X20(c)DC2395

1 General information

This module is a multifunctional counter module. It can be connected to one SSI encoder, one ABR encoder, two AB encoders or four event counters. Two outputs are available for pulse width modulation. The functions can also be mixed.

- 24 VDC encoder inputs
- SSI, ABR, AB or event counters for inputs
- Pulse width modulation for outputs
- 24 VDC and GND for encoder supply

Information:

This module is a multifunctional module. Some bus controllers only support the default function model.

Default function model:

- **2x event counter (24 V)**
- **2x PWM output (24 V)**

1.1 Coated modules

Coated modules are X20 modules with a protective coating for the electronics component. This coating protects X20c modules from condensation and corrosive gases.
The modules' electronics are fully compatible with the corresponding X20 modules.

For simplification purposes, only images and module IDs of uncoated modules are used in this data sheet.

The coating has been certified according to the following standards:

- Condensation: BMW GS 95011-4, 2x 1 cycle
- Corrosive gas: EN 60068-2-60, method 4, exposure 21 days

1.2 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	Installation / EMC guide

2 Order data

Order number	Short description	Figure
	Counter functions	
X20DC2395	X20 digital counter module, 1 SSI absolute encoder, 24 V, 1 ABR incremental encoder, 24 V, 2 AB incremental encoders, 24 V, 4 event counters or 2 PWM, local time measurement function	
X20cDC2395	X20 digital counter module, coated, 1 SSI absolute encoder, 24 V, 1 ABR incremental encoder, 24 V, 2 AB incremental encoders, 24 V, 4 event counters or 2 PWM, local time measurement function	
	Required accessories	
	Bus modules	
X20BM11	X20 bus module, 24 VDC keyed, internal I/O power supply connected through	
X20BM15	X20 bus module, with node number switch, 24 VDC keyed, internal I/O power supply connected through	
X20cBM11	X20 bus module, coated, 24 VDC keyed, internal I/O power supply connected through	
	Terminal blocks	
X20TB12	X20 terminal block, 12-pin, 24 VDC keyed	

Table 1: X20DC2395, X20cDC2395 - Order data

3 Technical description

3.1 Technical data

Order number	X20DC2395	X20cDC2395
Short description		
I/O module	1 SSI absolute encoders 24 V, 1 ABR incremental encoders 24 V, 2 AB incremental encoders 24 V, 4x event counter or 2x pulse width modulation, time measurement, relative timestamp	
General information		
Input voltage	24 VDC -15% / +20%	
B&R ID code	0x1CD4	0xE503
Status indicators	I/O function per channel, operating state, module status	
Diagnostics		
Module run/error	Yes, using LED status indicator and software	
Outputs	Yes, using LED status indicator and software (output state)	
Power consumption		
Bus	0.01 W	
Internal I/O	1.4 W	
Additional power dissipation caused by actuators (resistive) [W]	-	
Type of signal lines	Shielded lines must be used for all signal lines.	
Certifications		
CE	Yes	
ATEX	Zone 2, II 3G Ex nA nC IIA T5 Gc IP20, Ta (see X20 user's manual) FTZÚ 09 ATEX 0083X	
UL	cULus E115267 Industrial control equipment	
HazLoc	cCSAus 244665 Process control equipment for hazardous locations Class I, Division 2, Groups ABCD, T5	
DNV	Temperature: **B** (0 - 55°C) Humidity: **B** (up to 100%) Vibration: **B** (4 g) EMC: **B** (bridge and open deck)	
LR	ENV1	
KR	Yes	
ABS	Yes	
EAC	Yes	
KC	Yes	-
Incremental encoders		
Quantity	2	
Encoder inputs	24 V, asymmetrical	
Counter size	16/32-bit	
Input frequency	Max. 100 kHz	
Evaluation	4x	
Encoder power supply	Module-internal, max. 600 mA	
Overload characteristics of encoder power supply	Short-circuit proof, overload-proof	
SSI absolute encoder		
Quantity	1	
Encoder inputs	24 V, asymmetrical	
Counter size	32-bit	
Max. transfer rate	125 kbit/s	
Encoder power supply	Module-internal, max. 600 mA	
Coding	Gray/Binary	
CLK: Output current	Max. 100 mA	
Overload characteristics of encoder power supply	Short-circuit proof, overload-proof	
Event counters		
Quantity	4	
Nominal voltage	24 VDC	
Signal form	Square wave pulse	
Evaluation	Each edge, cyclic counter	
Input frequency	Max. 100 kHz	
Input current at 24 VDC	Approx. 1.3 mA	
Input resistance	18.4 kΩ	
Insulation voltage between channel and bus	500 V_{eff}	
Counter frequency	200 kHz	
Counter size	16/32-bit	
Input filter		
Hardware	≤2 µs	
Software	-	

