field feedback from the motor.
- The SCT speed values set the threshold value from which speed the determined SCT correction value becomes active. These include:

Threshold value 1	From 0 to SGT_Speed01a
Threshold value 2	From SGT_Speed01b to maximum

  Since the motor load value drops to 0 during acceleration from standstill, this would be incorrectly detected as a stall and the motor would stop again immediately. A minimum speed can therefore be set at which stall detection is enabled.

Information:

- **At very low motor speeds (<1 revolution/second), stall detection cannot be carried out reliably due to low back EMF values.**
- **At very high motor speeds, at which the sinusoidal motor current can no longer be impressed into the motor coil, poor responsiveness can also occur.**

Information:

The registers are described in "Stall detection (function model 0)" on page 47 and "Stall detection (function models 3 and 254)" on page 47.

3.5 Homing

Homing can be carried out in forward and reverse direction.

Before homing can be performed, the motor must be at a standstill.

If the homing condition occurs, the motor stops and the values of the position counter that are valid at the moment when the homing condition occurs are applied as the homed zero position.

Whether homing should take place via low/high level on the digital input, stall or unconditionally must be set in the homing configuration.

Homing via digital input

Case 1: Active homing level not yet reached → Motor not yet at end position:

Movement continues at the homing speed in the homing direction until the active level for "Stop homing" is on the input.

Case 2: Active homing level already reached → Motor at end position:

Movement continues at the homing speed against the homing direction until the active level for "Stop homing" is no longer on the digital input. Movement continues at the homing speed in the homing direction until the active level for "Stop homing" is on the digital input again.

Homing during stall

Movement continues in the homing direction until a stall is detected. When a stall is detected, the value of the position counter is applied as the homed zero position within one millisecond. The motor is then stopped abruptly (not using the deceleration ramp). However, it can take up to 25 ms to stop the motor because the ramp generator runs with a configurable internal cycle of up to 25 ms.

In order to prevent unwanted homing due to brief stalling, an additional delay time can be defined for stall detection. Homing is only performed after the delay time has elapsed.

In this mode, the nominal current is always used instead of the maximum current, even in acceleration phases.

To test the responsiveness of this homing mode, the motor load value used for identifying a stall can be made visible in the status word.

Immediate homing

The current values of the position counter are immediately applied as the homed zero position (no motor movement).

> **Information:**
>
> **Registers are described in "Homing" on page 51.**

3.6 Limitations

Limitations can be implemented both in hardware using limit switches and in software using software limits.

Negative/Positive limit switch:

When one of the limit switches is reached, a warning is triggered and the speed is decelerated to 0. There is no state change of "Device control state machine". This keeps current flowing to the motor.

The error that occurred can be read from the error code register. Normal operation can be resumed through acknowledgment of the warning. This will not restrict motor movement to a specific direction and the limit switch will not be triggered until the next active edge.

Overshooting the limit switch while braking

The limit switches are not linked with the corresponding direction of movement. If the limit switch is traversed, another error will be triggered when reversing after acknowledging the initial error.

Direction monitoring

If this function is enabled, then the two limit switches will be linked with the respective direction of movement. This means that the negative limit switch is only triggered in the negative and the positive limit switch only in the positive direction of movement (specified direction).

This prevents specifying a movement in the incorrect direction when direction monitoring is enabled and limit switches are active.

Warning!

If the motor is wired incorrectly with this configuration (incorrect direction of movement), then the limit switch will not be triggered and the actual correct direction of movement will be denied. This will also be the case when the limit switch connections are reversed.

Monitoring software limits

This function is enabled if at least one of the two software limit registers is not equal to zero.

This limit monitoring is effective in all positioning modes. Position overflow is not possible when this function is enabled. Movement is always contained within the two limits.

If a position is specified that overshoots/undershoots the software limits or in the event of an invalid configuration (minimum > maximum), bit "Internal limit active" is set in the status word. The motor movement will be stopped until a position is specified within the limits.

Information:

Monitoring software limits works only when in connection with the following CANopen bus controllers:

- **X20BC0043-10**
- **X20BC0143-10**
- **X67BC4321-10**
- **X67BC4321.L08-10**
- **X67BC4321.L12-10**

Information:

The registers are described in "Limitations" on page 51.

3.7 Motion generator

Mode

The module can independently generate a number of different movements based on specified parameters:

- Position mode
- Velocity setpoint
- Homing

	Information
	No mode selected
Positioning	
	Velocity setpoint
	Position mode Depending on bit 0 in the general configuration, the position mode behaves as follows: • Without extended control word: Move to target position as soon as the target position is changed. • Position mode with extended control word: Move to the target position as described in "Extended control word".
Moving to absolute positions	
	Home position with residual distance
	Moving to the target position if digital input is set
	Moving to a fixed position depending on digital input
	Moving to a fixed position (first or second position)
Homing	
	Homing in the forward direction
	Homing in the reverse direction
Applying positions	
	Applying the actual position (homing or actual position)

Information:

For all modes: Bit "Target reached" is set in when the current action is ended (i.e. when the position or speed is reached, depending on the mode).

A new position or speed can be specified even before the current action is finished.

Velocity setpoint

The desired velocity setpoint is specified to the module (microsteps per cycle).

Observing the maximum permissible acceleration, the motor moves with a ramp to the desired speed set-point and maintains this speed until a new speed setpoint is specified.

Position mode

A position setpoint is specified. The motor is then moved to this new position. This is done with a ramp function that accounts for the defined maximum speed and acceleration values.

The position setpoint can also be changed during an active positioning procedure.

The position setpoint is specified in microsteps (1/256 of a full step).

How the position is applied can be controlled in the configuration by bit 0:

- If bit 0 equals 0 (no extended control word), the position setpoint will be applied as soon as it is not equal to the current position. The new position is then used for the movement.
- If bit 0 equals 1, the position setpoint is accepted as described in "Extended control word".

Extended position mode

Position mode with extended control word behaves like the previously described position mode (without the extended control word), but the new position setpoint is applied according to the extended control word.

Extended control word

Additional commands can be sent depending on the state of the module:

- (Do not) Apply the new target position.
- Process or interrupt the current positioning and start the next positioning.
- Define the target position as an absolute or relative value.
- Stopping a movement

Relative position setting

If bit abs / rel in extended control word is set, then the target position is interpreted as a relative value. At each New setpoint trigger, the target position will be increased by this value (or decreased if the value is negative).

If the mode changes between the position settings, relative movement will then proceed starting at the last specified position. The position setpoint mode is initialized with 0 when the module is started.

Specifying the target position

The target position can be defined in 2 different ways:

Type of setpoint definition	Description
Single setpoint	After the target position is reached, bit Target reached is set in the status word. A new target position (setpoint) is then defined. The drive stops at each target position before starting the movement to the next target position.
Set of setpoints	After the target position has been reached, the movement to the next target position is started immediately without stopping the drive. It is therefore possible to initiate a new positioning by specifying another target position during active positioning.

The two options "Single setpoint" and "Set of setpoints" are controlled by the timing of bits New setpoint and Change set immediately in the extended control word and Setpoint acknowledge in the extended status word register.

These bits can be used to create a Request-Response mechanism. This makes it possible to specify a target position while a previous position specification is still being processed.

Specifying the target position

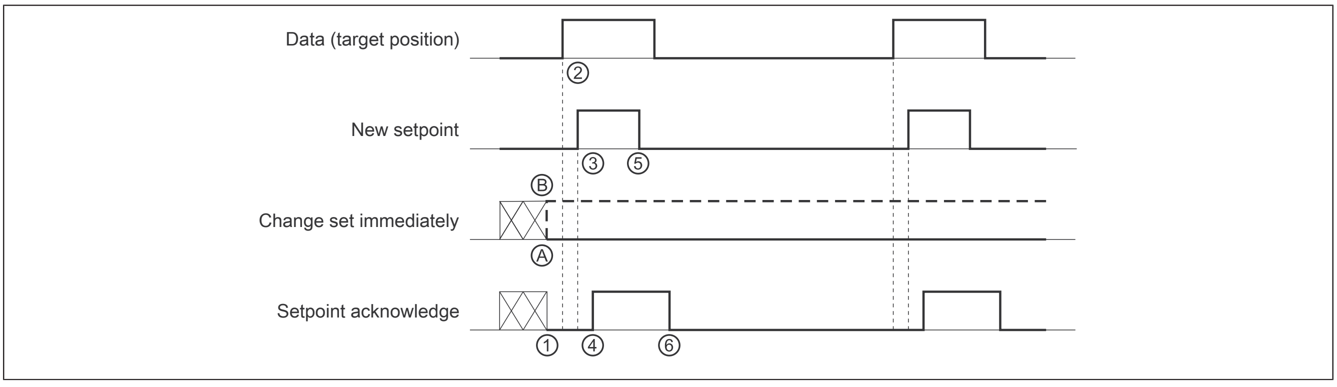


Figure 2: Principle for applying the setpoint

Transferring a new setpoint:

1) If bit Setpoint acknowledge in the extended status word is 0, the module will accept a new target position.
2) Specify the new target position.
3) A rising edge on bit New setpoint in the extended control word indicates that the new target position is valid and can be used for the next positioning movement.
4) After the module has received and saved the new target position, bit Setpoint acknowledge is set to 1 in register Status word.
5) Now the controller can reset the New setpoint bit to 0.
6) Then the module resets bit Setpoint acknowledge to 0 to signal when a new target position is accepted.

Position specification "Single setpoint"

If bit Change set immediately is set to 0 (Ⓐ in figure "Principle for applying the setpoint"), then the module is operating with position specification Single setpoint. This mechanism results in a speed of 0 when the motor reaches target position x_1 at time t_1. After the controller has been notified that the setpoint has been reached, the next target position x_2 will be processed at time t_2 and reached at t_3.

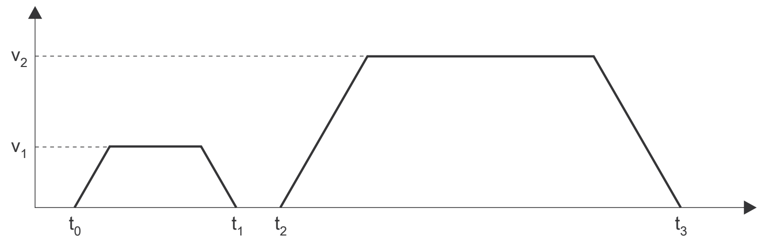


Figure 3: Ramp characteristics in mode Single setpoint

Position specification "Set of setpoints"

If bit Change set immediately is set to 1 (Ⓑ in figure "Principle for applying the setpoint"), then the module is operating with position specification Set of setpoints. This means that the module receives the first target position at t_0. A second target position is received at time t_1. The drive immediately adapts the current movement to the new target position.

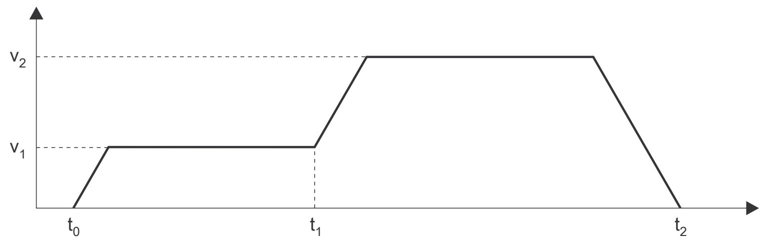


Figure 4: Ramp characteristics in mode Set of setpoints

Home position with residual distance

In the event of a rising or falling edge on digital input 1, the current target position is discarded and only a set number of steps is moved forward or backward.

Negative values are also permitted for the set offset.

A new target position is no longer accepted after the trigger event. There must first be a switch made to mode 0 and then back to mode -121.

Bit "Target reached" in the status word is only set to 1 when the end position (after the trigger event) is reached.

The homing configuration defines whether a rising or falling edge of the digital input is used as a trigger.

The reversing loop is not active in this mode. Any configured values are ignored.

Moving to the target position

A preset position setpoint is approached on a rising edge of digital input 1.

A changed position setpoint is only applied on a new rising edge of the associated digital input. This can also take place during the ongoing positioning process and is then immediately effective.

In addition, a reversing loop can be set.

Moving to a fixed position

2 fixed positions can be transferred acyclically to the module that are approached differently depending on the mode.

- Mode -124:
  "1" on digital input 1 moves to the first fixed position.
  "0" on digital input 1 moves to the second fixed position.
  Switching is possible during an active positioning procedure.
- Mode -125: Move to the first fixed position
- Mode -126: Move to the second fixed position

In addition, a reversing loop can be set.

Applying the actual position

Before a position can be applied, the motor must be at a standstill and physically located at the point for which the position being set should be applied. The new position is applied when the state machine is in state "Operation enable".

Depending on the mode, the following positions can be applied:

- The set target position is applied as the current actual position in the internal position counter.
- The current actual position is modified such that the specified position exists at the reference. If moved to this position, the motor is at the home position. The home position in the referenced position register is also set to this value. Before this mode is called, the motor must be at a standstill and the home position must have been determined using the positive/negative homing mode.

Reversing loop

This parameter is only available for absolute positioning.

A reversing loop can be used to avoid mechanical backlash and different movement tolerances.

If the value is not equal to 0, the target position is approached directly when coming from one direction; when coming from the other direction, the target position is initially overshot by the configured number of steps before finally moving to the target position. The target position is therefore always approached from the same direction. The sign of the defined value determines the direction in which the reversing loop runs.

