X20CM1201

1 General information

The module can be used to configure and carry out simple movements. For this purpose, the module has one AB encoder input and a total of 8 digital channels. Four of them are inputs, and the other 4 can be set as either an input or an output. Various output bit patterns are stored directly in the module.

The module is perfectly suited for easy to create drive control tasks for program and event controlled motor movements. Feed movements using drives with 2 speeds and forward/reverse movement are created easily and efficiently.

- Command-dependent digital pattern output
- Counter-dependent output circuit
- Event-controlled abort criteria
- 4 digital inputs
- 4 digital channels, configurable as inputs or outputs

2 Order data

Model number	Short description	Figure
	Digital signal processing and preparation	
X20CM1201	X20 combination module, 1 AB incremental encoder, 24 V, 4 digital inputs 24 V, 4 channels 24 V configurable as inputs or outputs, flexible digital controller logic	
	Required accessories	
	Bus modules	
X20BM11	X20 bus module, 24 VDC keyed, internal I/O supply continuous	
X20BM15	X20 bus module, with node number switch, 24 VDC keyed, internal I/O supply continuous	
	Terminal blocks	
X20TB12	X20 terminal block, 12-pin, 24 VDC keyed	

Table 1: X20CM1201 - Order data

3 Technical data

Model number	X20CM1201
Short description	
I/O module	1 AB incremental encoder, 24 V, 4 digital inputs, 4 channels configurable as inputs or outputs
General information	
Input voltage	24 VDC -15 % / +20 %
B&R ID code	0x21EF
Status indicators	I/O function per channel, operating state, module status
Diagnostics	
Module run/error	Yes, using status LED and software
Outputs	Yes, using the status LED and software (output error status)
Power consumption	
Bus	0.01 W
Internal I/O	1.5 W
Additional power dissipation caused by actuators (resistive) [W]	-
Type of signal lines	Shielded cables must be used for all signal lines.
Certifications	
CE	Yes
KC	Yes
EAC	Yes
UL	cULus E115267 Industrial control equipment
HazLoc	cCSAus 244665 Process control equipment for hazardous locations Class I, Division 2, Groups ABCD, T5
ATEX	Zone 2, II 3G Ex nA nC IIA T5 Gc IP20, Ta (see X20 user's manual) FTZÚ 09 ATEX 0083X
Digital inputs	
Quantity	4 + 4 additional channels, configurable as inputs or outputs
Nominal voltage	24 VDC
Input voltage	24 VDC -15 % / +20 %
Input current at 24 VDC	Approx. 1.3 mA
Input circuit	Sink
Input filter	
Hardware	≤2 µs
Software	-
Connection type	1-wire connections
Input resistance	18.4 kΩ
Switching threshold	
Low	<5 VDC
High	>15 VDC
Isolation voltage between channel and bus	500 V_{eff}
AB incremental encoder	
Quantity	1
Encoder inputs	24 V, asymmetrical
Counter size	32-bit
Input frequency	Max. 100 kHz
Evaluation	4x
Encoder power supply	Module-internal, max. 600 mA
Overload characteristics of encoder power supply	Short circuit protection, overload protection
Digital outputs	
Quantity	Up to 4, configurable as inputs or outputs using software
Variant	Push / Pull / Push-Pull
Nominal voltage	24 VDC
Switching voltage	24 VDC -15 % / +20 %
Nominal output current	0.1 A
Total nominal current	0.4 A
Connection type	1-wire connections
Output circuit	Sink or source
Output protection	Thermal cutoff if overcurrent or short circuit occurs, integrated protection for switching inductances
Actuator power supply	Module-internal, max. 600 mA
Diagnostic status	Output monitoring
Leakage current when switched off	Max. 25 µA
Residual voltage	<0.9 V at 0.1 A rated current
Peak short-circuit current	<10 A
Switch-on in the event of overload shutdown or short-circuit shutdown	Approx. 10 ms (depends on the module temperature)
Switching delay	
0 → 1	<2 µs
1 → 0	<2 µs

Table 2: X20CM1201 - Technical data

Model number	X20CM1201
Switching frequency	
Resistive load	Max. 24 kHz
Inductive load	See section "Switching inductive loads" (at 90% duty cycle).
Braking voltage when switching off inductive loads	Switching voltage + 0.6 VDC
Isolation voltage between channel and bus	500 V_{eff}
Electrical properties	
Electrical isolation	Channel isolated from bus Channel not isolated from channel
Operating conditions	
Mounting orientation	
Horizontal	Yes
Vertical	Yes
Installation elevation above sea level	
0 to 2000 m	No limitations
>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	-25 to 60°C
Vertical mounting orientation	-25 to 50°C
Derating	-
Storage	-40 to 85°C
Transport	-40 to 85°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 X20TB12 terminal block separately Order 1x X20BM11 bus module separately
Spacing	$12.5^{+0.2}$ mm

Table 2: X20CM1201 - Technical data

4 LED status indicators

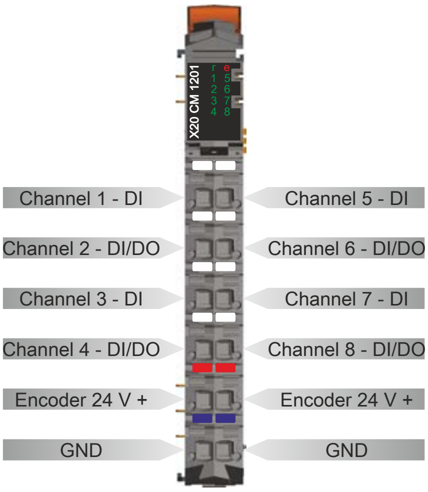
For a description of the various operating modes, see section "Additional information - Diagnostic LEDs" of the X20 system user's manual.