Table 2: X20DC2395, X20cDC2395 - Technical data

Order number	X20DC2395	X20cDC2395
Switching threshold		
Low	<5 VDC	
High	>15 VDC	
Edge detection / Time measurement		
Possible measurements	Gate time, period duration, edge offset for various channels	
Measurements per module	Up to 9	
Measurements per channel	Up to 2	
Counter size	16-bit	
Counter frequency		
Internal	8 MHz, 4 MHz, 2 MHz, 1 MHz, 500 kHz, 250 kHz, 125 kHz, 62.5 kHz	
Signal form	Square wave pulse	
Measurement type	Continuous or triggered	
Digital outputs		
Quantity	2	
Variant	Push / Pull / Push-Pull	
Nominal voltage	24 VDC	
Switching voltage	24 VDC -15% / +20%	
Nominal output current	0.1 A	
Total nominal current	0.2 A	
Output circuit	Sink or source	
Output protection	Thermal shutdown in the event of overcurrent or short circuit, integrated protection for switching inductive loads	
Pulse width modulation [1]		
Period duration	41.6 µs to 1.36 s	
Factor for period duration	n/48000 s, n = 2 to 65535	
Pulse duration	0 to 100%	
Resolution for pulse duration	0.1%	
Actuator power supply	Module-internal, max. 600 mA	
Diagnostic status	Output monitoring	
Leakage current when the output is switched off	Max. 25 µA	
Residual voltage	<0.9 V at 0.1 A nominal 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	
Switching frequency		
Resistive load	Max. 24 kHz	
Inductive load	See section "Switching inductive loads".	
Braking voltage when switching off inductive loads	Switching voltage + 0.6 VDC	
Insulation voltage between channel and bus	500 V_{eff}	
Electrical properties		
Electrical isolation	Bus isolated from encoder and output Output not isolated from output and encoder Encoder not isolated from encoder	
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	-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	Up to 100%, 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 X20BM11 separately.	Order 1x terminal block X20TB12 separately. Order 1x bus module X20cBM11 separately.
Pitch	$12.5^{+0.2}$ mm	

Table 2: X20DC2395, X20cDC2395 - Technical data

1) Dead time when switching between push and pull: Max. 1.5 µs.

3.2 LED status indicators

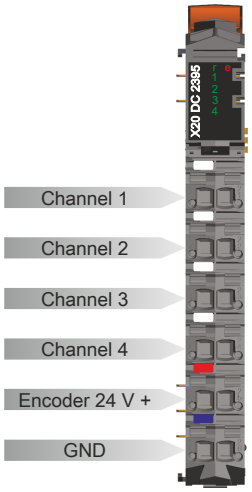
For a description of the various operating modes, see section "Additional information - Diagnostic LEDs" in the X20 system user's manual.

Figure	LED	Color	Status	Description
	r	Green	Off	No power to module
			Single flash	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 - 4	Green		Status of the corresponding digital signal

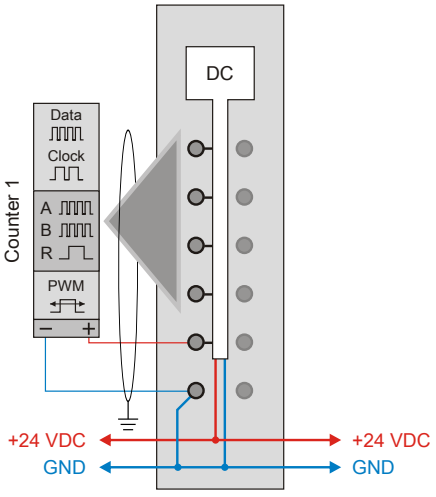
1) Depending on the configuration, a firmware update can take up to several minutes.