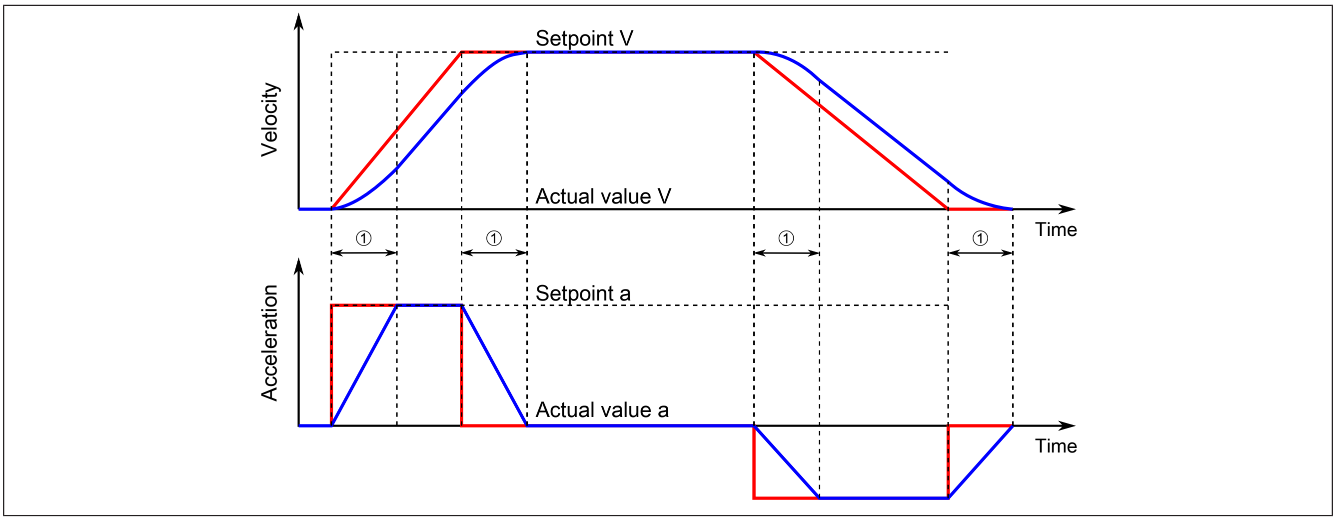
- Positive sign: Forward movement
- Negative sign: Backward movement

Jerk limitation

To avoid noticeable jumps when starting the motor or when the acceleration changes (e.g. from 1 m/s^2 to 3 m/s^2), a jerk time (①) can be defined. This indicates the number of cycles during which the acceleration is adapted to the new setpoint. If more than 80 cycles are entered, they are limited to 80.

Changes made to the jerk time while the motor is running are only applied when the preset position is reached or the next motor standstill.

The following diagram shows the change of acceleration and speed with (blue line) and without (red line) jerk limitation time.



① Set jerk time in cycles

3.8 Counters

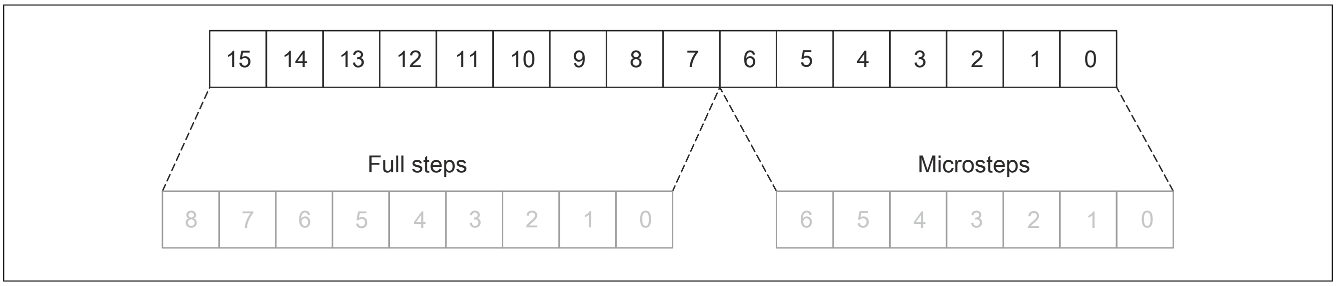
The module has 2 different modes that can be used to determine the position.

Internal position counter

The internal position counter is the position calculated by the module (position setpoint). This is a cyclic 16-bit counter.

The lowest 5 to 8 bits represent microsteps, and the higher bits represent full steps. In function model Standard with SDC, this value is defined as "8-bit microstep" and cannot be changed.

Example of the internal position counter format (7-bit microsteps, i.e. set bit 5 and 6 of the module configuration to binary 10):



ABR counter

This counter is a cyclic 16-bit counter. The relationship between this counter and the internal position counter depends on the resolution of the ABR encoder and the microsteps defined for the internal position counter.

To evaluate the counter values, latch events can be defined where the counter values are frozen and transferred to specific registers.

Information:

The registers are described in "Counter configuration" on page 47.

3.9 ABR interface and digital inputs

The stepper motor module has a 5 V differential ABR interface in which the inputs R+ and R- can also be used as digital inputs.

The desired pinout can be configured in Automation Studio. The ABR interface is set by default.

Function models "Standard" and "Standard with SDC"

Channel	Function			
A+ / A-	A	A	A	A
B+ / B-	B	B	B	B
R+ / DI1	R	Digital input	Latch input	Quick stop
R- /DI2		Digital input	Trigger input	Enable

Function model "Ramp"

Channel	Function			
A+ / A-	A	A	A	
B+ / B-	B	B	B	
R+ / DI1	R	Digital input	Negative limit switch	
R- /DI2		Digital input	Positive limit switch	

Open-circuit detection

The 5 V differential ABR interface is equipped with open-circuit detection. When using inputs R+ and R- as 24 V digital inputs, open-circuit detection is not possible.

Information:

An open circuit is detected when the differential voltage of an ABR input undershoots 350 mV.

Information:

Registers are described in "Module configuration" on page 44.

3.10 Quick stop and enable

Digital inputs 1 and 2 can be assigned special functions for high-speed motor control.

Quick stop

Digital input 1 can be used as a quick stop. The input is edge-sensitive, i.e. the quick stop process is triggered when the input signal changes from inactive to active or vice versa (configurable, see "CounterConfig2" on page 45).

- QuickStop = 0. The motor is operated in normal motor mode.
- QuickStop = 1. The motor is stopped with the set braking ramp and then energized with holding current.

Example for calculating a braking ramp

The following configuration is used:

X2X cycle time:	800 µs
Motor steps:	300 microsteps per X2X cycle
Deceleration ramp:	1 microstep / 800 µs²

1. Convert acceleration from microsteps/cycle² to steps/seconds² (1 step = 256 microsteps).

$$\frac{\text{Steps}}{\text{Microsteps} * \text{Cycle time}^2} = \frac{1\text{step}}{256\text{microsteps} * 0.0008^2 \mu s} = 6103 \text{steps per second}^2$$

2. Calculate the speed.

$$\frac{\text{Motor steps}}{\text{Microsteps} * \text{Cycle time}} * \frac{300\text{steps}}{256\text{microsteps} * 0.0008 \mu s} = 1464 \text{steps per second}$$

3. Calculate the deceleration ramp.

$$\text{Brake ramp} = \frac{\text{Speed}}{\text{Acceleration}} = \frac{1464}{6103} = 0.24 \text{seconds}$$

Enable

Digital input 2 can be used as an enable input.

- Enable = 0. The motor is operated in normal motor mode.
- Enable = 1. The motor is not energized and is freewheeling.

Information:

Registers are described in "Configuration of inputs" on page 45 and "Deceleration ramp" on page 47.

3.11 Automatic shutdown

To prevent damage to the module or motor, both the voltage of the module power supply and the module temperature are monitored. In addition, the communication of the module with the controller can be monitored.

3.11.1 Motor shutdown in the event of overvoltage

The module power supply voltage is monitored. Its status can be read. An error is reported in the event of a voltage greater or less than the limit values.

If the supply voltage in the module rises above the limit value, e.g. due to regenerative operation, or falls below the limit value, then the motor output is cut off.

If the supply voltage is within the permissible range again, the error must first be acknowledged. The output stage can then be switched on again.

Limit values for the supply voltage

	Drive cut off	Drive switched back on
Lower limit	≤16.5 V	≥18 V
Upper limit	≥61 V	≤58 V

Information:

The registers for the error message and acknowledgment are described in "Input counter value" on page 55 and "Error acknowledgment" on page 57.

3.11.2 Overtemperature shutdown

If the module temperature reaches or overshoots the limit value, the module performs the following actions:

- Sets the "overtemperature" error bit
- The outputs are cut off.

As soon as the temperature is reduced below the module temperature limit value again, the error must first be acknowledged so that the channels can be switched on again.

Module temperature limit value
91°C

Information:

The registers for the error message and acknowledgment are described in "Error status" on page 56 and "Error acknowledgment" on page 57.

3.11.3 Monitoring the module communication

The communication between the module and controller can be monitored with SDC life sign monitoring.

After life sign monitoring is enabled, counter and timestamp information is exchanged and evaluated between the module and controller. If bit "SDC information" is additionally enabled, bit "EncOK01" is displayed in the Automation Studio I/O mapping. This bit is permanently linked to bit ModulOK and always indicates its value.

If the NetTime timestamp specified by the controller is in the past, then an error is triggered for the motor axis (only when the motor is switched on). The module performs the following steps:

1) The controller is informed of the error using the drive bit (DrvOk) = 0.

2) Braking at the configured nominal current with speed setpoint = 0

3) Wait for the configured switch-off delay to expire

4) Switch off the motor current

When the timestamp is back within the valid range, the motor can be started up again by a rising edge on bit DriveEnable.

Information:

The registers for the configuration are described in "SDC configuration" on page 45 and "Motor current" on page 53.

3.12 NetTime Technology

NetTime refers to the ability to precisely synchronize and transfer system times between individual components of the controller or network (controller, I/O modules, X2X Link, POWERLINK, etc.).

This allows the moment that events occur to be determined system-wide with microsecond precision. Upcoming events can also be executed precisely at a specified moment.

NetTime TECHNOLOGY

3.12.1 Time information

Various time information is available in the controller or on the network:

- System time (on the PLC, Automation PC, etc.)
- X2X Link time (for each X2X Link network)
- POWERLINK time (for each POWERLINK network)
- Time data points of I/O modules

The NetTime is based on 32-bit counters, which are increased with microsecond resolution. The sign of the time information changes after 35 min, 47 s, 483 ms and 648 µs; an overflow occurs after 71 min, 34 s, 967 ms and 296 µs.

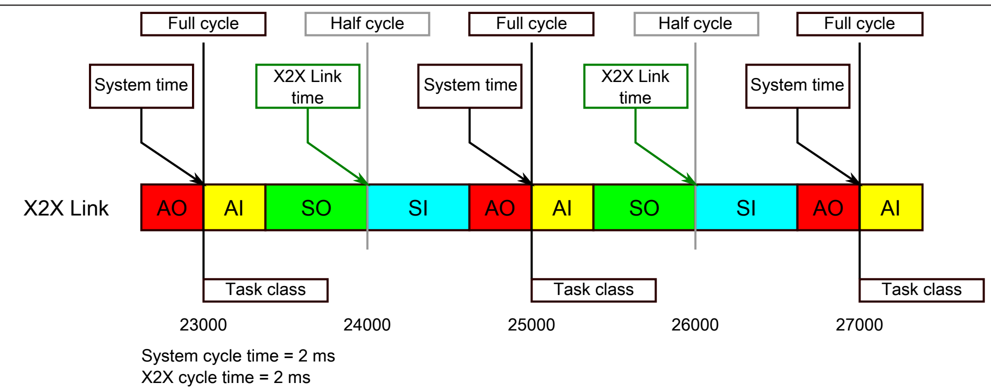
The initialization of the times is based on the system time during the startup of the X2X Link, the I/O modules or the POWERLINK interface.

Current time information in the application can also be determined via library AsIOTime.

3.12.1.1 Controller data points

The NetTime I/O data points of the controller are latched to each system clock and made available.

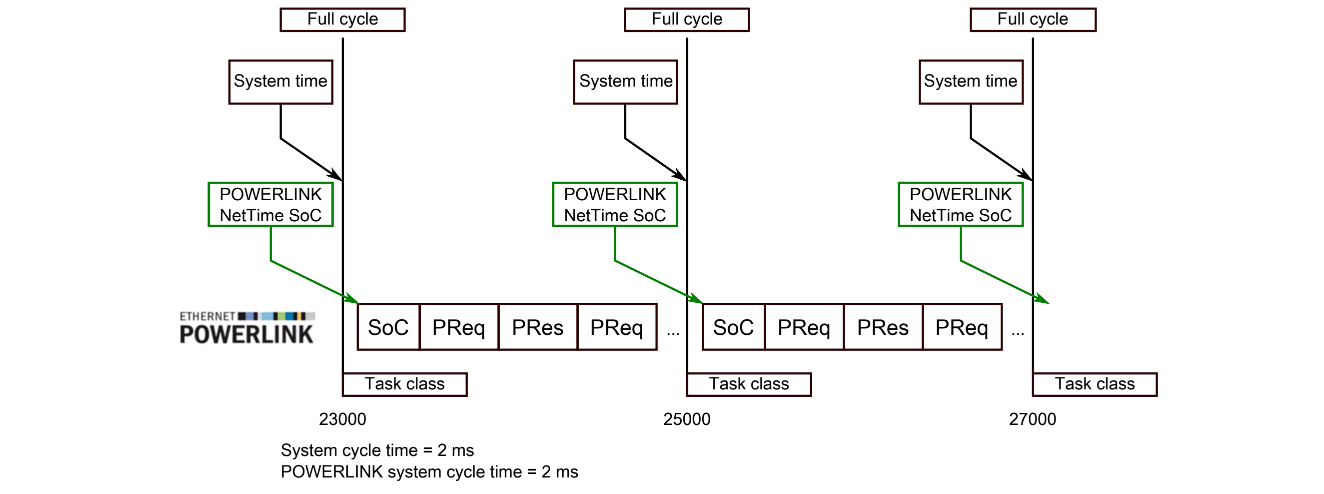
3.12.1.2 X2X Link - Reference time point



The reference time point on the X2X Link network is always calculated at the half cycle of the X2X Link cycle. This results in a difference between the system time and the X2X Link reference time point when the reference time is read out.