Figure	LED	Color	Status	Description
	r	Green	Off	No power to module
			Single flash	RESET mode
			Double flash	BOOT mode (during firmware update)[1]
			Blinking	PREOPERATIONAL mode
			On	RUN mode
	e	Red	Off	No power to module or everything OK
			On	Error or reset status
	1 - 8	Green		Status of the corresponding digital signal

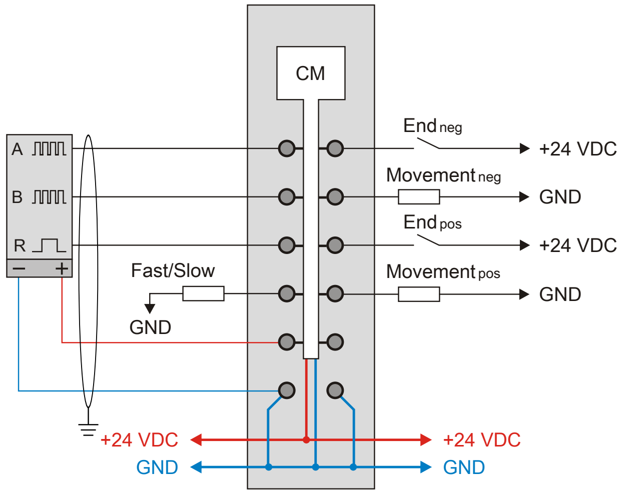
1) Depending on the configuration, a firmware update can take up to several minutes.

5 Pinout

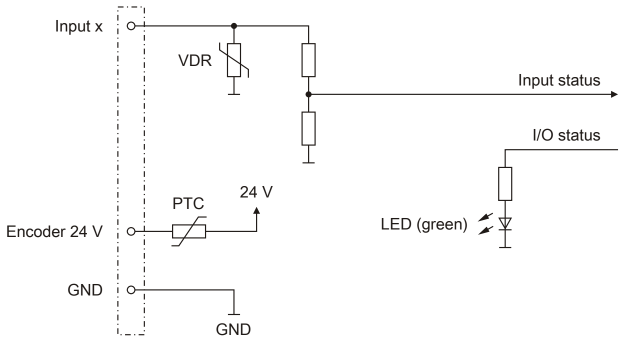
Shielded cables must be used for all signal lines.



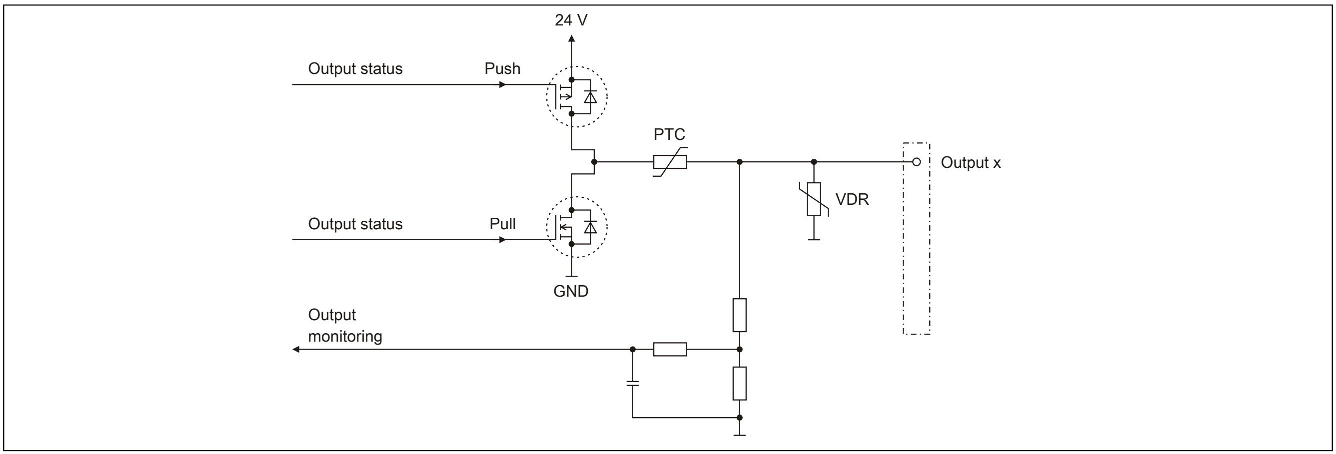
6 Connection example



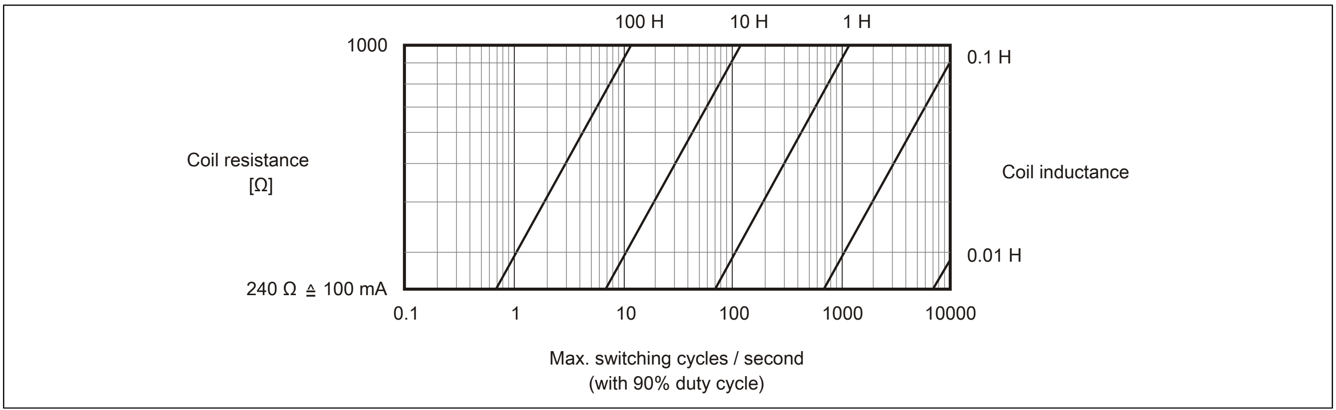
7 Input circuit diagram



8 Output circuit diagram



9 Switching inductive loads



10 Register description

10.1 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" of the X20 system user's manual.