3.3 Pinout

Shielded cables must be used for all signal lines.



3.4 Connection example



3.5 Function overview

The following functions can be configured on the module. They cannot all be used at the same time due to the multiple use of the hardware channels and the limited cyclic data length.

- 4 digital channels, 2 of which can be configured as outputs
- 4 event counters with configurable counting direction and optional referencing via digital input
- 2 PWM outputs
- 2 up/down counters, each with optional latch inputs and comparator output
- 2 AB counters, each with optional latch inputs and comparator output
- 1 ABR encoder with configurable reference pulse edge and reference position, optional reference enable input, latch input and comparator output
- 1 SSI counter with optional latch input and comparator output
- 2 edge-triggered time measurement functions with configurable start edge based on current configuration settings

3.5.1 Description of channel assignments

The functions listed here are directly assigned to the respective hardware channels and cannot be changed:

Channel	Signal connections
1	• Digital input 1 • Event counter 1 • AB encoder 1 - signal line A • Up/down counter 1 - frequency • SSI encoder 1 - data line • ABR encoder 1 - signal line A
2	• Digital input 2 • Digital output 2 • Event counter 2 • PWM output 2 • AB encoder 1 - signal line B • Up/down counter 1 - direction • SSI encoder 1 - clock line • ABR encoder 1 - signal line B
3	• Digital input 3 • Event counter 3 • AB encoder 2 - signal line A • Up/down counter 2 - frequency • ABR encoder 1 - signal line R
4	• Digital input 4 • Digital output 4 • Event counter 4 • PWM output 4 • AB encoder 2 - signal line B • Up/down counter 2 - direction • ABR encoder 1 - reference enable input

Options available in addition to these basic functions, such as comparator outputs or latch inputs, can be configured freely to unused input/output channels.

3.5.2 Connection options

Channels 1 to 4 can be connected as follows:

Channel	Function					
1	I	Event counter	A	A	SSI data	
2	I/O	Event counter	B	B	SSI cycle	PWM
3	I	Event counter	A	R		
4	I/O	Event counter	B	Enable reference		PWM

The functions can also be mixed. For example:

Example 1	
Channel	**Function**
1	SSI data
2	SSI cycle
3	Event counter
4	PWM

Example 2	
Channel	**Function**
1	SSI data
2	SSI cycle
3	A
4	B

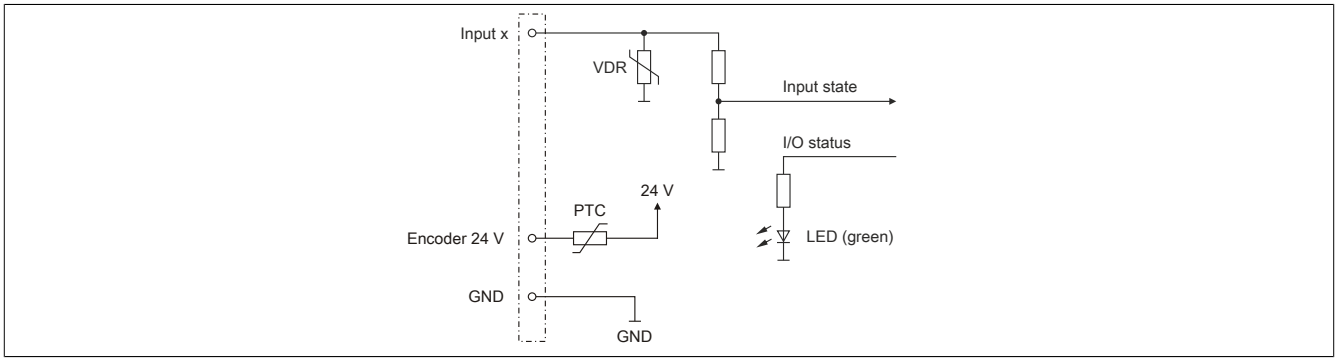
Example 3	
Channel	**Function**
1	Event counter
2	PWM
3	Event counter
4	PWM