In the example above, this results in a difference of 1 ms, i.e. if the system time and X2X Link reference time are compared at time 25000 in the task, then the system time returns the value 25000 and the X2X Link reference time returns the value 24000.

3.12.1.3 POWERLINK - Reference time point

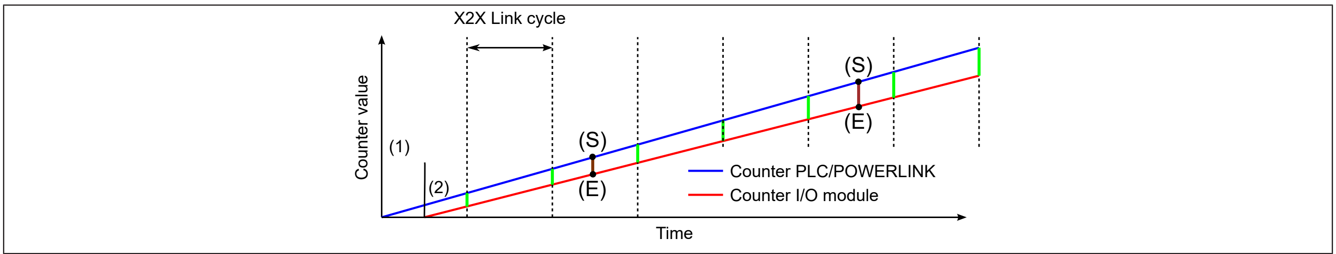


The POWERLINK reference time point is always calculated at the start of cycle (SoC) of the POWERLINK network. The SoC starts 20 µs after the system clock due to the system. This results in the following difference between the system time and the POWERLINK reference time:

POWERLINK reference time = System time - POWERLINK cycle time + 20 µs

In the example above, this means a difference of 1980 µs, i.e. if the system time and POWERLINK reference time are compared at time 25000 in the task, then the system time returns the value 25000 and the POWERLINK reference time returns the value 23020.

3.12.1.4 Synchronization of system time/POWERLINK time and I/O module



At startup, the internal counters for the controller/POWERLINK (1) and the I/O module (2) start at different times and increase the values with microsecond resolution.

At the beginning of each X2X Link cycle, the controller or POWERLINK network sends time information to the I/O module. The I/O module compares this time information with the module's internal time and forms a difference (green line) between the two times and stores it.

When a NetTime event (E) occurs, the internal module time is read out and corrected with the stored difference value (brown line). This means that the exact system moment (S) of an event can always be determined, even if the counters are not absolutely synchronous.

Note

The deviation from the clock signal is strongly exaggerated in the picture as a red line.

3.12.2 Timestamp functions

NetTime-capable modules provide various timestamp functions depending on the scope of functions. If a timestamp event occurs, the module immediately saves the current NetTime. After the respective data is transferred to the controller, including this precise moment, the controller can then evaluate the data using its own NetTime (or system time), if necessary.
For details, see the respective module documentation.

3.12.2.1 Time-based inputs

NetTime Technology can be used to determine the exact moment of a rising edge at an input. The rising and falling edges can also be detected and the duration between 2 events can be determined.

Information:

The determined moment always lies in the past.

3.12.2.2 Time-based outputs

NetTime Technology can be used to specify the exact moment of a rising edge on an output. The rising and falling edges can also be specified and a pulse pattern generated from them.

Information:

The specified time must always be in the future, and the set X2X Link cycle time must be taken into account for the definition of the moment.

3.12.2.3 Time-based measurements

NetTime Technology can be used to determine the exact moment of a measurement that has taken place. Both the starting and end moment of the measurement can be transmitted.

4 Commissioning

4.1 CAN I/O bus controller

The module occupies 2 analog logical slots on CAN I/O.

4.1.1 Using the module on the bus controller

Function model 254 "Bus controller" is used by default only by non-configurable bus controllers. All other bus controllers can use other registers and functions depending on the fieldbus used.

For detailed information, see section "Additional information - Using I/O modules on the bus controller" in the X20 user's manual (version 3.50 or later).

4.2 Setting full step limit values

A rotational speed is configured with the Full step limit value. When the defined speed has been reached, the drive will automatically change from microstep to full step mode. This makes it possible to optimize the torque at higher speeds, while microstep mode ensures optimal radial runout at lower speeds.

It does not make sense to switch to full step mode at a standstill since fine positioning would then no longer be possible. For this reason, the value 0 is interpreted as deactivating full step mode, i.e. the motor is always operated in microstep mode.

Example

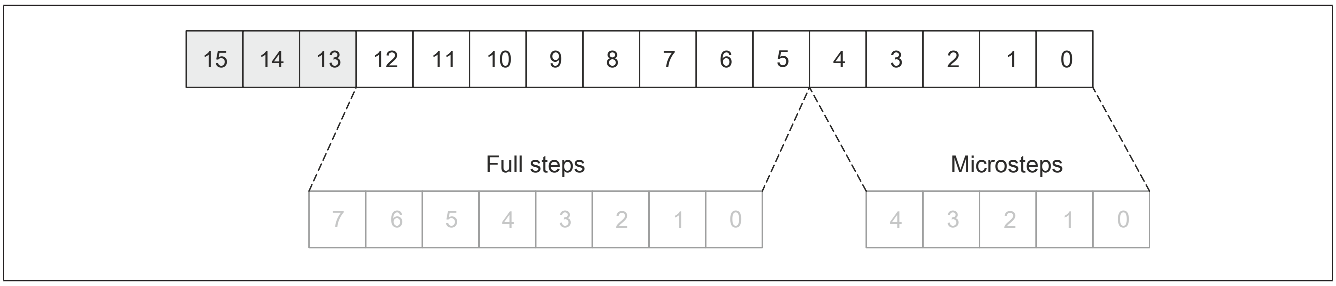
Microstep mode should change to full step mode at 500 steps/second. A Full step limit value of 500 for a motor with 200 steps/revolution corresponds to a speed of:

$$T^{-1} = \frac{500\ steps / second}{200\ steps / rotation} = 2.5\ \frac{rotations}{second} = 150\ min^{-1}$$

Setting the microsteps

Depending on the required resolution and maximum configurable speed, bits 5 and 6 of the module configuration can be used to set the bit position at which the 1s position of the full steps starts.

Example for 5-bit microsteps, i.e. bits 5 and 6 of the module configuration are set to binary 00:



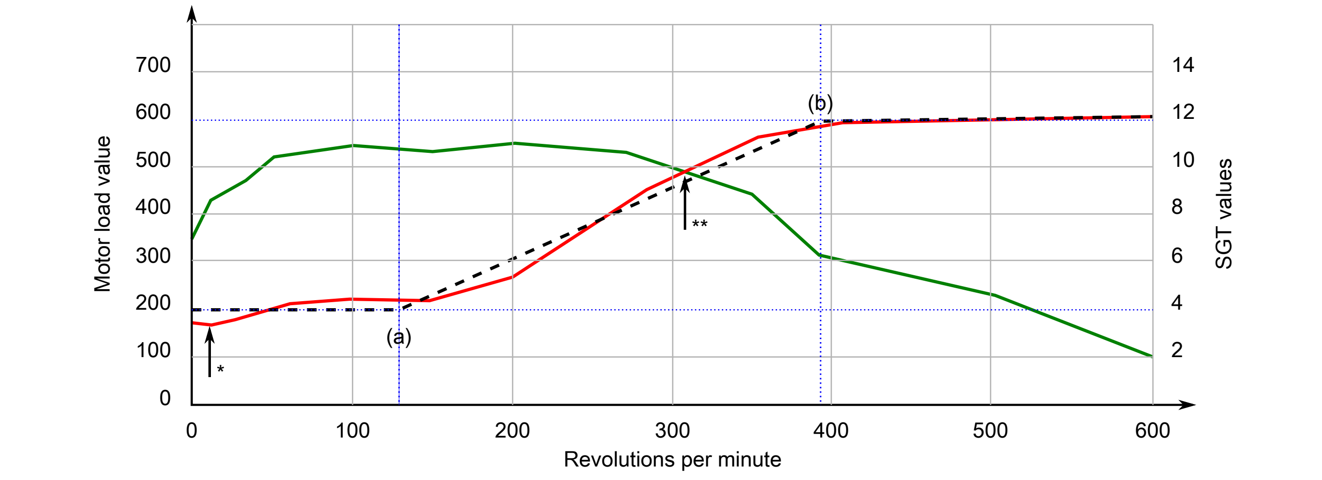
(For the meaning of bits 13 to 15, see "Motor settings" on page 53.)

4.3 Configuring the motor load curve

Creating a motor load characteristic curve is helpful to set the points (SGT_Value01a and SGT_Value01b) for the offset of the motor load value.

In the following example, assuming 200 steps per revolution results in the following values:

	SGT_Value	SGT_Speed
Value (a)	4	129 revolutions/minute = 430 steps/second
Value (b)	12	392 revolutions/minute = 1307 steps/second



Legend

Green line	Motor load characteristic curve without load
Red line	Offset-corrected motor load characteristic curve
Black line	Interpolated motor load characteristic curve
*	Minimum speed for stall detection (>10 rpm)
**	Back EMF reaches supply voltage

Since the back EMF increases with increasing speed (motor load value decreases to 0) and counteracts, there is a maximum speed for stall detection.

Information:

The registers are described in "Motor identification" on page 52 .

4.4 Configuring the SCT values / stall detection

4.4.1 Stall detection for variable speeds

If stall detection is needed for a range of speeds (SGT_Speed01a and SGT_Speed01b), the module can be configured as follows:

1) Set the initial values.
    - Disable current control. (CurrentControlEnabel01 = 0)
    - Disable stall detection. (StallDetectMinSpeed01 = 65535)
    - Set register SGT_Speed01x to 0.
    - Set register SGT_Value01x to 0.
2) Operate the motor at the start of the speed range (SGT_Speed01a) without load and observe the motor load value.
3) Adjust the value.
    a) Slowly increase the load on the motor. If the motor stops before the motor load value indicates 0, reduce SGT_Value01x by 1.
    b) If the motor load value is 0 before the motor stops, increase SGT_Value01x by 1.
    The optimal setting is achieved when the motor load value remains above 0 at the maximum load necessary. If the value becomes 0, the maximum load on the motor is exceeded. In this case, bit 0 "StallError" is set in register Error status.
4) Operate the motor at the end of the speed range (SGT_Speed01b) without load and observe the motor load value.
5) Adjust the values, see item 3).
6) Set the determined range values.
    The SGT_Value0x values between the range start and range end are interpolated linearly.
    - Set SGT_Value01a and SGT_Speed01a for the range start with the determined values.
    - Set SGT_Value01b and SGT_Speed01b for the range end with the determined values.
7) In register StallDetectMinSpeed01, set the speed value from which stall detection is enabled.

4.4.2 Stall detection for constant speed

If stall detection is only needed for a constant speed, the module can be configured as follows:

1) Set the initial values.
    - Disable current control. (CurrentControlEnabel01 = 0)
    - Disable stall detection. (StallDetectMinSpeed01 = 65535)
    - Set register SGT_Speed01x to 0.
    - Set register SGT_Value01x to 0.
2) Operate the motor at the speed needed for the application without load and observe the motor load value.
3) Adjust the value.
    a) Slowly increase the load on the motor. If the motor stops before the motor load value indicates 0, reduce SGT_Value01x by 1.

    Information:

    Registers SGT_Value01a and SGT_Value01b must always be set to the same value!

    b) If the motor load value is 0 before the motor stops, increase SGT_Value01x by 1.
    The optimal setting is achieved when the motor load value remains above 0 at the maximum load necessary. If the value becomes 0, the maximum load on the motor is exceeded. In this case, bit 0 "StallError" is set in register Error status.
4) In register StallDetectMinSpeed01, set the speed value from which stall detection is enabled.

4.5 Operating function model "Ramp"

Commands for controlling the module are written to the "Control word" on page 65. In register "Status word" on page 66, the current state of the module is reported. The function mode (absolute position, constant speed, homing, etc.) is set in the "mode register" on page 62.