10.2 Function model 0 - Standard

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Configuration						
130	CycleTimeCff					•
Communication - Command interface						
1	SendCommand	USINT			•	
3	SendCommandParam	USINT			•	
12	SendData	DINT			•	
1	ReadStatus	USINT	•			
3	ReadIndex	USINT	•			
12	ReadData	DINT	•			
Communication - Display register						
20	ABRposition	DINT	•			
28	TargetARBposition	DINT	•			
36	ErrorInfo	UDINT	•			
47	Displaying the digital inputs	USINT	•			
	DigitalInput01	Bit 0				
	...	...				
	DigitalInput08	Bit 7				
55	Status of encoder supply	USINT	•			
	PowerSupply01	Bit 0				

10.3 Function model 254 - Bus controller

Register	Offset[1]	Name	Data type	Read		Write	
				Cyclic	Acyclic	Cyclic	Acyclic
Configuration							
130	-	CycleTimeCff					•
Communication - Command interface							
1	1	SendCommand	USINT			•	
3	0	SendCommandParam	USINT			•	
12	4	SendData	DINT			•	
1	1	ReadStatus	USINT	•			
3	0	ReadIndex	USINT	•			
12	4	ReadData	DINT	•			
Communication - Display register							
20	-	ABRposition	DINT		•		
28	-	TargetABRposition	DINT		•		
36	-	ErrorInfo	UDINT		•		
47	-	Displaying the digital inputs	USINT		•		
		DigitalInput01	Bit 0				
		...	...				
		DigitalInput08	Bit 7				
55	-	Status of encoder supply	USINT		•		
		PowerSupply01	Bit 0				

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

10.3.1 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 additional registers and functions depending on the fieldbus used.

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

10.3.2 CAN I/O bus controller

The module occupies 1 analog logical slot on CAN I/O.

10.4 General

This is a low-end positioning module that supports 2 speed movements in positive and negative directions. No active position check is performed. The movements are started using a command interface and stopped by the position comparator (target position) or user-defined trigger conditions (input edge/ comparison). Every movement step is time-monitored. Up to 8 movement steps can be linked to form one continuous movement.

Position, input states and timeout periods are checked during each system cycle.

10.4.1 Types of movement

The module supports the following types of movement:

- Negative direction - fast
- Negative direction - slow
- Stop
- Positive direction - slow
- Positive direction - fast

The initial state of each type of movement is defined by the user. In order to avoid false input levels on the motor (caused by signal runtimes), and ensure timing (e.g. during direction change), there are additional command parameters to describe a operating mode change:

- 0x93 Negative directional setup state
- 0x88 Negative directional setup time
- 0x8A Negative directional stop time
- 0x95 Positive direction setup state
- 0x89 Positive direction setup time
- 0x8B Positive direction stop time
- 0x94 Stop state

Information:

No directional stop state is defined. To allow error handling, the directional stop state must be the same as STOP. Speed changes in the same direction of movement are not evaluated as changes in the operating mode of the movement.

10.4.2 Movement blocks

The module supports 4 movement blocks: Each movement block contains up to 8 movement steps. Each step is comprised of the following parameters:

- Target position - relative or absolute
- Timeout or delay
- Trigger condition - edge or comparator value (signal level)
- Error information

A block's movement steps can be executed as one continuous movement. The following parameters must be configured before the movement start command is issued:

- Step activation
- Step target position interpolation - relative or absolute
- Step speed - slow or fast
- Trigger mode - off or "Comparator value = true" or "Comparator value = false"

10.4.3 Movement generator

When a movement start command is issued, the mode of the active movement step is calculated based on the preceding target position. Step parameters may also be changed after the start as long as the step direction is not changed. Otherwise a movement error occurs. To ensure correct directional interpretation, the movement step position/range is limited to ±1073741824.

The target position of a step configured with a trigger is evaluated as the end position (error position). This means the position at the time of the trigger condition becomes the effective target position. Because this position is unknown when the calculation is made by the movement generator, the set end position is used for calculating the next absolute movement step. As a result, it is recommended to proceed with a relative movement step following a triggered step. A successive absolute movement step must be outside the positioning range of the triggered step.

If a movement step is configured as a standstill, i.e. relative position = 0, or the new absolute position = previous target position, a delay has occurred. If no trigger is configured, the parameter step timeout is evaluated as a simple delay time and not as an error state.

10.4.4 Tolerance monitoring

The module constantly monitors the position tolerance, even when no movements are active. Jitter and overshoot tolerances must be configured for both directions. Depending on the previous movement direction, a tolerance window is calculated based on the current target position. Because the movement generator uses the last target position, movements within the tolerance window must be avoided to prevent errors from occurring.

10.4.5 Homing

Homing is not implemented in this module as a movement function. The target position of a completed movement can be applied as the home position via command.

10.4.6 Safety monitoring

Safety monitoring refers to the hardware limit switches and software limit positions.

A safe input status (masks and comparator values) for positive and negative movements must be configured. Software end positions – minimum and maximum positions – can also be configured for both directions.

The module monitors these two positions from the time the parameter 0x93 or 0x95 "Positive directional setup state" is set. Monitoring is ended when the parameter 0x94 "Stop state" is set.

Because a trigger condition aborts the movement step before a safety check, a hardware limit switch can also be used as a trigger condition without generating an error.

10.5 Command description

10.5.1 No action

This command can be used as a placeholder during development or to separate 2 identical commands.

Code	0x00
Parameter	0
Data 0 to 3	0

10.5.2 Configure display mode

This command can be used to configure how the values in the "ReadIndex" on page 18 and "ReadData" on page 18 registers are displayed. Up to 4 display values can be displayed simultaneously. Possible selections include the command parameters 0xC0 = current position, to 0xC3 = I/O states.

Code	0x01
Parameter	Display control: 0 Scheduler off; Data 0 used for display 1 Scheduler cycle = X2X cycle; The next display cycle starts with each X2X cycle 2 Scheduler cycle = Command cycle; The next display cycle starts with each completed command
Data 0	Parameter number of display cycle 1 (Default: 0xC0 = current position)
...	...
Data 3	Parameter number of display cycle 4 (Default: 0xC0 = current position)

10.5.3 Enables the interface

This command activates the movement interface. The status of the interface is displayed in the "ReadStatus" on page 17 register (bit 5). The interface is disabled following a reset. This is necessary to ensure a consistent parameter field.