Example 4	
Channel	**Function**
1	A
2	B
3	R
4	Enable reference

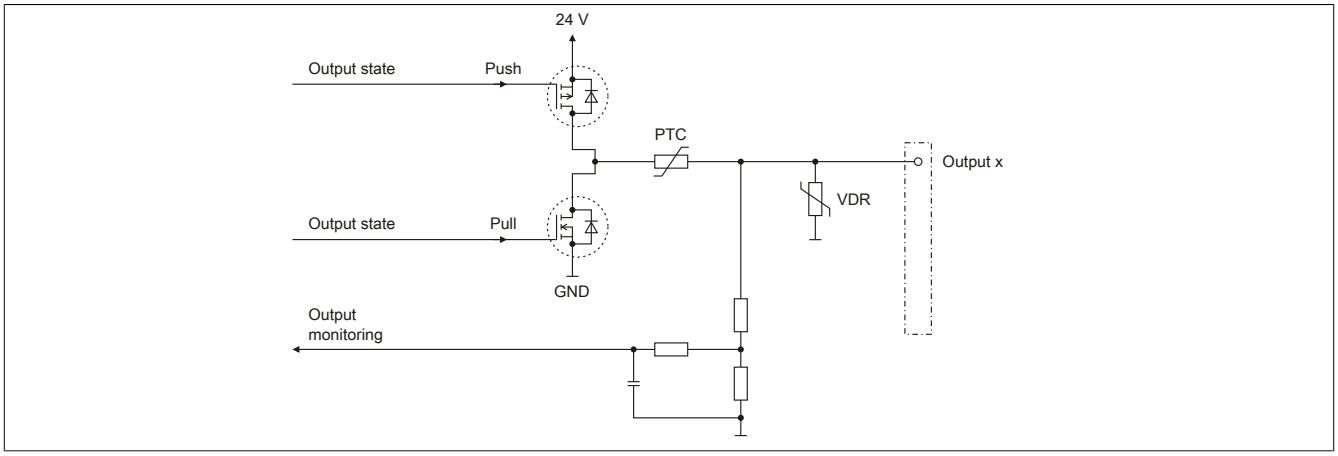
Example 5	
Channel	**Function**
1	A
2	B
3	Event counter
4	PWM

Example 6	
Channel	**Function**
1	Event counter
2	PWM
3	A
4	B

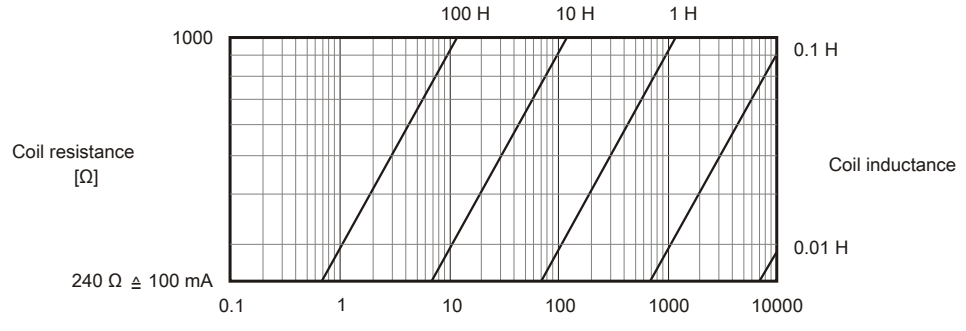
3.6 Input circuit diagram



3.7 Output circuit diagram



3.8 Switching inductive loads



3.9 Calculating the period duration

The outputs of the module can be operated as PWM outputs. The period duration is calculated using the following formula:

$$\text{Period duration} = \frac{n}{48000}\,\text{s}$$

A value of 2 to 65535 can be defined for n.

Example

n	Period duration	Frequency
2	416 µs	24 kHz
24000	500 ms	2 Hz
48000	1 s	1 Hz
65535	1.36 s	0.73 Hz

4 Register description

4.1 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" in the X20 system user's manual.

4.2 Function model 0 - Standard and Function model 1 - 32-bit counter

The following 2 models can be selected:

- 16-bit counter, Function model 0
- 32-bit counter, Function model 1 (identified in the table with a "(D)" in the data type and "(_32Bit)" in the name.)

The only difference between these two models is that they use either 16-bit or 32-bit registers for incremental counter functions. The following belong to this group:

- ABR encoders
- AB encoders
- Up/down counters
- Event counters

All other module functions e.g. SSI, PWM and time measurement, as well as their data types, are identical for the two models.