4.5.1 Structure of the control word

Control word bits and their state for the commands of the state machine:

Command	Stall detection	Encoder position sync/async	Current error detection	Warning reset	Motor ID trigger	Reserved	CurrentControlEnable	Stop	Fault reset	Mode-specific	Mode-specific	Mode-specific	Enable operation	Quick stop	Enable voltage	Switch on
Bit[1]	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Shutdown	x	x	x	x	x	0	0	x	0	x	x	x	x	1	1	0
Switch on	x	x	x	x	x	0	0	x	0	x	x	x	0	1	1	1
Disable voltage	x	x	x	x	x	0	0	x	0	x	x	x	x	x	0	x
Quick stop	x	x	x	x	x	0	0	x	0	x	x	x	x	0	1	x
Disable operation	x	x	x	x	x	0	0	x	0	x	x	x	0	1	1	1
Enable operation	x	x	x	x	x	0	0	x	0	x	x	x	1	1	1	1
Fault reset	x	x	x	x	x	0	0	x	↑	x	x	x	x	x	x	x

1) x ... Any, ↑ ... Rising edge

Bits 0, 1, 2, 3 and 7 (light gray in the previous table)	These bits control the state of the State machine according to the commands in the table above.
Fault reset	A rising edge resets errors and warnings (see "State machine" on page 36)
Stop	0 ... Perform motor movement 1 ... Stop axis with deceleration This bit is only evaluated when the extended control word is enabled in register "General configuration" on page 48.
CurrentControlEnable	0 ... Load-dependent current control disabled 1 ... Load-dependent current control enabled
Motor ID trigger	A rising edge enables the motor ID measurement.
Warning reset	A rising edge resets warnings (no effect on errors, which are reset using "Fault reset"; the state machine is not affected by this bit).
Current error detection	0 ... Current error detection disabled 1 ... Current error detection enabled
Encoder position sync/async	0 ... Value of the ABR counter on register "Current position (acyclic)" on page 68. Internal position counter of the ramp generator on register "Current position (cyclic)". 1 ... Value of the ABR counter on register "Current position (cyclic)" on page 68. Internal position counter of the ramp generator on register "Current position (acyclic)".
Stall detection	0 ... Stall detection disabled 1 ... Stall detection enabled

4.5.2 Structure of the status word

The individual bits of this register and its states depend on the current state of the state machine:

Status	Reserved	Reserved	Reserved	Reserved	Int. limit active	Target reached	Remote	Reserved	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Not ready to switch on	x	x	x	x	x	x	1	0	x	0	x	0	0	0	0	0
Switch-on disabled	x	x	x	x	x	x	1	0	x	1	x	0	0	0	0	0
Ready to switch on	x	x	x	x	x	x	1	0	x	0	1	0	0	0	0	1
Switched on	x	x	x	x	x	x	1	0	x	0	1	1	0	0	1	1
Operation enable	x	x	x	x	x	x	1	0	x	0	1	1	0	1	1	1
Quick stop active	x	x	x	x	x	x	1	0	x	0	0	1	0	1	1	1
Fault reaction active	x	x	x	x	x	x	1	0	x	0	x	0	1	1	1	1
Fault	x	x	x	x	x	x	1	0	x	0	x	0	1	0	0	0

Information about the status word:

Bits 0, 1, 2, 3, 5 and 6 (light gray in the previous table)	These bits are set according to the current state of the State machine.	
Voltage enabled	Becomes 1 as soon as the motor is powered	
Warning	Becomes 1 if a warning is detected ("Overcurrent", "Undercurrent"). The type of warning is indicated in register "Error code" on page 67. The highest priority error / warning is shown in each case, with the priority corresponding to the order in the respective table. Warnings can be reset with a rising edge on the "Warning reset" bit in the control word.	
Remote	Always 1	
Target reached[1)], depends on bit 8 (Stop) in Structure of the control word	**If stop = 0**	**If stop = 1**
	In modes 1, -123, -124, -125 and -126 (absolute positioning): 0...Positioning begins 1...Target has been reached **In mode 2 (constant speed):** 0...Motor accelerates/brakes 1...Speed setpoint reached **In modes -127 and -128 (homing):** 0...Homing started 1...Homing ended **In mode -122 (set actual position):** The bit briefly becomes 0 and immediately becomes 1 again as soon as the position is set.	**In all modes:** 0...Axis decelerating 1...Axis speed = 0
Internal limit active	0 ... No limit violation 1 ... Internal limit is active (upper/lower software limit violated)	

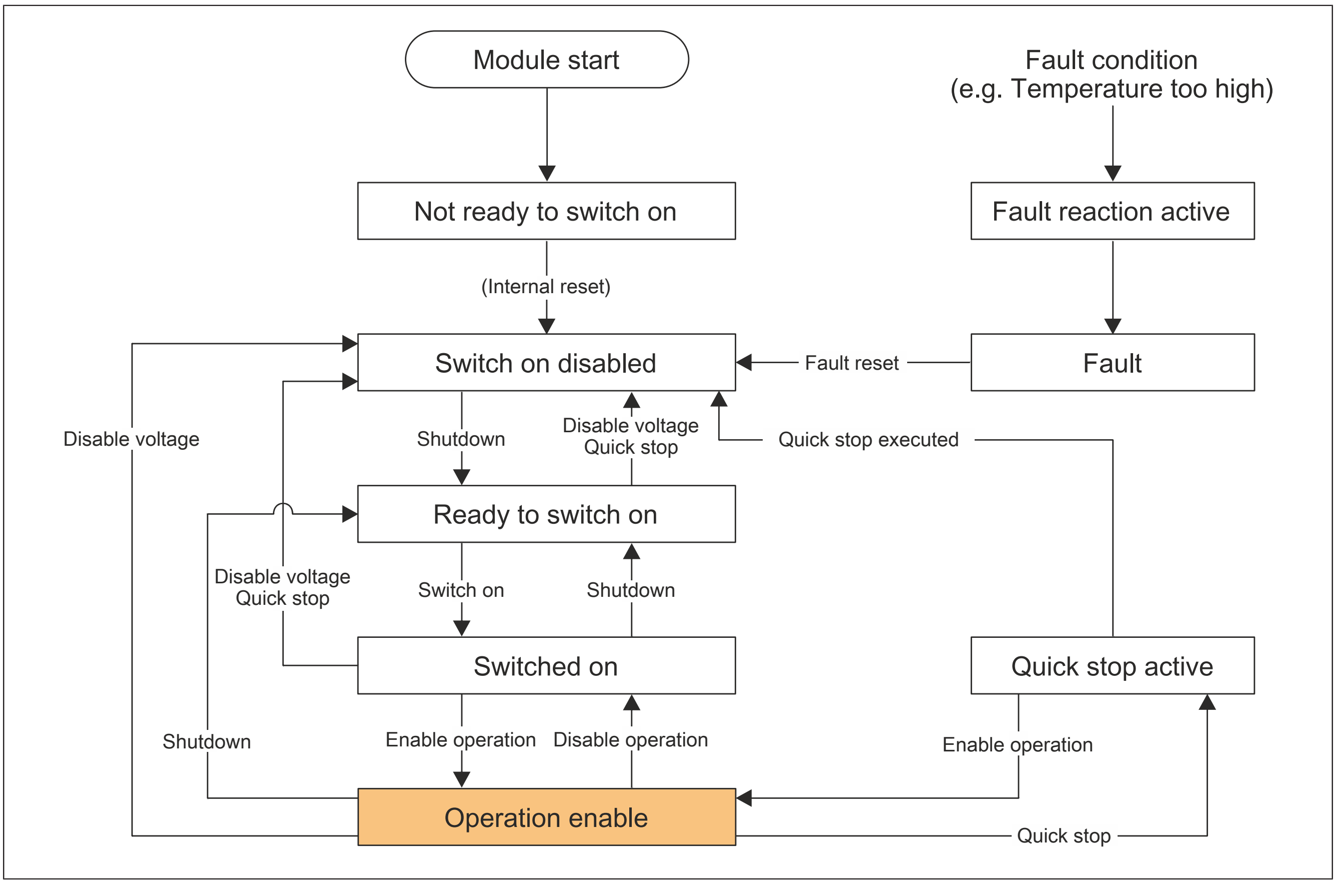
1) If the extended control word in register "General configuration" on page 48 was not enabled, "Target reached" behaves the same as if "Stop = 0".

4.5.3 State machine

Control of the motor takes place according to the state machine illustrated below. After the module is started, the state machine automatically changes to state "Not ready to switch on". The application then operates the state machine by writing commands to the "Control word" on page 65.

The state machine successively reaches states "Ready to switch on", "Switched on" and "Operation enable" by writing the consecutive commands "Shutdown", "Switch on" and "Enable operation".

> **Information:**
>
> **Motor movements are performed according to the setting in register "Mode" on page 62 only in state** "Operation enable"**.**



State change	Description
Not ready to switch on → Switch on disabled	This state change occurs automatically after starting the module and internal initialization has taken place.
Switch on disabled → Ready to switch on	This state change is initiated by command Shutdown. No others actions are performed.
Ready to switch on → Switch on disabled	This state change is initiated by command Disable voltage or Quick stop. No others actions are performed.
Switched on → Switch on disabled	This state change is initiated by command Disable voltage or Quick stop. The motor voltage is switched off immediately.
Ready to switch on → Switched on	This state change is initiated by command Switch on. The motor voltage is switched on. When this state change occurs for the first time since the module is started, the motor ID measurement is performed before state Switched on is achieved. This can take approximately 1 second.
Switched on → Ready to switch on	This state change is initiated by command Shutdown. The motor voltage is switched off immediately.
Switched on → Operation enable	This state change is initiated by command Enable operation. Motor movements are now performed depending on the defined mode.
Operation enable → Switched on	This state change is initiated by command Disable operation. If in motion, the motor is decelerated with the configured deceleration. Motor voltage remains on in state Switched on.
Operation enable → Ready to switch on	This state change is initiated by command Shutdown. The motor voltage is switched off immediately.
Operation enable → Switch on disabled	This state change is initiated by command Disable voltage. Motor voltage switched off. It is strongly recommended to only make this state change on a stopped motor since regeneration on a motor running at no load can cause an overvoltage error on the DC bus (0x3210).
Operation enable → Quick stop active	This state change is initiated by command Quick stop. If in motion, the motor is decelerated with the configured deceleration. During the deceleration, the state machine remains in state Quick stop active. If the motor comes to standstill, the switch to state Switch on disabled takes place automatically. While the state machine is in state Quick stop active, command Enable operation can be used to switch back to state Operation enable.
→ Fault reaction active	This state change is brought on when an error occurs and cannot be triggered by a command from the user. It can be triggered by an error type classified as "Error" (see "Error code" on page 67). (Other error types listed as a "warning" only cause bit "Warning" to be set in the status word and do not cause a state change in the state machine.) The motor voltage is cut off, and the state machine then changes immediately to state Fault. The error type is listed in the error code register (see table in "Error code" on page 67). The highest priority error is shown. The priority corresponds to the order in the error code table.
Fault → Switch on disabled	This state change is initiated by command Fault reset. However, the state only changes if no more errors are present when the command is written. All errors and warnings are reset. The error code register contains 0 or the warning code if a warning is still present.

5 Register description

5.1 mapp Motion system requirements

This module can be operated with mapp Motion function blocks. The following minimum versions are required for this:

- Automation Studio: 4.7.2
- Automation Runtime: 4.72
- mapp Technology package: mapp Motion 5.13

5.2 General data points

In addition to the registers described in the register description, the module has additional general data points. These are not module-specific but contain general information such as serial number and hardware variant.

General data points are described in section "Additional information - General data points" in the X20 System user's manual.

5.3 Function model 0 - Standard

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Configuration						
Module / Motor configuration						
46	ConfigOutput02 (module configuration 1)	UINT				•
96	CounterConfig2	UINT				•
Configure currents						
33	ConfigOutput03 (holding current)	USINT				•
34	ConfigOutput04 (nominal current)	USINT				•
35	ConfigOutput05 (maximum current)	USINT				•
Counter configuration						
32	ConfigOutput09 (counter configuration)	USINT				•
Stall detection						
84	FullStepThreshold01	UINT				•
92	StallDetectMinSpeed01	UINT				•
98	QuickStopRamp	UINT				•
Load-dependent current control						
130	CfO_SmartEnable01	UINT				•
SGT configuration						
120	SGT_Speed01a	UINT				•
122	SGT_Speed01b	UINT				•
124	SGT_Value01a	INT				•
126	SGT_Value01b	INT				•
Reading back the configuration						
33	ConfigOutput03Read (holding current)	USINT		•		
34	ConfigOutput04Read (nominal current)	USINT		•		
35	ConfigOutput05Read (maximum current)	USINT		•		
Communication						
Motor detection						
81	MotorIdentTrigger	USINT				•
12	Motoridentification01	UINT		•		
74	MotorLoad	UINT	•			
Motor control						
54	Load-dependent current control	USINT			•	
	CurrentControlEnable	Bit 7				
Index* 2 + 16	MotorStepN (index N = 0 to 3)	UINT			•	
Input state						
4	Input counter value	USINT	•			
	StatusInput01	Bit 2				
	StatusInput02	Bit 3				
	StatusInput03	Bit 4				
	DigitalInput01	Bit 5				
	DigitalInput02	Bit 6				