Code	0x02
Parameter	0
Data 0 to 3	0

10.5.4 Configure parameters

Code	0x03
Parameter	See parameter list
Data 0 to 3	Parameter data

Parameter list

Address	Parameters	Information
Movement blocks		
0x00	"Motion block"	Block 1
0x20	"Motion block"	Block 2
0x40	"Motion block"	Block 3
0x60	"Motion block"	Block 4
Configuration		
0x80	"Jitter tolerance"	Jitter tolerance, must be a negative value
0x81	"Jitter tolerance"	Jitter tolerance, must be a positive value
0x82	"Overshoot tolerance"	Overshoot tolerance, must be a negative value
0x83	"Overshoot tolerance"	Overshoot tolerance, must be a positive value
0x84 - 0x87	Reserved	
0x88	"Setup time"	Setup time: negative direction
0x89	"Setup time"	Setup time: positive direction
0x8a	"Stop time"	Stop time: negative direction
0x8b	"Stop time"	Stop time: positive direction
0x8C - 0x8F	Reserved	
0x90	"Output configuration"	Output configuration
0x91	"Output states"	Output state: negative direction, high speed
0x92	"Output states"	Output state: negative direction, low speed
0x93	"Output states"	Output state: negative direction, setup
0x94	"Output states"	Output state: stop
0x95	"Output states"	Output state: positive direction, setup
0x96	"Output states"	Output state: positive direction, low speed
0x97	"Output states"	Output state: positive direction, high speed
0x98	"Safe inputs"	Safe input state: negative direction
0x99	"Safe inputs"	Safe input state: positive direction
0x9A - 0x9B	Reserved	
0x9C	"Safe position"	Safe minimum position: negative direction
0x9D	"Safe position"	Safe maximum position: negative direction
0x9E	"Safe position"	Safe minimum position: positive direction
0x9F	"Safe position"	Safe maximum position: positive direction
0xA0 - 0xBF	Reserved	
Status indicators		
0xC0	Current position	See register "ABRPosition" on page 18.
0xC1	Target position	See register "TargetABRposition" on page 18.
0xC2	Error information	See register "ErrorInfo" on page 19.
0xC3	I/O states	See register "DigitalInput0x" on page 19.
0xC4 - 0xFF	Reserved	

10.5.4.1 Motion block

Parameter addresses: 0x00 to 0x7F

Each motion block contains 8 steps with 4 parameters each.

Calculating step addresses
Step address = Motion block address + Step address offset

Step address offset Offset = (Step - 1) * 4	Step number
+ 0x0	Step 1
+ 0x4	Step 2
...	...
+ 0x18	Step 7
+ 0x1B	Step 8

4 parameters per step

Parameter address	Parameters
Step address + 0	"Position" (relative or absolute)
Step address + 1	"Timeout or delay"
Step address + 2	"Trigger condition" (edge or comparator value)
Step address + 3	"Error information" (read-only)

Example

Calculating the trigger parameter in the 5th step of the 3rd motion block.
Parameter address: 0x40 (motion block 3) + 0x10 (offset of step 5) + 0x02 (offset of parameter address) = 0x52

Position

This parameter specifies a position in steps.

Data type	Value	Information
DINT	-1073741824 to 1073741823	For a description of this, see "Movement generator" on page 8.

Timeout or delay

This parameter sets the time within which a specified position must be reached. A corresponding error is output in the event of a timeout. In the stop state, this parameter specifies the duration of the stop state.

Data type	Value	Information
DINT	0 to 2,147,483,647	Default: 50, time in µs

Trigger condition

Depending on bits 2 and 3 in the data structure of each motion block (see "Start a movement block" on page 14), either the "Edge" or "Comparator value" structure is selected as the trigger condition.

Edge

Bit	Description	Value	Information
0	Falling edge - channel 01	0	Disabled
		1	Enabled
...		...	
7	Falling edge - channel 08	0	Disabled
		1	Enabled
8 - 15	Reserved	0	
16	Rising edge - channel 01	0	Disabled
		1	Enabled
...		...	
23	Rising edge - channel 08	0	Disabled
		1	Enabled
24 - 31	Reserved	0	

Comparator value

Bit	Description	Value	Information
0	Enable mask - channel 01	0	Disabled
		1	Enabled
...		...	
7	Enable mask - channel 08	0	Disabled
		1	Enabled
8 - 15	Reserved	0	
15	Comparative state - channel 01	0 or 1	
...			
23	Comparative state - channel 08	0 or 1	
24 - 31	Reserved	0	

Error information

This parameter contains the read back display value. For a description of the error information, see register "ErrorInfo" on page 19.
The parameter for the display command is 0.

10.5.4.2 Jitter tolerance

Parameter addresses: 0x80 to 0x81

This parameter determines the jitter range of the encoder.

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

10.5.4.3 Overshoot tolerance

Parameter addresses: 0x82 to 0x83

This parameter specifies the permitted tolerance range between the planned stop position and the stop position that can actually be reached in practice.

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

10.5.4.4 Setup time

Parameter addresses: 0x0x88 to 0x89

This parameter determines the length of the setup state between the stop time and the next positive or negative motion step.

Data type	Value	Information
DINT	0 to 2,147,483,647	Default: 50, time in µs

10.5.4.5 Stop time

Parameter addresses: 0x8A to 0x8B

This parameter defines the time of the stop state between 2 active motion steps.

Data type	Value	Information
DINT	0 to 2,147,483,647	Default: 50, time in µs

10.5.4.6 Output configuration

Parameter address: 0x90

This parameter defines the configuration of the digital outputs.

Bit	Description	Value	Information
0 - 1	Reserved	0	
2	Push driver - channel 02	0	Disabled
		1	Enabled
3	Pull driver - channel 02	0	Disabled
		1	Enabled
4 - 5	Reserved	0	
6	Push driver - channel 04	0	Disabled
		1	Enabled
7	Pull driver - channel 04	0	Disabled
		1	Enabled
8 - 9	Reserved	0	
10	Push driver - channel 06	0	Disabled
		1	Enabled
11	Pull driver - channel 06	0	Disabled
		1	Enabled
12 - 13	Reserved	0	
14	Push driver - channel 08	0	Disabled
		1	Enabled
15	Pull driver - channel 08	0	Disabled
		1	Enabled
16 - 31	Reserved	0	

10.5.4.7 Output states

Parameter addresses: 0x91 to 0x97

The output states of channels can be configured individually for each motion state (see "Types of movement" on page 7).