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Module configuration - General						
(N-1) * 2	CfO_CFGchannel0N (Index N = 1 to 4)	USINT				•
64 + N * 2	CfO_LEDNsource (Index N = 0 to 3)	USINT				•
Configuration - Input for ABR encoders						
512	CfO_DIREKTIOevent0IDwr	UINT				•
516	CfO_DIREKTIOevent0mode	USINT				•
522	CfO_DIREKTIOevent0compState	UINT				•
520	CfO_Ev0CompMask	USINT				•
2,064	CfO_Counter1PresetValue1(_32Bit)	U(D)INT				•
2,068	CfO_Counter1PresetValue2(_32Bit)	U(D)INT				•
2,320	CfO_Counter2PresetValue1(_32Bit)	U(D)INT				•
2,324	CfO_Counter2PresetValue2(_32Bit)	U(D)INT				•
2,048	CfO_Counter1config	USINT				•
2,056	CfO_Counter1configReg0	USINT				•
2,058	CfO_Counter1configReg1	USINT				•
2,112	CfO_Counter1event0IDwr	UDINT				•
2,120	CfO_Counter1event0config	UINT				•
2,144	CfO_Counter1event1IDwr	UINT				•
2,152	CfO_Counter1event1config	UINT				•
2,148	CfO_Counter1event1mode	USINT				•
Configuration - Inputs for AB, up/down and event counters						
2,048	CfO_Counter1config	USINT				•
2,056	CfO_Counter1configReg0	USINT				•
2,058	CfO_Counter1configReg1	USINT				•
2,112	CfO_Counter1event0IDwr	UDINT				•
2,120	CfO_Counter1event0config	UINT				•
2,116	CfO_Counter1event0mode	USINT				•
2,144	CfO_Counter1event1IDwr	UINT				•
2,152	CfO_Counter1event1config	UINT				•
2,148	CfO_Counter1event1mode	USINT				•
2,304	CfO_Counter2config	USINT				•
2,312	CfO_Counter2configReg0	USINT				•
2,314	CfO_Counter2configReg1	USINT				•
2,368	CfO_Counter2event0IDwr	UINT				•
2,376	CfO_Counter2event0config	UINT				•
2,372	CfO_Counter2event0mode	USINT				•
2,400	CfO_Counter2event1IDwr	UINT				•
2,408	CfO_Counter2event1config	UINT				•
2,404	CfO_Counter2event1mode	USINT				•
Configuration - Inputs for SSI encoders						
7,176	CfO_SSI1cfg	UINT				•