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Error handling						
4	Module power supply error	USINT	•			
	ModulePowerSupplyError	Bit 0				
10	Error status	USINT	•			
	StallError	Bit 0				
	OvertemperatureError	Bit 1				
	CurrentError	Bit 2				
	OvercurrentError	Bit 3				
	QuickStopError01	Bit 5				
	EnableError01	Bit 6				
	QuickStopDone01	Bit 7				
	OpenCircuit01	Bit 12				
	OpenCircuit02	Bit 13				
	OpenCircuit03	Bit 14				
54	Error acknowledgment	USINT			•	
	ClearError	Bit 5				
Positioning						
0	PositionSync	INT	•			
6	Positionasync	UINT		•		
86	PositionSync02	INT	•			
60	PositionLatchedSync	INT	•			
64	PositionLatchedASync	INT		•		
Latch and trigger						
54	Latch configuration	USINT			•	
	StartLatch	Bit 0				
	TriggerEdgePos	Bit 1				
	TriggerEdgeNeg	Bit 2				
	TriggerEdge	Bit 3				
	StartTrigger	Bit 4				
72	Latch trigger status	USINT	•			
	LatchInput	Bit 0				
	LatchDone	Bit 1				
	TriggerInput	Bit 4				
68	usSinceTrigger	UINT	•			

5.4 Function model 0 - Standard with SDC and function model MotionConfiguration

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Configuration						
Module / Motor configuration						
46	ConfigOutput02 (module configuration 1)	UINT				•
96	CounterConfig2	UINT				•
102	SDCConfig01	USINT				•
103	MotorSettlingTime01	USINT				•
107	DelayedCurrentSwitchOff01	USINT				•
Configure currents						
33	ConfigOutput03 (holding current)	USINT				•
34	ConfigOutput04 (nominal current)	USINT				•
35	ConfigOutput05 (maximum current)	USINT				•
Counter configuration						
32	ConfigOutput09 (counter configuration)	USINT				•
Stall detection						
84	FullStepThreshold01	UINT				•
92	StallDetectMinSpeed01	UINT				•
98	QuickStopRamp	UINT				•
Load-dependent current control						
130	CfO_SmartEnable01	UINT				•
SGT configuration						
120	SGT_Speed01a	UINT				•
122	SGT_Speed01b	UINT				•
124	SGT_Value01a	INT				•
126	SGT_Value01b	INT				•
Reading back the configuration						
33	ConfigOutput03Read (holding current)	USINT		•		
34	ConfigOutput04Read (nominal current)	USINT		•		
35	ConfigOutput05Read (maximum current)	USINT		•		

Register description

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Communication						
Motor detection						
81	MotorIdentTrigger	USINT				•
12	Motoridentification01	UINT		•		
74	MotorLoad	UINT	•			
Motor control						
100	Motor current	USINT			•	
	DriveEnable01	Bit 0				
	BoostCurrent01	Bit 1				
	StandstillCurrent01	Bit 2				
	CurrentControlEnable01	Bit 3				
16	Motor1Step0	INT			•	
SDC life sign monitoring						
112	SetTime01	INT			•	
73	LifeCnt	SINT	•			
Input state						
4	Input counter value	USINT	•			
	StatusInput01	Bit 2				
	StatusInput02	Bit 3				
	StatusInput03	Bit 4				
	DigitalInput01	Bit 5				
	DigitalInput02	Bit 6				
Error handling						
4	Module power supply error	USINT	•			
	ModulePowerSupplyError	Bit 0				
10	Error status	USINT	•			
	StallError01	Bit 0				
	OvertemperatureError01	Bit 1				
	CurrentError01	Bit 2				
	OvercurrentError01	Bit 3				
	DrvOk01	Bit 4				
	QuickStopError01	Bit 5				
	EnableError01	Bit 6				
	QuickStopDone01	Bit 7				
	OpenCircuit01	Bit 12				
	OpenCircuit02	Bit 13				
	OpenCircuit03	Bit 14				
54	Error acknowledgment	USINT			•	
	ClearError01	Bit 5				
Homing						
200 204	RefPulsePos01 RefPulsePos01	INT	•			
212 214	RefPulseCnt01 RefPulseCnt01	SINT	•			
Positioning						
0	ActPos01	INT	•			
6	Positionasync	UINT		•		
64	PositionLatchedAsync	INT		•		
220	ActTime01	INT	•			
Trigger						
216	TriggerCnt01	SINT	•			
208	TriggerTime01	INT	•			

5.5 Function model 254 - "Bus controller" and function model 3 - "Ramp"

Register	Offset[1]	Name	Data type	Read		Write	
				Cyclic	Acyclic	Cyclic	Acyclic
Configuration							
Configure currents							
48	-	ConfigOutput03a (holding current)	USINT				•
49	-	ConfigOutput04a (nominal current)	USINT				•
50	-	ConfigOutput05a (maximum current)	USINT				•
Motion generator							
306	-	GeneralConfig01	USINT				•
52	-	MaxSpeed01pos	UINT				•
54	-	MaxAcc01	UINT				•
56	-	MaxDec01	UINT				•
58	-	RevLoop01	INT				•
75	-	JoltTime01	USINT				•
60	-	FixedPos01a	DINT				•
64	-	FixedPos01b	DINT				•
Stall detection							
72	-	FullStepThreshold01	UINT				•
74	-	StallRecognitionDelay01	USINT				•
78	-	StallDetectMinSpeed01	UINT				•
Homing							
68	-	RefSpeed01	UINT				•
70	-	RefConfig01	SINT				•
Limitations							
308	-	LimitSwitchConfig01	USINT				•
344	-	PositionLimitMin01	DINT				•
348	-	PositionLimitMax01	DINT				•
Load-dependent current control							
130	-	CfO_SmartEnable01	UINT				•
SGT configuration							
120	-	SGT_Speed01a	UINT				•
122	-	SGT_Speed01b	UINT				•
124	-	SGT_Value01a	INT				•
126	-	SGT_Value01b	INT				•
Reading back the configuration							
48	-	ConfigOutput03aRead (holding current)	USINT		•		
49	-	ConfigOutput04aRead (nominal current)	USINT		•		
50	-	ConfigOutput05aRead (maximum current)	USINT		•		
Communication							
Motor detection							
84	-	Motoridentification01	UINT		•		
8	8	MotorLoad	UINT	•			
Motor control							
6	6	MpGenMode01	SINT			•	
82	-	ModeReadback01	SINT		•		
4	4	MpGenControl01	UINT			•	
80	-	ControlReadback01	UINT		•		
4	4	MpGenStatus01	UINT	•			
Input state							
6	6	InputStatus	USINT	•			
Error handling							
98	-	ErrorCode01	UINT		•		
Homing							
86	-	RefPos01CyclicCounter	DINT		•		
94	-	RefPos01AcyclicCounter	DINT		•		
Positioning and speed							
0	0	AbsPos01	DINT			•	
0	0	AbsPos01ActVal	INT	•			
90	-	AbsPos1ActValAcyclic	DINT		•		

1) The offset specifies the position of the register within the CAN object.

5.6 All function models - Configuration registers

5.6.1 Load-dependent current control

5.6.1.1 Configuring current control

Name:
CfO_SmartEnable01

The values for load-dependent current control can be set in this register. In order to save energy, the power supply can thus be reduced to a minimum of 25% of the nominal current when the motor load is low.

For increased motor loads, the module automatically adjusts the power supply according to the set values.

Data type	Values	Bus controller default setting
UINT	See the bit structure.	0

Bit structure:

Bit	Name	Value	Information
0 - 3	MotorLoad_LowerLimit	0	Current control disabled (bus controller default setting)
		1 to 15	
4	Reserved	0	
5 - 6	Current increase[1)]	0	1 measured values (bus controller default setting)
		1	2 measured values
		2	4 measured values
		3	8 measured values
7	Reserved	0	
8 - 11	MotorLoad_UpperLimit	0 to 15	
12	Reserved	0	
13 - 14	Current decrease[1)]	0	32 measured values (bus controller default setting)
		1	8 measured values
		2	2 measured values
		3	1 measured value
15	Current reduction	0	Reduction to 50% of the nominal current (bus controller default setting)
		1	Reduction to 25% of the nominal current

1) The measured values are recorded with each full step of the motor.

MotorLoad_LowerLimit

Increase of the current up to a maximum of 100% of the nominal current set in register "ConfigOutput04" on page 46 if the motor load is less than (MotorLoad_LowerLimit * 32). If MotorLoad_LowerLimit = 0, load-dependent current control is disabled.

Current increase

Specifies the number of motor load measured values that are less than or equal to the limit value (MotorLoad_LowerLimit * 32) in order to increase current.

MotorLoad_UpperLimit

Reduces the current if the motor load is greater than (MotorLoad_LowerLimit + MotorLoad_UpperLimit + 1) * 32.

Current decrease

Number of motor load measured values that are greater than or equal to the limit value (MotorLoad_LowerLimit + MotorLoad_UpperLimit + 1) * 32 in order to decrease the current.

Current reduction

Maximum reduction of the current to 25% or 50% of the nominal current.

5.6.2 SGT configuration

5.6.2.1 SGT speed

Name:
SGT_Speed01a to SGT_Speed01b

The associated speed value for the respective SGT_Value can be set in these registers.

Data type	Values	Information
UINT	0 to 65535	In steps/second (function models Standard and MotionConfiguration) In microsteps/cycle (function models "Bus controller" and "Ramp")

5.6.2.2 SGT correction values

Name:
SGT_Value01a to SGT_Value01b

These registers optimize the motor load measurement.

The SGT value (stall guard threshold) optimizes the motor load measurement. A negative value increases the sensitivity of the measurement; a positive value decreases it.

The correction value used depends on the value of SGT_Speed. The following applies:

- SGT_Value0xa is used if the motor speed ≤ SGTSpeed0xa.
- SGT_Value0xb is used if the motor speed ≥ SGTSpeed0xb.
- If the motor speed lies between the values, the SGT_Value is interpolated linearly.

The initial value for the optimization is 0. Values below -10 or over +10 should not be used.

Data type	Values
INT	-64 to 63

5.7 Configuration registers - Function model 0 and function model MotionConfiguration

5.7.1 Module configuration

5.7.1.1 Module configuration 1

Name:
ConfigOutput02

The number of transfer values and the resolution of microsteps for the drive can be configured in this register.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	The setting for these two bits determines the meaning of bits 2 and 3 in register "Input counter value" on page 55.	x	
1 - 2	Reserved	0	
3 - 4	Number of transfer values per X2X Link cycle (see "Motor settings" on page 53)	00	1 x Δs / Δt (transfer values: MotorStep0)
		01	2 x Δs / Δt (transfer values: MotorStep0 - MotorStep1)
		10	4 x Δs / Δt (transfer values: MotorStep0 - MotorStep3)
		11	Reserved
5 - 6	Resolution of microsteps for the following registers: • "Motor settings" on page 53 • "Position sync/async" on page 59	00	Resolution: 5 bits (bits 0 to 4) microsteps, 8 bits (bits 5 to 12) full steps
		01	Resolution: 6 bits (bits 0 to 5) microsteps, 7 bits (bits 6 to 12) full steps
		10	Resolution: 7 bits (bits 0 to 6) microsteps, 6 bits (bits 7 to 12) full steps
		11	Resolution: 8 bits (bits 0 to 7) microsteps, 5 bits (bits 8 to 12) full steps
7 - 15	Reserved	0	

5.7.1.2 Module configuration 1 with SDC

Register "Module configuration 1" on page 44 is ignored in function model "Standard" with SDC information enabled. The module behaves as follows, i.e. whether the module configuration was written in this way:

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Meaning of bits 2 and 3 in register "Input counter value" on page 55	x	
1 - 2	Reserved	0	
3 - 4	Number of transfer values per X2X cycle	00	1x Δs / Δt (see register "Motor1Step0" on page 54)
5 - 6	Resolution of microsteps	11	8-bit microsteps
7 - 15	Reserved	0	

5.7.1.3 Configuration of inputs

Name:
CounterConfig2

Whether inputs are used as incremental encoders or digital inputs can be defined in this register.

When configured as "AB encoder with 2 additional digital inputs", digital input 1 can be configured as a quick stop and digital input 2 as an enable input. See also "ABR interface and digital inputs" on page 24.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Operating mode of the inputs	0	ABR encoders
		1	AB encoder with 2 additional digital inputs
1	Reserved	0	
2 - 4	Quick stop	000	Disabled
		001	If DI1 = Low
		010	If DI1 = High
		011 - 111	Reserved
5 - 7	Enable	000	Disabled
		001	If DI2 = Low
		010	If DI2 = High
		011 - 111	Reserved

5.7.1.4 SDC configuration

Name:
SDCConfig01

This register can be used to enable or disable additional SDC information.

The additional cyclic registers are hidden or shown depending on whether SDC information is disabled or enabled. It is comparable to the two variants of function model "Standard" with and without SDC information.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Trigger edge	0	Rising trigger edge
		1	Falling trigger edge
1 - 5	Reserved	0	
6	SDC life sign monitoring	0	Disabled
		1	Enabled
7	SDC information[1)]	0	Disabled
		1	Enabled

1) If bit "SDC information" is enabled, bit "EncOK01" is displayed in the Automation Studio I/O mapping. This bit is permanently linked to bit ModulOK and always indicates its value.

Information:

Neither SDC information nor SDC life sign monitoring is permitted to be changed at runtime.

5.7.1.5 Motor settling time

Name:
MotorSettlingTime01

This register determines the motor setting time. The motor settling time determines the minimum time between when the motor is powered on to when the DrvOk bit is set (see "Error status" on page 56). The setting is made in steps of 10 ms.

Data type	Values	Information
USINT	1 to 255	10 ms to 2.55 s, default: 10 ms

5.7.1.6 Switch-off delay

Name:
DelayedCurrentSwitchOff01

If the SDC life sign monitoring is triggered (i.e. the NetTime timestamp is in the past), the motor is decelerated at nominal current with speed setpoint = 0.