Bit	Description	Value	Information
0	Reserved	0	
1	Channel 02	0	No action
		1	Clear channel
2	Reserved	0	
3	Channel 04	0	No action
		1	Clear channel
4	Reserved	0	
5	Channel 06	0	No action
		1	Clear channel
6	Reserved	0	
7	Channel 08	0	No action
		1	Clear channel
8 - 16	Reserved	0	
17	Channel 02	0	No action
		1	Set channel
18	Reserved	0	
19	Channel 04	0	No action
		1	Set channel
20	Reserved	0	
21	Channel 06	0	No action
		1	Set channel
22	Reserved	0	
23	Channel 08	0	No action
		1	Set channel
24 - 31	Reserved	0	

10.5.4.8 Safe inputs

Parameter addresses: 0x98 to 0x99

This parameter enables the hardware limit switches according to their channel assignment and defines their logical state for starting a movement. The comparison state specifies the logical level on the input at which a motion is permitted or can be started.

Bit	Description	Value	Information
0	Enable mask - channel 01	0	Disabled
		1	Enabled
...		...	
7	Enable mask - channel 08	0	Disabled
		1	Enabled
8 - 15	Reserved	0	
15	Comparative state - channel 01	0 or 1	
...			
23	Comparative state - channel 08	0 or 1	
24 - 31	Reserved	0	

10.5.4.9 Safe position

Parameter addresses: 0x9C to 0x9F

These parameters define the minimum and maximum software limit positions of the path to be moved.

Data type	Value
DINT	-1073741824 to 1073741824

10.5.5 Configure counters

This command can be used to assign the hardware channels to the AB counter. With an ABR counter, the R input can be connected to any hardware channel as the trigger signal.

Code	0x04
Parameter	See parameter structure
Data 0	See data structure
Data 1 to 3	0

Parameter structure:

Bit	Description	Value	Information
0 - 1	Counter connection pair	00	Pair 1 (A: channel 01, B: channel 02)
		01	Pair 2 (A: channel 03, B: channel 04)
		10	Pair 3 (A: channel 05, B: channel 06)
		11	Pair 4 (A: channel 07, B: channel 08)
2 - 7	Reserved	0	

Data structure:

Bit	Description	Value	Information
0 - 1	Counter mode	00	AB encoder
		01	Up/down counter (A: timing, B: up/down signal)
		10	Edge counter - channel A
		11	Edge counter - channel B
2	Counting direction	0	Positive
		1	Negative
3 - 7	Reserved	0	

10.5.6 Homing

Assumes the target position of the last successful movement step as a reference position.

Code	0x05
Parameter	0
Data 0 to 3	Home position

10.5.7 Stops the movement.

The movement step in progress is stopped. This command always results in a movement error.

Code	0x06
Parameter	0
Data 0 to 3	0

10.5.8 Acknowledge movement error

The movement error is is cleared. If this command is executed when the error is still present, the current position is assumed as the target position. The basis of the relative position becomes unclear.

Code	0x07
Parameter	0
Data 0 to 3	0

10.5.9 Start a movement block

This command starts a movement block consisting of up to 8 steps.

Code	**0x08 (Block 1)** **0x09 (Block 2)** **0x0A (Block 3)** **0x0B (Block 4)**
Parameter	See parameter structure
Data 0 to 3	See data structure

Parameter structure:

Bit	Description	Value	Information
0	Step 1	0	No movement.
		1	Perform movement step.
...		...	
7	Step 8	0	No movement.
		1	Perform movement step.

Data structure:

Bit	Description	Value	Information
0	Step 1 position setting:	0	Relative
		1	Absolute
1	Step 1 speed:	0	Slow
		1	Fast
2 - 3	Step 1 trigger mode:	00	No trigger
		01	Edge trigger
		10	Comparator value "true"
		11	Comparator value "false"
4 - 7	Step 2	x	Like step 1 / Bits 0 to 2
...			
28 - 31	Step 8	x	Like step 1 / Bits 0 to 2

10.5.10 Selecting error information

At the end of each movement step, the command parameter "Addr + 3" (see "Motion blocks - Calculating the address" on page 9) can be used to read the error information selected in this register. This error information is displayed in registers "ReadIndex" on page 18 and "ReadData" on page 18.

Code	**0x00**
Parameter	0 Error information (default) 1 Timestamp 2 Current position 3 Target position
Data 0 to 3	0

10.6 Command interface

A command interface is available to the user. A command consists of:

- "Command" on page 16 (in the command description: code)
- "Command parameter" on page 16 (in the command description: parameters)
- "Command data" on page 17 (in the command description: Data 0 to 3)

The following commands can be executed:

- "No action" on page 8
- "Configure display mode" on page 8
- "Enable the interface" on page 9
- "Configure parameters" on page 9
- "Configure counters" on page 13
- "Perform homing" on page 13
- "Stop the movement" on page 13
- "Acknowledge movement error" on page 13
- "Start a movement block" on page 14
- "Select debug information" on page 14

The module returns:

- "System status" on page 17
- "Display parameter number" on page 18
- "Display data content" on page 18