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
7,180	CfO_SSI1control	USINT				•
7,168	CfO_SSI1eventIDwr	UINT				•
7,232	CfO_SSI1event0IDwr	UINT				•
7,240	CfO_SSI1event0config	UINT				•
7,236	CfO_SSI1event0mode	USINT				•
7,172	ConfigAdvanced01	UDINT				•
Configuration - Comparator function for ABR, AB and SSI encoders as well as up/down counters						
256	CfO_OutClearMask	USINT				•
258	CfO_OutSetMask	USINT				•
1,024	CfO_DIREKTIOoutevent0IDwr	UINT				•
1,034	CfO_DIREKTIOoutsetmask0	USINT				•
1,032	CfO_DIREKTIOoutclearmask0	USINT				•
1,066	CfO_DIREKTIOoutsetmask1	USINT				•
1,064	CfO_DIREKTIOoutclearmask1	USINT				•
1,056	CfO_DIREKTIOoutevent1IDwr	UINT				•
Configuration - Outputs for PWM (pulse width modulation)						
6,144	CfO_PWM0prescaler	UINT				•
6,160	CfO_PWM1prescaler	UINT				•
Module communication - General						
40	Status of encoder power supply	USINT	•			
	PowerSupply01	Bit 0				
Communication - Digital inputs						
264	Input states of the channels	USINT	•			
	DigitalInput01	Bit 0				
	...	...				
	DigitalInput04	Bit 3				
Communication - Event counters						
2,080	EventCounter01	U(D)INT	•			
2,084	EventCounter02	U(D)INT	•			
2,336	EventCounter03	U(D)INT	•			
2,340	EventCounter04	U(D)INT	•			
Communication - Input for ABR encoders (optionally with comparator)						
2,080	ABREncoder01	(D)INT	•			
2,116	ReferenceModeABR01	USINT			•	
2,160	OriginComparator01	(D)INT			•	
2,164	MarginComparator01	U(D)INT			•	
264	Input states of the channels	USINT	•			
	ReferenceEnableSwitch01 (without comparator) ComparatorActualValue01 (with comparator)	Bit 3				
2,172	Latch01ABR01	(D)INT	•			
2,118	StatusABR01	USINT	•			
Communication - Input for AB						
2,080	ABEncoder01	(D)INT	•			
2,336	ABEncoder02	(D)INT	•			
2,160	OriginComparator01	(D)INT			•	
2,164	MarginComparator01	U(D)INT			•	
264	Input states of the channels	USINT	•			
	ComparatorActualValue01	Bit 3				
2,140	Latch01AB01	(D)INT	•			
2,172	Latch02AB01	(D)INT	•			
2,396	Latch01AB02	(D)INT	•			
2,428	Latch02AB02	(D)INT	•			
Communication - Up/down counters						
2,080	Counter01	U(D)INT	•			
2,336	Counter02	U(D)INT	•			
2,160	OriginComparator01	U(D)INT			•	
2,164	MarginComparator01	U(D)INT			•	
264	Input states of the channels	USINT	•			
	ComparatorActualValue01	Bit 3				
2,140	Latch01Counter01	U(D)INT	•			
2,172	Latch02Counter01	U(D)INT	•			
2,396	Latch01Counter02	U(D)INT	•			
2,428	Latch02Counter02	U(D)INT	•			
Communication - Input for SSI encoders						
7,184	SSIEncoder01	UDINT	•			
7,248	OriginComparator01	UDINT			•	
7,252	MarginComparator01	UDINT			•	
264	Input states of the channels	USINT	•			
	ComparatorActualValue01	Bit 3				
7,260	Latch01SSI01	UDINT	•			
Communication - Digital outputs						
260	Output states of the channels	USINT			•	
	DigitalOutput02	Bit 1				
	DigitalOutput04	Bit 3				

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
264	Input states of the channels	USINT	•			
	StatusDigitalOutput02	Bit 1				
	StatusDigitalOutput04	Bit 3				
Communication - Outputs for PWM (pulse width modulation)						
6,146	PWMOutput02	UINT			•	
6,162	PWMOutput04	UINT			•	
Configuration - Edge detection						
4,104	CfO_EdgeDetectFalling	USINT				•
4,106	CfO_EdeDetectRising	USINT				•
4,108	CfO_FallingDisProtection	USINT				•
4,110	CfO_RisingDisProtection	USINT				•
Configuration - Time measurement						
4,336	CfO_EdgeTimeglobalenable	USINT				•
4344 + N * 8	CfO_EdgeTimeFallingMode0N (Index N = 1 to 4)	UINT				•
4472 + N * 8	CfO_EdgeTimeRisingMode0N (Index N = 1 to 4)	UINT				•
Communication - Time measurement						
4,342	Trigger rising edge detection	USINT			•	
	TriggerRisingCH01	Bit 0				
	...	...				
	TriggerRisingCH04	Bit 3				
4,350	Show first rising trigger edge	USINT	•			
	BusyTriggerRisingCH01	Bit 0				
	...	...				
	BusyTriggerRisingCH04	Bit 3				
4,340	Trigger falling edge detection	USINT			•	
	TriggerFallingCH01	Bit 0				
	...	...				
	TriggerFallingCH04	Bit 3				
4,348	Show first falling trigger edge	USINT	•			
	BusyTriggerFallingCH01	Bit 0				
	...	...				
	BusyTriggerFallingCH04	Bit 3				
4474 + N * 8	CountRisingCH0N (Index N = 1 to 4)	USINT	•			
4476 + N * 8	TimeStampRisingCH0N (Index N = 1 to 4)	UINT	•			
4478 + N * 8	TimeDiffRisingCH0N (Index N = 1 to 4)	UINT	•			
4346 + N * 8	CountFallingCH0N (Index N = 1 to 4)	USINT	•			
4348 + N * 8	TimeStampFallingCH0N (Index N = 1 to 4)	UINT	•			
4350 + N * 8	TimeDiffFallingCH0N (Index N = 1 to 4)	UINT	•			

4.3 Function model 254 - Bus controller

Unlike the function models 0 and 1, this model only offers a selection of functions with a limited scope of configuration on the module.