Then the motor is switched off after the delay configured with this register.

Data type	Values	Information
USINT	0 to 255	0 to 25.5 s in steps of 100 ms (default: 100 ms)

5.7.2 Configuring currents

5.7.2.1 Holding current, nominal current and maximum current

Name:
ConfigOutput03 (holding current)
ConfigOutput04 (nominal current)
ConfigOutput05 (maximum current)

The holding current, nominal current and maximum current registers are used to configure the desired motor current.

Register	Description
Nominal current	Current during operation at constant speed
Maximum current	Current during acceleration phases
Holding current	Current when motor at standstill

Switching between preset current values (holding current, rated current, maximum current):

Function model	Switching between preset current values at runtime
Standard	Using bits 14 and 15 in registers "Motor settings" on page 53
Standard with enabled SDC information	Using register "Motor current" on page 53

Data type	Values	Unit
USINT	0 to 200	Percentage of the nominal motor current • 100% corresponds to the nominal current of the motor bridge power unit listed in the technical data • 200% corresponds to the maximum current of the motor bridge power unit listed in the technical data.

5.7.2.2 Reading back the holding current, nominal current and maximum current

ConfigOutput03Read (holding current)
ConfigOutput04Read (nominal current)
ConfigOutput05Read (maximum current)

These registers are used to read the respective current values in percent.

Data type	Values	Unit
USINT	0 to 255	Percentage of module's nominal current (100% corresponds to the nominal current of the motor bridge power unit in the technical data)

5.7.3 Counter configuration

5.7.3.1 Counter configuration

Name:
ConfigOutput09

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	ABR latch function	0	Negative edge: Disable ABR latch function
		1	Positive edge: Enables latch function ABR. After a latch event has occurred, the latch function can be started again on a new rising edge.
1 - 2	Definition of the latch mode	00	Latch counter value: ABR unconditional
		01	Latch counter value: ABR on a positive edge of input R
		10	Latch counter value: ABR on a negative edge of input R
		11	Reserved
3		0	• Position sync: Internal position counter • Position async: ABR counter state • Position latched sync: Internal position counter[1)] • Position latched async: ABR counter value[1)]
		1	• Position sync: ABR counter state • Position async: Internal position counter • Position latched sync: ABR counter value[1)] • Position latched async: Internal position counter[1)]
4 - 7	Reserved		

1) These registers are not available in function model "Standard" with SDC information enabled.

5.7.4 Stall detection

5.7.4.1 Full step limit value

Name:
FullStepThreshold01

Starting with the speed specified in this register, the motor is operated in full step mode; below it, it is operated in steps per second.

Data type	Values	Information
UINT	0	Full step mode disabled
	1 to 65535	Steps/second

5.7.4.2 Minimum speed for stall detection

Name:
StallDetectMinSpeed01

If the motor speed exceeds the value set in this register, then stall detection is enabled. No stall error is reported below this threshold value.

Data type	Values	Information
UINT	0 to 65535	Minimum speed in steps/second.

5.7.4.3 Deceleration ramp

Name:
QuickStopRamp

The deceleration ramp can be set in microsteps per X2X cycle^2 in this register.

Data type	Values
UINT	0 to 65535

For a calculation example, see "Quick stop and enable" on page 25.

5.8 Function models 3 and 254 - Configuration registers

5.8.1 Configuring currents

5.8.1.1 Holding current, nominal current and maximum current

Name:
ConfigOutput03a (holding current)
ConfigOutput04a (nominal current)
ConfigOutput05a (maximum current)

The holding current, nominal current and maximum current registers are used to configure the desired motor current.

Register	Description
Nominal current	Current during operation at constant speed
Maximum current	Current during acceleration phases
Holding current	Current when motor at standstill

Data type	Values	Unit
USINT	0 to 200	Percentage of the nominal motor current • 100% corresponds to the rated current of the motor bridge power unit listed in the technical data • 200% corresponds to the maximum current of the motor bridge power unit listed in the technical data Bus controller default setting: 0

5.8.1.2 Reading back the holding current, nominal current and maximum current

ConfigOutput03aRead (holding current)
ConfigOutput04aRead (nominal current)
ConfigOutput05aRead (maximum current)

These registers are used to read the respective current values in percent.

Data type	Values	Unit
USINT	0 to 255	Percentage of the nominal motor current 100% corresponds to the nominal current of the motor bridge power unit listed in the technical data.

5.8.2 Motion generator

5.8.2.1 General configuration

Name:
GeneralConfig01

This register can be used to switch the positioning mode with bit 0 and to configure the cycle time of the movement profile generator.

- 0: "Mode 1: Position mode" without extended control word
- 1: "Mode 1: Position mode with extended control word"

Data type	Values	Bus controller default setting
USINT	See the bit structure.	0

Bit structure:

Bit	Description	Value	Information
0	Position mode	0	Without extended control word (bus controller default setting)
		1	With extended control word
1 - 2	Cycle time of the motion profile generator[1]	00	25 ms (bus controller default setting)
		01	10 ms
		10	5 ms
		11	Reserved
3 - 7	Reserved	0	

1) The cycle time for the motion profile generator is configured with this cycle. This cycle time affects the unit for specifying the speed and acceleration:

- Unit for speed: Microsteps/Cycle
- Unit for acceleration: Microsteps/Cycle^2

5.8.2.2 Reversing loop

Name:
RevLoop01

This register defines the steps for the reverse loop.

This parameter is only used in modes 1, -123, -124, -125, -126 (absolute positioning modes).

Data type	Values	Information
INT	-32768 to 32767	Bus controller default setting: 0

5.8.2.3 Jerk time

Name:
JoltTime01

This register is used to specify the number of cycles for the jerk limitation time. If more than 80 cycles are entered, they are limited to 80.

Data type	Values	Information
USINT	0	No jerk limitation time. Bus controller default setting: 0
	1 to 80	Number of cycles

5.8.2.4 Fixed position A

Name:
FixedPos01a

Depending on the mode, this register can be used to define a number of steps or a target position:

- -121: Number of steps to be moved
- -124 (with 1 on the digital input) and -125: Value of target position to be approached

Data type	Values	Information
DINT	-2,147,483,648 to 2,147,483,647	Bus controller default setting: 0

5.8.2.5 Fixed position B

Name:
FixedPos01b

This register defines the position to move to in modes -124 (if the digital input is set to 0) and -126.

Data type	Values	Information
DINT	-2,147,483,648 to 2,147,483,647	Bus controller default setting: 0

5.8.2.6 Maximum speed

Name:
MaxSpeed01pos

This register defines the maximum speed for the absolute positioning modes (1, -123, -124, -125, -126).

Information:

The setting does not apply to the speed and homing modes (2, -127, -128).

Data type	Values	Information
UNIT	0 to 65535	Speed in microsteps/cycle. Bus controller default setting: 0

5.8.2.7 Maximum acceleration

Name:
MaxAcc01

This register defines the maximum acceleration. (also applies for homing modes).

Data type	Values	Information
UINT	0 to 65535	Acceleration in microsteps/cycle². Bus controller default setting: 0

5.8.2.8 Maximum deceleration

Name:
MaxDec01

This register defines the maximum deceleration. (also applies for homing modes).

Data type	Values	Information
UINT	0 to 65535	Brake deceleration in microsteps/cycle². Bus controller default setting: 0

5.8.3 Stall detection

5.8.3.1 Full step limit value

Name:
FullStepThreshold01

Starting with the speed specified in this register, the motor is operated in full step mode; it is operated in microstep mode below it.

Data type	Values	Information
UINT	1 to 65534	Speed in microsteps/cycle. Bus controller default setting: 0
	65535	Motor is always operated in microstep mode

5.8.3.2 Stall recognition delay

Name:
StallRecognitionDelay01

The value in this register is only relevant for mode Home during stall.

A stall is only detected after the time specified here has expired and after the homing procedure has started.

The delay time must be set in cycles.
Delay time = Value * Cycle time of the movement profile generator

For example, a setting of 4 (and a cycle time of 25 ms) means that a stall will not be detected until 100 ms after the motor starts moving (start of the homing procedure).

Data type	Values	Information
USINT	0	No delay (bus controller default setting)
	1 to 255	Delay time in cycles, see "General configuration" on page 48.

5.8.3.3 Minimum speed for stall detection

Name:
StallDetectMinSpeed01

If the motor speed exceeds the value set in this register, then stall detection is enabled. No stall error is reported below this threshold value.

Data type	Values	Information
UINT	0 to 65535	Minimum speed in microsteps/cycle. Bus controller default setting: 0

5.8.4 Homing

5.8.4.1 Homing speed

Name:
RefSpeed01

This register sets the speed for homing modes -127 and -128.

Data type	Values	Information
UINT	0 to 65535	Speed in microsteps/cycle. Bus controller default setting: 0

5.8.4.2 Homing configuration

Name:
RefConfig01

The homing mode can be set with this register.

Data type	Values	Information
SINT	-120	Set home position
	-121	Homing on positive edge of input DI 2
	-122	Homing on negative edge of input DI 2
	-125	Homing on positive edge of input DI 1 (R pulse). (Bus controller default setting)
	-126	Homing on negative edge of input DI 1 (R pulse)
	-127	Homing during stall detection
	-128	Immediate homing
	All others	No effect

5.8.5 Limitations

5.8.5.1 Limit switch configuration

Name:
LimitSwitchConfig01

The behavior of limit switches for limit monitoring can be configured with this register.

Data type	Values	Bus controller default setting
USINT	See the bit structure.	0

Bit structure:

Bit	Description	Value	Information
0 - 1	Negative limit switch	00	Switched off (bus controller default setting)
		01	Active if low
		10	Reserved
		11	Active if high
2 - 3	Positive limit switch	00	Switched off (bus controller default setting)
		01	Active if low
		10	Reserved
		11	Active if high
4	Inputs	0	ABR
		1	AB and DI
5 - 6	Reserved	0	
7	Direction monitoring	0	Off (bus controller default setting)
		1	On

5.8.5.2 Software limit

Name:
PositionLimitMin01
PositionLimitMax01

This register configures software limits. The function is enabled if at least one of the two registers is not equal to zero.

Data type	Values	Information
DINT	-2,147,483,648 to 2,147,483,647	Bus controller default setting: 0

5.9 Communication registers - Function model 0 and function model Motion-Configuration

5.9.1 Motor detection

5.9.1.1 Motor ID trigger

Name:
MotorIdentTrigger

With this register, a measurement of the motor ID can be initiated acyclically (see "Motor identification" on page 52). The application must ensure that the conditions for measurement are met (see section "Integrated motor detection" on page 14).

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Starting motor detection	0	No effect
		1	Positive edge triggers motor identifier measurement
1 - 7	Reserved	0	

5.9.1.2 Motor identification

Name:
Motoridentification01

This register is used for differentiation in the application and to identify the connected motor type (for service purposes). After successful measurement, this register contains the time [µs] required to apply a current increase of ΔI = 1 A to a motor winding.

Data type	Motor ID values	Explanation
UINT	0	No motor identifier available (after switching on for as long as the measurement conditions are not met)
	1 to 32767	Valid range of values for the motor ID register (unit: µs)
	65534	Invalid value: Overflow

5.9.1.3 Measuring the motor load

Name:
MotorLoad

This register contains the current load measurement value of stall detection and can be used to tune stall detection.

A high value indicates a small load on the motor; the lower the value, the higher the motor load. The SGT values (see "SGT speed" on page 43 and "SGT correction values" on page 43) should be set so that the motor load returns value 0 at maximum load (shortly before stall).

This register can be shown and hidden via the module configuration.

Data type	Values
UINT	0 to 1023

5.9.2 Motor control

5.9.2.1 Motor current

Name:
DriveEnable01
BoostCurrent01
StandstillCurrent01
CurrentControlEnable01

Bits 0 to 2 of this register control the motor's current feed.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	DriveEnable01	x	Motor powered
1	BoostCurrent01	x	Maximum current
2	StandstillCurrent01	x	Holding current
3	CurrentControlEnable01	0	Disables load-dependent current control
		1	Enables load-dependent current control
4 - 7	Reserved	0	

The possible status of bits 0 to 2

StandstillCurrent01	BoostCurrent01	DriveEnable01	Description
x	x	0	Motor not supplied with current
0	0	1	Nominal current supplied to motor
0	1	1	Maximum current supplied to motor
1	0	1	Holding current supplied to motor
1	1	1	Holding current supplied to motor

5.9.2.2 Motor settings

Name:
MotorStep0 to MotorStep3

These registers specify the number and direction of steps that must be moved by the module during the next X2X cycle and also select the motor current (see also "Holding current, nominal current and maximum current" on page 46).

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 12	Number of steps for the module to move during the next X2X cycle	x	
13	Direction of movement	0	Positive
		1	Negative
14 - 15	Selection of motor current	00	Motor not powered
		01	Holding current
		10	Nominal current
		11	Maximum current

The number of transfer values per X2X cycle is specified by bits 3 and 4 in the module configuration 1 (see "Module configuration 1" on page 44). If only one transfer value (bits 3 and 4 = 00) is specified, then the motor is advanced by MotorStep0 until the next X2X cycle. If 2 or 4 transfer values are specified, they are added and executed in the next X2X cycle.