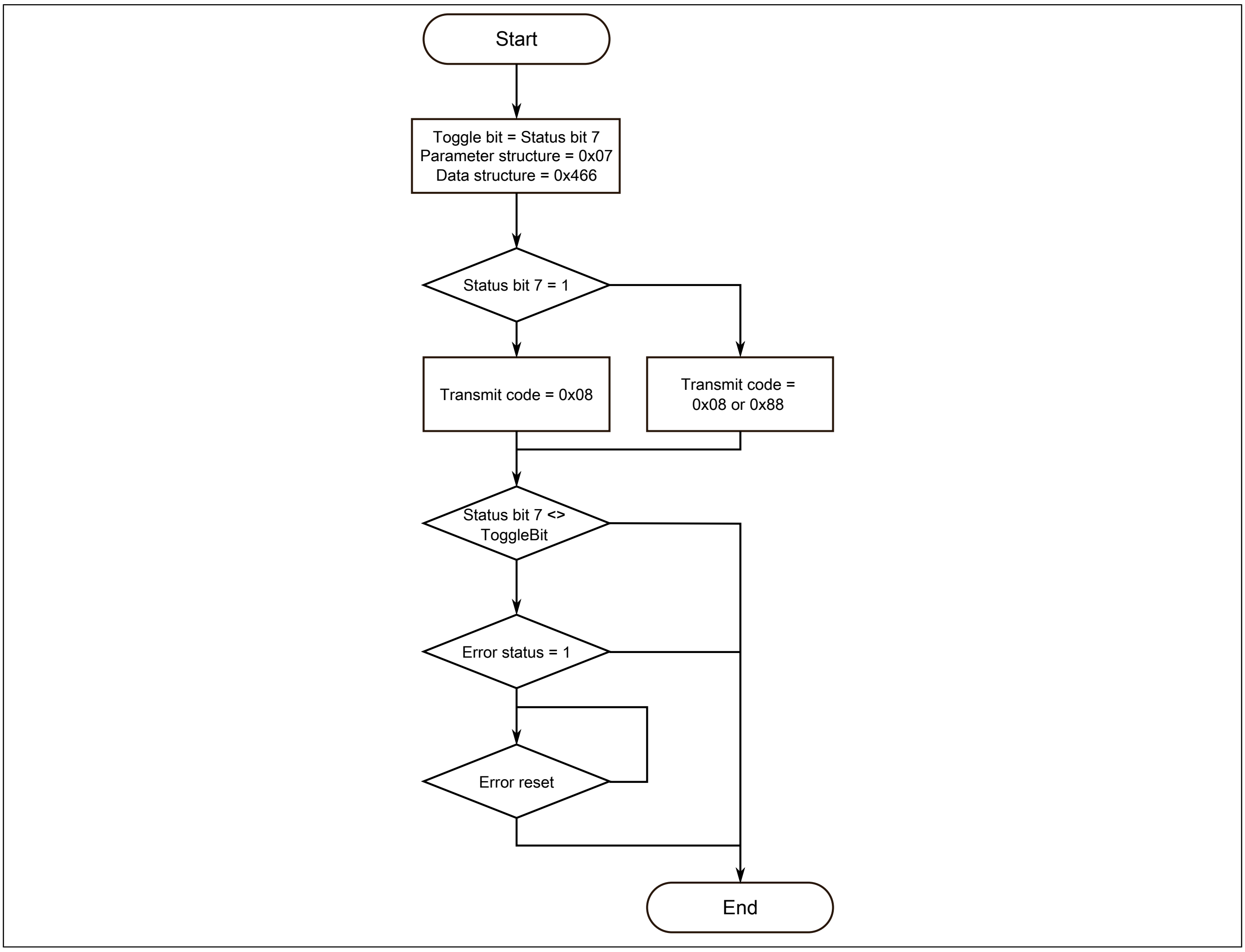
The module detects a new command through a change in the command register. The toggle bit must be changed in order to detect when the command issued in the "ReadStatus" on page 17 register is applied. Identical commands can be executed immediately following one another simply by changing the toggle bit.

10.6.1 Execution of a command

Commands must be sent by the application using the command interface. Due to the simple structure of the command interface, it is also possible to send them via CAN.

All commands are executed as follows:

1) Write command parameters (register "SendCommandParam" on page 16) and command data (register "SendData" on page 17).
2) Write command with changed toggle bit.
   When bit 7 in the command register (register "SendCommand" on page 16) is toggled, the module executes the command with the command parameter and command data.
3) Wait until bit 7 in the response register (register "ReadStatus" on page 17) matches bit 7 in the command register.
4) Read additional status information from the response register if necessary.
5) If additional commands should be sent, proceed with step 1.



10.6.2 Send command

Name:
SendCommand

The commands described under "Command description" on page 8 can be sent from this register. Bit 7 must be toggled to apply the commands.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 6	Command code	x	
7	Toggle bit for applying a new command	x	

10.6.3 Send command parameters

Name:
SendCommandParam

Specific parameters for the command to be sent must be entered in this register. The required parameters are listed under "Command description" on page 8 for the respective commands.

Data type	Value	Information
USINT	x	Command parameter

10.6.4 Send command data

Name:
SendData

Specific parameters for the command to be sent must be entered in this register. The required data is listed under "Command description" on page 8 for the respective commands.

Data 0 to 3 are sent as a single DINT value. The following structure is used:

	Data 3	Data 2	Data 1	Data 0	
Bit	31	24	16	8	0

Data type	Value	Information
DINT	x	Command data 0 to 3

10.6.5 Read status

Name:
ReadStatus

The commands and the current status can be checked in this register. Bit 7 can be used to check whether an issued command has been applied.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Reserved	0	
2	Position	0	Not yet reached
		1	Reached
3	Motion	0	In motion
		1	completed
4	Numerator	0	Not yet configured
		1	Configured
5	Interface	0	Not enabled
		1	Enabled
6	Command	0	No error
		1	Error occurred
7	Command toggle bit	x	Value that was read

10.6.6 Read parameter number

Name:
ReadIndex

This register displays the parameter number returned in response to a display command. See "Configure display mode" on page 8 and "Selecting error information" on page 14.

Data type	Value	Information
USINT	x	Parameter numbers

10.6.7 Read parameter data

Name:
ReadData

This register displays the parameter data returned in response to a display command. See "Configure display mode" on page 8 and "Selecting error information" on page 14.

Data type	Value	Information
DINT	x	Parameter data

10.6.8 Special display parameters

The following 4 registers correspond to display parameters 0xC0 to 0xC3 in the command description "Configure parameters" on page 9. This frees up the "ReadData" on page 18 register for other data.

10.6.8.1 Indicates the current position.

Name:
ABRPosition

This register shows the current position in the current step. It corresponds with the parameter 0xC0 in "Configure parameters" on page 9.

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

10.6.8.2 Indicates the current target position

Name:
TargetABRposition

This register shows the target position of the current step. It corresponds with the parameter 0xC1 in "Configure parameters" on page 9.

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

10.6.8.3 Displays the error information

Name:
ErrorInfo

This register displays error information. It corresponds with the parameter 0xC2 in "Configure parameters" on page 9.