Example: X2X cycle = 1 ms (1000 µs)

Time	Number of transfer values (see "Module configuration 1" on page 44)		
	1 (bits 3 - 4 = 00)	2 (bits 3 - 4 = 01)	4 (bits 3 - 4 = 10)
0 - 250 µs)	MotorStep0	MotorStep0	MotorStep0
250 - 500 µs)			MotorStep1
500 - 750 µs)		MotorStep1	MotorStep2
750 - 1000 µs)			MotorStep3

5.9.2.3 Step specification

Name:
Motor1Step0

This registers is used to specify the number and direction of steps that should be carried out by the module during the next X2X cycle.

The value is specified with a resolution of 1/256 of a full step (corresponds to 8-bit microsteps).

The direction of movement is derived from the value's sign:

Data type	Values	Information
INT	>0	Movement in the positive direction in 1/256 full steps
	<0	Movement in the negative direction in 1/256 full steps

5.9.3 Life sign monitoring

5.9.3.1 SDC life sign monitoring

Name:
SetTime01

The module uses SDC life sign monitoring to check whether valid values have been received for the speed setpoint.

SDC life sign monitoring is enabled in register "SDC configuration" on page 45 by setting bit 6 (SDCSetTime = On).

Data type	Values
INT	-32768 to 32767

5.9.3.2 Lifecycle counter

Name:
LifeCnt

This register is incremented by one with each X2X Link cycle.

Data type	Values
SINT	-128 to 127

5.9.4 Input state

5.9.4.1 Input counter value

Name:
ModulePowerSupplyError
StatusInput01 to StatusInput03
DigitalInput01 to DigitalInput02

This register contains the state of the digital inputs and counters depending on the module and limit switch configuration:

- **Bits 2 and 3:** Meaning depends on the module configuration.
- **Bits 4 to 6:** Meaning depends on the limit switch configuration.

Bit	Description	Value	Information
0	ModulePowerSupplyError	0	OK
		1	Module power supply error
1	Reserved	-	
If bit 0 in Module configuration 1 = 0			
2	StatusInput01	0 or 1	Input state of encoder input A
3	StatusInput02	0 or 1	Input state of encoder input B
If bit 0 in Module configuration 1 = 1			
2	StatusInput01	0 or 1	Homing toggle bit for counter 1: After homing is completed, the state of this bit is changed.
3	StatusInput02	0	The homing procedure for the ABR counter is active.
		1	The homing procedure of the ABR counter is completed.
If bit 4 in "LimitSwitchConfig01" on page 51 = 0			
4	StatusInput03	0 or 1	Input state of encoder input R
5 - 6	Reserved	-	
If bit 4 in "LimitSwitchConfig01" on page 51 = 1			
4	Reserved	-	
5	DigitalInput01	0 or 1	Input state - Digital input 1
6	DigitalInput02	0 or 1	Input state - Digital input 2
7	Reserved	-	

5.9.5 Error handling

5.9.5.1 Error status

Name: The names of the bits are different depending on whether SDC information is enabled or disabled.

Without SDC	With SDC
StallError	StallError01
Overtemperature	Overtemperature01
ErrorCurrentError	ErrorCurrentError01
OvercurrentError	OvercurrentError01
-	DrvOK01
	QuickStopError01
	EnableError01
	QuickStopDone01
	OpenCircuit01 to OpenCircuit03

The current error status of the drive is indicated in this register. Each bit indicates a certain error or status. If an error is registered in bits 0 to 3, then the corresponding bit remains set until the error has been acknowledged (see "Latch configuration" on page 60 and "Error acknowledgment" on page 57).

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	StallError(01)	0	No stall
		1	Stall
1	Overtemperature error OvertemperatureError(01)	0	No overtemperature
		1	Overtemperature
2	Current error CurrentError(01)	0	No current error
		1	Current error
3	Overcurrent error OvercurrentError(01)	0	No overcurrent
		1	Overcurrent
4	Status of the drive DrvOk0[1)]	0	An error was triggered for the motor axis
		1	The drive is running error-free
5	Quick stop error QuickStopError01	0	No quick stop
		1	Quick stop triggered
6	Enable input error EnableError01	0	Enable input OK
		1	Enable input error
7	Quick stop finished QuickStopDone01	0	No quick stop or not ready yet
		1	Quick stop ramp decelerated and motor stopped
8 - 11	Reserved	0	
12	Open circuit OpenCircuit01	0	No open circuit
		1	Open circuit detected
...		...	
14	Open circuit OpenCircuit03[2)]	0	No open circuit
		1	Open circuit detected
15	Reserved	0	

1) Only when SDC information is enabled
2) Only if bit 0 = 0 (ABR encoder) in register "CounterConfig2" on page 45

Overtemperature error

The "Overtemperature" error bit can be set for the following reasons:

- The maximum permissible temperature is exceeded due to overload in the area of a channel.
- The module temperature exceeds the defined limit value (see "Overtemperature shutdown" on page 26).

Current error

This error bit occurs whenever the required current cannot be supplied to the motor windings. This can be (but is not necessarily) caused by an open circuit. At higher speeds (depending on the motor), this error can also occur without an open circuit. In this case it is simply no longer possible to supply the desired current to the motor windings. Because of the Back-EMF on the motor, this bit is set at slightly lower speeds if the motor is operated with no load compared with full or partial loads.

Overcurrent error

Overcurrent occurs if there is not enough voltage for the motor winding (e.g. short circuit).

Status of the drive

The status of the drive is only shown when SDC information is enabled. Bit "Drive" is 1 when the following conditions are met:

- Motor turned on (see "Motor current" on page 53)
- MotorID measurement is completed
- Motor is supplied with current
- Motor settling time has passed
- Supply voltage is in the valid range
- No overtemperature fault
- Preset position value is valid (see "SDC life sign monitoring" on page 54)

Quick stop error

This error is triggered by actuating the quick stop input. The motor is brought to a standstill via a predefined ramp. The motor can only be started normally again after the error has been acknowledged. No additional disable/enable from the motor is necessary.

Information:

If the motor is switched on when the error is acknowledged and movement data is present, this causes the motor to start.

Enable input error

This error is triggered by actuating the enable input.

Quick stop finished

This bit indicates that after a quick stop event has been triggered, the quick stop ramp has been decelerated and the motor has come to a standstill.

Open circuit

The module is equipped with open-circuit detection for the digital inputs. If the digital input is not connected to ground or to 24 V, an open circuit is detected.

5.9.5.2 Error acknowledgment

Name:
ClearError01

This register can be used to acknowledge errors that have occurred on the motor.

For more info, see "Error status" on page 56.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 4	Reserved	0	
5	ClearError01	0	No effect
		1	Error acknowledgment for motor
6 - 7	Reserved	0	

5.9.6 Homing

5.9.6.1 Home position

Name:
RefPulsePos01

This register is duplicated and has the following contents:

Register	Description
Home position of the internal position counter	This register contains the home position of the internal position counter.
Home position for the ABR counter	This register contains the home position of the ABR counter.

Data type	Values
INT	-32768 to 32767

Setting "Position sync" in the Automation Studio I/O configuration can be used to select which of the 2 registers is addressed by variable RefPulsePos01.

Variables in Automation Studio	I/O configuration, counter 01, option "Position sync"	
	Step counter 01 is indicated on ActPos01.	ABR counter 01 is indicated on ActPos01 .
RefPulsePos01	Home position of the internal position counter	Home position of the ABR counter
Option "Position sync" for counter 1 also sets bit 3 in register "Counter configuration" on page 47:		
Bit 3 (counter 1)	0	1

5.9.6.2 Reference pulse counter

Name:
RefPulseCnt01

This register is duplicated and has the following contents:

Register	Description
Reference pulse counter for the internal position counter	The reference pulses of the internal position counter are counted in this register.
Reference pulse counter for the ABR counter	The reference pulses of the ABR counter are counted in this register.

Data type	Values
SINT	-128 to 127

Setting "Position sync" in the Automation Studio I/O configuration can be used to select which of 2 registers is addressed by variable RefPulsePos01.

Variables in Automation Studio	I/O configuration, counter 01, option "Position sync"	
	Step counter 01 is indicated on ActPos01.	ABR counter 01 is indicated on ActPos01.
RefPulseCnt01	Reference pulse counter Internal position counter	Reference pulse counter ABR counter
Option "Position sync" for counter 1 also sets bit 3 in register "Counter configuration" on page 47:		
Bit 3 (counter 1)	0	1

5.9.7 Positioning

5.9.7.1 NetTime of the position value

Name:
ActTime01

This register contains the NetTime of the most recent valid position value.

For additional information about NetTime and timestamps, see "NetTime Technology" on page 28.

Data type	Values
INT	-32768 to 32767

5.9.7.2 Position sync/async

Name:
PositionSync (Function model 0 - Standard)
ActPos01 (function model 0 - Standard with SDC)
Positionasync

Depending on the Counter configuration, these registers can be used to read either the internal position counter or the counter value on the ABR input.

Data type	Values
INT	-32768 to 32767

Register	Counter configuration	
	Bit 3 = 0	Bit 3 = 1
PositionSync (ActPos01)	Internal position counter	ABR counter
Positionasync	ABR counter	Internal position counter

5.9.7.3 Position sync 2

Name:
PositionSync02

This register contains the state of either the position counter or the ABR counter depending on Counter configuration (bit 3). It behaves exactly complementary to register "PositionSync" on page 59.

If "PositionSync" contains the position counter, then register "PositionSync02" contains the ABR counter state, and vice versa.

By default, the register is not visible in the I/O mapping; it must first be enabled in the I/O configuration.

Data type	Values
INT	-32768 to 32767

5.9.7.4 Position latched sync-async

Name:
PositionLatchedSync
PositionLatchedAsync

The position counter (internal position counter or ABR counter) is applied at the latch event (see "Latch configuration" on page 60). Bits 3 and 7 of register "Counter configuration" on page 47 are used to determine which counter value (internal position counter or ABR encoder) is saved in registers PositionLatchedSync and PositionLatchedAsync.

Data type	Values
INT	-32768 to 32767

Register	Counter configuration	
	Bit 3 = 0	Bit 3 = 1
PositionLatchedSync	Internal position counter	ABR counter
PositionLatchedAsync	ABR counter	Internal position counter

5.9.8 Latch and trigger

5.9.8.1 Latch configuration

Name:
StartLatch
TriggerEdgePos
TriggerEdgeNeg
StartTrigger
TriggerEdge
ClearError
CurrentControlEnable

The trigger functions for the stepper motor can be configured with this register.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Latch function for stepper motor 2 Latch byte	0	The latch function for stepper motor position is disabled on the negative edge of this bit.
		1	The latch function for stepper motor position is disabled on the positive edge of this bit.
1 - 2	Latch mode for stepper motor TriggerEdgePos (bit 1) TriggerEdgeNeg (bit 2) 1)	00	Latch position of stepper motor unconditional
		01	Latch position of stepper motor on positive edge of input DI1 or R
		10	Latch position of stepper motor on negative edge of input DI 1 or R
		11	Reserved
3	TriggerEdge 2)	0	Trigger edge (input DI 2) = Positive
		1	Trigger edge (input DI 2) = Negative
4	Enable trigger (on change) StartTrigger	x	
5	ClearError	0	No effect
		1	Error acknowledgment for motor (for more information, see "Error status" on page 56)
6	Reserved	-	
7	CurrentControlEnable	0	Disables load-dependent current control
		1	Enables load-dependent current control

1) Depends on bit 4 in register "LimitSwitchConfig01" on page 51
2) Only available if bit 4 = 1 (AB + DI) in register "LimitSwitchConfig01" on page 51

Trigger function procedure

- Selection of the desired trigger edge using bit 3
- Enabling of the trigger function by changing the state of bit 4. Changing this bit deletes Time since trigger event (µs counter).
- When the trigger event occurs, µs counter Time since trigger event is started.
- Counter Time since trigger event cannot overflow, i.e. the counter is stopped at 2^{16} - 1 and retains this value until the next time the trigger function is enabled.

The trigger function can be re-enabled at any time by changing the state of bit 4, regardless of whether a trigger event has occurred or if Time since trigger event has reached the maximum value.

5.9.8.2 Latch trigger status

Name:
LatchInput
LatchDone
TriggerInput

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	LatchInput	x	Digital input for the latch event (level)
1	LatchDone	x	Changes its state each time the counter state is successfully latched (reset value = 0)
2 - 3	Reserved	-	
4	TriggerInput	x	Trigger input (level)
5 - 7	Reserved	0	

5.9.8.3 Trigger counter

Name:
TriggerCnt01

This register contains a cyclic counter that is incremented with each trigger event.

Data type	Values
SINT	-128 to 127

5.9.8.4 Trigger timestamp

Name:
TriggerTime01

This register contains the NetTime instant of the most recent trigger event. The trigger edge must be configured in register "SDC configuration" on page 45.

For additional information about NetTime and timestamps, see "NetTime Technology" on page 28.

Information:

The absolute accuracy of the trigger can be delayed up to 5 µs due to the input filter of the digital inputs.

Data type	Values
INT	-32768 to 32767

5.9.8.5 Time since trigger event

Name:
usSinceTrigger

This register indicates the time in microseconds that has passed since the trigger event occurred (see "Latch configuration" on page 60).

Information:

The absolute accuracy of the trigger can be delayed up to 5 µs due to the input filter of the digital inputs.

Data type	Values
UINT	0 to 65535

5.10 Function models 3 and 254 - Communication registers

5.10.1 Motor detection

5.10.1.1 Motor identification

Name:
Motoridentification01

This register is used for differentiation in the application and to identify the connected motor type (for service purposes). After successful measurement, this register contains the time [µs] required to apply a current increase of ΔI = 1 A to a motor winding.

Data type	Motor ID values	Explanation
UINT	0	No motor identifier available (after switching on for as long as the measurement conditions are not met)
	1 to 32767	Valid range of values for the motor ID register (unit: µs)
	65534	Invalid value: Overflow

5.10.1.2 Measuring the motor load

Name:
MotorLoad

This register contains the current load measurement value of stall detection and can be used to tune stall detection.

A high value indicates a small load on the motor; the lower the value, the higher the motor load. The SGT values (see "SGT speed" on page 43 and "SGT correction values" on page 43) should be set so that the motor load returns value 0 at maximum load (shortly before stall).

This register can be shown and hidden via the module configuration.

Data type	Values
UINT	0 to 1023

5.10.2 Motor control

5.10.2.1 Mode

Name:
MpGenMode01

Data type	Values	Information
SINT	0	No mode selected
	1	Depending on bit 0 in the "General configuration" on page 48 register, the position mode will behave as follows: • Position mode without extended control word: Move to target position as soon as the target position is changed • Position mode with extended control word: Move to the target position as described in "Mode 1 - Position mode with extended control word" on page 63
	2	Speed mode: Constant speed
	-120	Set home position
	-121	Remaining distance mode
	-122	Set actual position
	-123	Move to target position when external input set
	-124	Two position mode
	-125	Move to fixed position A
	-126	Move to fixed position B
	-127	Positive homing (see also "Homing configuration" on page 51)
	-128	Negative homing (see also "Homing configuration" on page 51)

Information:

For all modes: The "Target reached" bit is set in the "Status word" on page 66 register when the current action is finished (i.e. when the position or speed is reached, depending on the mode).

A new position or speed can be specified even before the current action is finished.

5.10.2.1.1 Mode 1 - Position mode

The position setpoint is specified in register "Set position/speed" on page 67. The motor is then moved to this new position.

How the position is applied can be controlled by bit 0 in register "GeneralConfig01" on page 48.

- If bit 0 equals 0 (no extended control word), the position setpoint will be applied as soon as it is not equal to the current position. The new position is then used for the movement.
- If bit 0 equals 1, the position setpoint is applied as described in "Mode 1 - Position mode with extended control word" on page 63.

5.10.2.1.2 Mode 1 - Position mode with extended control word

Position mode with extended control word behaves like Position mode 1 as described previously (without the extended control word), but the new position setpoint ("Position/Speed" on page 67 register) is applied according to the extended control word.

5.10.2.1.2.1 Extended control word

Commands can be issued using this register depending on the state of the module (see "Operating function model "Ramp"" on page 34).

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 3	Corresponds to the default Control word	x	
4	New setpoint	0	Do not apply target position.
		1	Apply target position.
5	Change set immediately	0	Complete current positioning movement and then start next positioning movement
		1	Interrupt current positioning movement and then start next positioning movement
6	abs / rel	0	Target position is an absolute value.
		1	Target position is a relative value.
7	Corresponds to the default Control word	x	
8	Stop[1]	0	Execute positioning
		1	Stop axis with deceleration
9 - 15	Corresponds to the default Control word	x	

1) This bit applies to all modes.

5.10.2.1.2.2 Extended status word

The bits in the status word reflect the state of the state machine (for a detailed description, see "Structure of the status word" on page 35 and "State machine" on page 36).

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 9	Corresponds to the default Status word	x	
10	Target reached, depends on bit 8 (Stop) in register Control word		**If stop = 0**
		0	Target position not reached.
		1	Target position reached
			If stop = 1
		0	Axis decelerating
		1	Axis speed = 0
11	Corresponds to the default Status word	x	
12	Setpoint acknowledge	0	Ramp generator did not apply the position value
		1	Ramp generator applied the position value
13 - 15	Corresponds to the default Status word	x	

5.10.2.1.2.3 Position setting

The target position can be defined in 2 different ways:

Type of setpoint definition	Description
Single setpoint	After the target position is reached, Bit Target reached in register "Status word" on page 66 is set. Then a new target position is defined. The drive stops at each target position before starting the movement to the next target position.
Set of setpoints	After the target position has been reached, the movement to the next target position is started immediately without stopping the drive. It is therefore possible to initiate a new positioning by specifying another target position during active positioning.

The two modes "Single setpoint" and "Set of setpoints" are controlled by the timing of bits New setpoint and Change set immediately in the "extended control word" on page 63 and Setpoint acknowledge in register "Extended control word" on page 63.

5.10.2.1.2.4 Relative position setting

If bit abs / rel bit in register Extended control word is set, then the target position is interpreted as a relative value. At each New setpoint trigger, the target position will be increased by this value (or decreased if the value is negative).

If the mode changes between the position settings, relative movement will then proceed starting at the last specified position. The position setpoint mode is initialized with 0 when the module is started.

5.10.2.1.3 Mode 2: Speed mode - Constant speed (pos./neg.)

The value in the "position/speed" on page 67 register is interpreted as the speed setpoint (microsteps / cycle).

Observing the maximum permissible acceleration, the motor moves with a ramp to the desired speed setpoint and maintains this speed until a new speed setpoint is specified.

Values are allowed within the range -65535 to 65535. When a value is entered outside of this range, it is readjusted to these limits.

5.10.2.1.4 Mode -120: Set home position

The current actual position is modified such that the position specified in the "position/speed" on page 67 register is the home position. If moved to this position, the motor is at the home position.

The home position in register "Home position" on page 67 is also set to this value.

5.10.2.1.5 Mode -121: Remaining distance mode (like mode 1)

With a rising or falling edge on digital input 1, the current target position is discarded and only the number of steps set in register "Fixed position A" on page 49 will be moved forward or backward.

The "homing configuration" on page 51 defines whether a rising or falling edge of the digital input is used as a trigger.

5.10.2.1.6 Mode -122: Set actual position

The target position set in register "Position/Speed" on page 67 is applied as the current actual position in the internal position counter.

Before this mode is started, the motor must be at a standstill and physically located at the point for which the position being set should be applied.

5.10.2.1.7 Mode -123: Move to the target position when the external input is set

The position setpoint set in register "Position/Speed" on page 67 is approached on a rising edge on digital input 1.

A new position setpoint is only applied on a new rising edge of the associated digital input. This can also take place during the ongoing positioning process and is then immediately effective.

5.10.2.1.8 Mode -124: Two-position mode

The positions to be approached are defined in registers Fixed position A and Fixed position B.

When value 1 is on digital input 1, fixed position A is approached. If value 0, fixed position B is approached. Toggling is also possible during an ongoing positioning operation.

5.10.2.1.9 Mode -125/-126: Move to fixed position X

These modes can be used to move to specified fixed positions.

- Mode -125: "Fixed position A" on page 49
- Mode -126: "Fixed position B" on page 49

5.10.2.1.10 Mode -127/-128: Positive/Negative homing

Mode -127 or -128 is used to select the direction used for homing. Whether referencing should occur at low/high level on the digital input, during stall or unconditionally must be set in the "referencing configuration" on page 51.

If the referencing condition occurs, then the motor stops and the values of the position counter and ABR counter valid at the moment when the referencing condition occurs are written to the "Referenced zero position" on page 67 register.

5.10.2.2 Read back mode

Name:
ModeReadback01

This register can be used to read the contents of register "Mode" on page 62.

Data type	Values
SINT	-128 to 127

5.10.2.3 Control word

Name:
MpGenControl01

Using this register, commands can be sent depending on the state of the module (see "Operating function model "Ramp"" on page 34).

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Switch on	x	
1	Enable voltage	x	
2	Quick stop	x	
3	Enable operation	x	
4 - 6	Mode-specific	x	
7	Fault reset	x	
8	Stop[1)]	x	
9	CurrentControlEnable	0	Disables load-dependent current control
		1	Enables load-dependent current control
10	Reserved	0	
11	Motor ID trigger	0	No effect
		1	Rising edge: Motor ID trigger[2)]
12	Warning reset	0	No effect
		1	Rising edge: Reset warnings
13	Current error detection	0	Disable current error detection (default)
		1	Enable current error detection
14	Encoder position sync/async	0	Default: • Internal position counter, cyclic • ABR counter, acyclic
		1	• Internal position counter, acyclic • ABR counter, cyclic
15	Stall detection	0	Disable stall detection (default)
		1	Enable stall detection

1) Bit "Stop" is only evaluated if the extended control word is enabled (see "General configuration" on page 48).
2) This bit can be used to trigger a measurement of the motor ID. Keep in mind that the application must ensure that the conditions for measurement are fulfilled (see table in the "Motor identification" on page 62 register).

5.10.2.4 Read back control word

Name:
ControlReadback01

This register can be used to read the contents of register "Control word" on page 65.

Data type	Values
UINT	0 to 65535

5.10.2.5 Status word

Name:
MpGenStatus01

The bits in this register reflect the state of the state machine. For a more detailed description, see "Structure of the status word" on page 35 and "State machine" on page 36.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Ready to switch on	x	
1	Switched on	x	
2	Operation enabled	x	
3	Fault (error bit)	x	
4	Voltage enabled	x	
5	Quick stop	x	
6	Switch on disabled	x	
7	Warning	x	
8	Reserved	0	
9	Remote	1	Always 1
10	Target reached	x	
11	Internal limit active	0	No limit violation
		1	Internal limit is active (upper or lower software limit violated)
12	Mode-specific	x	
13 - 15	Reserved	0	Always 0

5.10.3 Input state

5.10.3.1 Digital input status

Name:
InputStatus

This register indicates the logical states of encoder and digital inputs.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Input state A	0 or 1	Input state of encoder input A
1	Input state B	0 or 1	Input state of encoder input B
2	Input state R [1]	0 or 1	Input state of encoder input R
3	Input state DI1 [1]	0 or 1	Input state - Digital input 1
4	Input state DI2 [1]	0 or 1	Input state - Digital input 2
5	Open circuit A	0	No error
		1	Open circuit of encoder input A
6	Open circuit B	0	No error
		1	Open circuit of encoder input B
7	Open circuit R	0	No error
		1	Open circuit of encoder input R

1) If bit 4 = 1 (AB + DI) in register "LimitSwitchConfig01" on page 51

5.10.4 Error handling

5.10.4.1 Error code

Name:
ErrorCode01

The cause of an error or warning can be read in this register.

Data type	Error code	Error type	Priority	Description
UINT	0x0000	-	-	No error
	0x3000	Error	High	Voltage
	0x4200	Error	↓	Overtemperature
	0xFF20	Warning	↓	Negative limit switch
	0xFF21	Warning	↓	Positive limit switch
	0x2300	Warning	↓	Overcurrent
	0xFF00	Warning	↓	Current error[1)]
	0xFF01	Warning	↓	Stall [2)]
	0xFF11	Warning	↓ low	Open circuit

1) A current error is only detected if bit 13 = 1 in the control word (current error detection enabled).
2) Stall is only detected if bit 15 = 1 in the control word (stall detection enabled).

Information regarding the handling of errors and warnings:

- Bit 3 (fault) and bit 8 (warning) in the status word can be used to query whether an error or a warning was reported in the error code register.
- Bit 7 (Fault Reset) and bit 8 (Warning Reset) in the control word are used to acknowledge pending errors and warnings.
- If two or more errors/warnings are pending, the one with the highest priority (the order in the table above) will be displayed in the error code register.

5.10.5 Homing

5.10.5.1 Homed zero position

Name:
RefPos01CyclicCounter
RefPos01AcyclicCounter

After a homing procedure, the homing point for the cyclic or acyclic position counter can be read back with these registers (either the internal position counter or ABR counter depending on bit 14 of register "Control word" on page 65).

The following two registers are provided for the motor:

- Homed zero position for cyclic counter
- Homed zero position for acyclic counter

Data type	Values
DINT	-2,147,483,648 to 2,147,483,647

5.10.6 Positioning and speed

5.10.6.1 Set position/speed

Name:
AbsPos01

This register is used to set position or speed, depending on the operating mode.

- Position mode (see "Mode" on page 62): Cyclic setting of the position setpoint in microsteps. In this mode, one micro-step is always 1/256 full-step.
- Speed mode (see "Mode" on page 62): In this mode, this register is considered a signed speed setpoint.

Data type	Values
DINT	-2,147,483,648 to 2,147,483,647

5.10.6.2 Current position - Cyclic

Name:
AbsPos01ActVal

This cyclic register contains the current position.

Default: Value of the internal position counter, possible to switch to ABR counter

Data type	Values
INT	-32768 to 32767

5.10.6.3 Current position - Acyclic

Name:
AbsPos1ActValAcyclic

This acyclic register contains the current position.

Default: Value of the ABR counter, can be changed to internal position counter

Data type	Values
DINT	-2,147,483,648 to 2,147,483,647

5.11 Minimum cycle time

The minimum cycle time specifies how far the bus cycle can be reduced without communication errors occurring. It is important to note that very fast cycles reduce the idle time available for handling monitoring, diagnostics and acyclic commands.

	Minimum cycle time
All function models	400 µs

5.12 Minimum I/O update time

The minimum I/O update time specifies how far the bus cycle can be reduced so that an I/O update is performed in each cycle.

	Minimum I/O update time
All function models (except ramp)	400 µs
Function model "Ramp"	
Inputs	400 µs
Outputs[1]	25 ms

1) Depends on the configuration of the "movement profile generator" on page 48