Data type	Values
UDINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Tolerance error - negative	0	No error
		1	Error occurred
1	Tolerance error - positive	0	No error
		1	Error occurred
2	Timeout	0	No timeout
		1	Timeout
3 - 7	Reserved	0	
8	Safety monitoring error - Inputs (hardware limit switch)	0	No error
		1	Error occurred
9	Safety monitoring error - Position (software limit position)	0	No error
		1	Error occurred
10 - 15	Reserved	0	
16 - 18	Error status information	000	Reserved
		001	Negative directional stop state
		010	Negative movement
		011	Negative directional setup state
		100	Stop state
		101	Positive directional setup state
		110	Positive movement
		111	Positive directional stop state
19	Reserved	0	
20 - 24	Invalid step number	000 to 111	Number of the step that does not contain any movement information.
		1000	Inactive motion step (tolerance check)
25 - 31	Reserved	0	

10.6.8.4 Displaying the digital inputs

Name:
DigitalInput01 to DigitalInput08

This register displays the status of the digital inputs or read back outputs. It corresponds with the parameter 0xC3 in "Configure parameters" on page 9.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	DigitalInput01	0 or 1	Input status - channel 1
...		...	
7	DigitalInput08	0 or 1	Input status - channel 8

10.7 Sample configurations

10.7.1 Movement example

The channels have been set as follows for this example:

Hardware channel	Direction	Function assignment
1	Input	ABR encoder - signal A
2	Input	ABR encoder - signal B
3	Input	ABR encoder - signal R
4	Output	Fast speed
5	Input	Negative limit switch
6	Output	Negative direction
7	Input	Positive limit switch
8	Output	Positive direction

Enable interface

	Value	Description
Code	0x02	
Parameter	0	
Data 0 to 3	0	

Configure parameters

	Value
Code	0x03
Parameter	Parameter numbers
Data 0 to 3	Parameter data

The following parameters must be configured:

Code	Parameter	Data	Description
0x03	0x80	APPL	Negative jitter tolerance [µs] (application-specific)
0x03	0x81	APPL	Positive jitter tolerance [µs] (application-specific)
0x03	0x82	APPL	Negative overshoot tolerance [µs] (application-specific)
0x03	0x83	APPL	Positive overshoot tolerance [µs] (application-specific)
0x03	0x88	APPL	Negative setup time [µs] (application-specific)
0x03	0x89	APPL	Positive setup time [µs] (application-specific)
0x03	0x8A	APPL	Negative stop time [µs] (application-specific)
0x03	0x8B	APPL	Positive stop time [µs] (application-specific)
0x03	0x90	0x0000CCC0	Output configuration: Channel 04, channel 06, channel 08 as push/pull outputs
0x03	0x91	0x00280080	Output states - fast negative movement: Set channels 04 and 06, clear channel 08
0x03	0x92	0x00200088	Output states - slow negative movement: Set channel 06, clear channels 04 and 08
0x03	0x93	0x000000A8	Output states - negative setup: Clear channels 04, 06 and 08
0x03	0x94	0x000000A8	Output states - stop: Clear channels 04, 06 and 08
0x03	0x95	0x000000A8	Output states - positive setup: Clear channels 04, 06 and 08
0x03	0x96	0x00800028	Output states - slow positive movement: Set channel 08, clear channels 04 and 06
0x03	0x97	0x00880020	Output states - fast positive movement: Set channels 04 and 08, clear channel 06
0x03	0x98	0x00000014	Safe input state - negative: Channel 03 active, status of channel 03 (level) = 0, Channel 05 active, status of channel 05 (level) = 0
0x03	0x99	0x00000044	Safe input state - positive: Channel 03 active, status of channel 03 (level) = 0, Channel 07 active, status of channel 07 (level) = 0

Configure counters

	Value	Description
Code	0x04	
Parameter	0x00	Counter pair 1
Data 0	0	AB encoder, positive direction
Data 1 to 3	0	

10.7.2 Homing example

Configure parameters

	Value
Code	0x03
Parameter	Parameter numbers
Data 0 to 3	Parameter data

The following parameters must be configured:

Code	Parameter	Data	Description
Step 1			
0x03	0x00	0x3FFFFFFF	Position positive (relative)
0x03	0x01	0x01C9C380	Timeout (30 s)
0x03	0x02	0x00400000	Trigger condition (edge of channel 07 = 1)
Step 2			
0x03	0x04	0xC0000001	Position negative (relative)
0x03	0x05	0x01C9C380	Timeout (30 s)
0x03	0x06	0x00100000	Trigger condition (edge of channel 05 = 1)
Step 3			
0x03	0x08	0x3FFFFFFF	Position positive (relative)
0x03	0x09	0x01C9C380	Timeout (30 s)
0x03	0x0A	0x00040000	Trigger condition (edge of channel 03 = 1)

Start movement

	Value	Description
Code	0x08	Block 1
Parameter	0x07	Activate steps 1 to 3
Data 0 to 3	0x00000**466**	Step 1: Relative, fast, trigger on edge of channel 07 Step 2: Relative, fast, trigger on edge of channel 05 Step 3: Relative, slow, trigger on edge of channel 03

Wait until the movement is complete.

Homing

	Value	Description
Code	0x05	
Parameter	0	
Data 0 to 3	x	Home position

10.7.3 Standard positioning example

Configure parameters

	Value
Code	0x03
Parameter	Parameter numbers
Data 0 to 3	Parameter data

The following parameters must be configured:

Code	Parameter	Data	Description
0x03	0x00	X1	Pre-stop position
0x03	0x04	X2	Stop position

Start movement

	Value	Description
Code	0x08	Block 1
Parameter	0x03	Activate steps 1 and 2
Data 0 to 3	0x000000**11**	Step 1: absolute, slow, trigger off Step 2: absolute, slow, trigger off

10.7.4 Standard positioning example with stop

Configure parameters

	Value
Code	0x03
Parameter	Parameter numbers
Data 0 to 3	Parameter data

The following parameters must be configured:

Code	Parameter	Data	Description
0x03	0x00	X1	Pre-stop position
0x03	0x04	X2	Stop position
0x03	0x08	0	Relative movement
0x03	0x09	T_STOP	Stop delay [µs]

Start movement

	Value	Description
Code	0x08	Block 1
Parameter	0x07	Activate steps 1 to 3
Data 0 to 3	0x00000**011**	Step 1: absolute, slow, trigger off Step 2: absolute, slow, trigger off Step 3: relative, trigger off

10.7.5 Example code for a basic configuration

```
(**********************************************************************
 * Data object file: X20CM1201
 **********************************************************************)
;  +-- Code
;  |       +-- Parameter
;  |       |      +-- Data
;  |       |      |             +-- Description
;  |       |      |             |
  $0002, $0000, 00000000     ; Enable interface

 ;-------------------------------------------------------------------------------------------------
 ;CONFIGURATION
 ;-------------------------------------------------------------------------------------------------
 ;Movement tolerances
  $0003, $0080, -00000100    ; Negative jitter tolerance [steps]
  $0003, $0081,  00000100    ; Positive jitter tolerance [steps]
  $0003, $0082, -00000100    ; Negative overshoot tolerance [steps]
  $0003, $0083,  00000100    ; Positive overshoot tolerance [steps]

 ;Time for safe operating mode change
  $0003, $0088, 01000000     ; Setup time for negative direction [µs]
  $0003, $0089, 01000000     ; Setup time for positive direction [µs]
  $0003, $008A, 01000000     ; Stop time for negative direction [µs]
  $0003, $008B, 01000000     ; Stop time for positive direction [µs]

 ;Digital outputs
  $0003, $0090, $0000CCC0    ; Channel04, Channel06, Channel08 as push/pull outputs

 ;Output states
  $0003, $0091, $00280080    ; Negative movement, fast: Set Channel04 and 06, clear Channel08
  $0003, $0092, $00200088    ; Negative movement, slow: Set Channel06, clear Channel04 and 08
  $0003, $0093, $000000A8    ; Negative setup: Clear Channel04, Channel06 and Channel08
  $0003, $0094, $000000A8    ; Stop: Clear Channel04, Channel06 and Channel08
  $0003, $0095, $000000A8    ; Positive setup: Clear Channel04, Channel06 and Channel08
  $0003, $0096, $00800028    ; Positive movement, slow: Set Channel08, clear Channel04 and 06
  $0003, $0097, $00880020    ; Positive movement, fast: Set Channel04 and 08, clear Channel06

 ;Safe input states, Channel03 = (R)Trigger, Channel05 and Channel07 = Hardware limit switches
  $0003, $0098, $00000014    ; Safe input state, negative:
                             ;   Channel03 active, Channel03 state (level) = 0,
                             ;   Channel05 active, Channel05 state (level) = 0
  $0003, $0099, $00000044    ; Safe input state, positive:
                             ;   Channel03 active state Channel03 (level) = 0,
                             ;   Channel07 active, Channel07 state (level) = 0

 ;Software limit position
```

```
 $0003, $009C, $C0000001    ; Safe minimum position in the negative direction (-1073741824)
 $0003, $009D, $3FFFFFFF    ; Safe maximum position in the negative direction (1073741823)
 $0003, $009E, $C0000001    ; Safe minimum position in the positive direction (-1073741824)
 $0003, $009F, $3FFFFFFF    ; Safe maximum position in the positive direction (1073741823)

;Counter
 $0004, $0000, $00000000    ; (Channel01 = A / Channel02 = B)
                            ; (Function = AB encoder, positive direction of rotation)

;Display mode
 $0001, $0001, $C3C2C1C0    ; I/O states, error information, target position, current position

;-----------------------------------------------------------------------------------------------
;BLOCK and STEP sequence example
;-----------------------------------------------------------------------------------------------
;Motion block 1 0x08 (homing with (R)Trigger)
;Step 1
 $0003, $0000, $3FFFFFFF    ; Position positive (relative)
 $0003, $0001, $01C9C380    ; Timeout (30 s)
 $0003, $0002, $00400000    ; Trigger condition (edge of Channel07 == 1)
;$0003, $0003, $00000000    ; Debug information (read-only)

;Step 2
 $0003, $0004, $C0000001    ; Position negative (relative)
 $0003, $0005, $01C9C380    ; Timeout (30 s)
 $0003, $0006, $00100000    ; Trigger condition (edge of Channel05 == 1)
;$0003, $0007, $00000000    ; Debug information (read only)

;Step 3
 $0003, $0008, $3FFFFFFF    ; Position positive (relative)
 $0003, $0009, $01C9C380    ; Timeout (30 s)
 $0003, $000A, $00040000    ; Trigger condition (edge of Channel03 == 1)
;$0003, $000B, $00000000    ; Debug information (read-only)

;Motion block 2 0x09
;Step 1
 $0003, $0020, $00000000
 $0003, $0021, $00000000
 $0003, $0022, $00000000
 $0003, $0023, $00000000

;Step 2
 $0003, $0024, $00000000
 $0003, $0026, $00000000

;Motion block 3 0x0A
;Step 1
 $0003, $0040, $00000000
 $0003, $0041, $00000000
 $0003, $0042, $00000000
 $0003, $0043, $00000000

;Step 2
 $0003, $0044, $00000000
 $0003, $0046, $00000000

;Motion block 4 0x0B
;Step 1
 $0003, $0060, $00000000
 $0003, $0061, $00000000
 $0003, $0062, $00000000
 $0003, $0063, $00000000

;Step 2
 $0003, $0064, $00000000
 $0003, $0066, $00000000

 $0000, $0000, $00000000     ; No action
```

10.8 General module register

10.8.1 Configures the system cycle time

Name:
CycleTimeCff

This register configures the module's system cycle time.

Data type	Value	Information
UINT	25 to 255	System cycle time in µs. Bus controller default setting: 50 µs

10.8.2 Status of encoder supply

Name:
PowerSupply01

This register shows the status of the integrated encoder supply. A faulty encoder power supply is displayed as a warning.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Name	Value	Information
0	PowerSupply01	0	24 VDC encoder power supply OK
		1	24 VDC encoder power supply faulty
1 - 7	Reserved	-	

10.9 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
100 µs

10.10 Minimum cycle time

The minimum cycle time specifies the time up to which 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
100 µs