The following functions are provided and can be run at the same time:

- 2 event counter with configurable counting direction
- 2 PWM outputs

Register	Offset[1)]	Name	Data type	Read		Write	
				Cyclic	Acyclic	Cyclic	Acyclic
Configuration							
(N-1) * 2	-	CfO_CFGchannel0N (Index N = 1 to 4)	USINT				•
64 + N * 2	-	CfO_LEDNsource (Index N = 0 to 3)	USINT				•
2,056	-	CfO_Counter1configReg0	USINT				•
2,312	-	CfO_Counter2configReg0	USINT				•
Communication							
2,080	0	EventCounter01	UINT	•			
2,336	2	EventCounter03	UINT	•			
6,146	0	PWMOutput02	UINT			•	
6,162	2	PWMOutput04	UINT			•	
40	4	Status of encoder power supply	USINT	•			
		PowerSupply01	Bit 0				

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

4.3.1 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 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).

4.3.2 CAN I/O bus controller

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

4.4 General module registers

4.4.1 Configuring LED status indicators

Name:
CfO_LED0source to CfO_LED3source

These registers can be used to define how the module's LED status indicators are used. Blinking patterns can be generated from the application, and the status of the physical inputs and outputs can be indicated.

Data type	Values	Bus controller default setting
USINT	See the bit structure.	CfO_LED**N**source N(0 to 3): 32 + N

Bit structure:

Bit	Description	Value	Information
0 - 3	MODE = 0	0	LED off
		1	Blinking quickly
		2	Blinking
		3	Blinking slowly
		4	Single flash
		5	Double flash
		6 to 15	Reserved
	MODE = 1 (inverted)	0	LED on
		1	Blinking quickly
		2	Blinking
		3	Blinking slowly
		4	Single flash
		5	Double flash
		6 to 15	Reserved
	MODE = 2	0 to 3	Number of the physical input channel
		4 to 15	Reserved
	MODE = 3	0 to 3	Number of the physical output channel
		4 to 15	Reserved
4 - 7	Selection of the mode for the LED status indicator	0	LED blinking pattern
		1	Inverted LED blinking pattern
		2	Displays a channel's physical input status
		3	Displays a channel's physical output status
		4 to 15	Reserved

4.4.2 Status of encoder power supply

Name:
PowerSupply01

This register indicates the state of the integrated encoder power supply. A faulty encoder power supply is output as a warning.

Data type	Values
USINT	See the bit structure.

Bit structure:

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

4.5 Digital inputs and outputs

4.5.1 Configure physical channels

Name:
CfO_CFGchannel01 to CfO_CFGchannel04

This register can be used to configure physical I/O channels 1 to 4.

> **Information:**
>
> **Except for bit 2 (inverted input), all other bits are only available for channels 2 and 4.**

Data type	Values	Bus controller default setting
USINT	See the bit structure.	CfO_CFGchannel0**N** N(1.3): 0 N(2.4): 99

Bit structure:

Bit	Description	Value	Information
0	Push[1)]	0	Disabled
		1	Enabled
1	Pull[1)]	0	Disabled
		1	Enabled
2	Inverted input	0	Disabled
		1	Enabled
3	Inverted output	0	Disabled
		1	Enabled
4 - 7	Output type	0	Direct I/O
		1 to 5	Reserved
		6	PWM (channel-specific)
		7	SSI clock (channel-specific)

1) To configure a channel as an output, Push and/or Pull must be enabled.

4.5.2 Reset mask of the digital channels

Name:
CfO_OutClearMask

The settings in this register only affect the values written to registers "DigitalOutput02 and 04" on page 15.

- 0 allows manual reset of digital outputs using registers DigitalOutput02 and 04
- 1 prevents manual reset of digital outputs using registers DigitalOutput02 and 04

When "1" is used, the output event function can be used to reset the outputs.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Reserved	-	
1	DigitalOutput02	0	Writing 0 to the DigitalOutput02 register resets the output
		1	Writing 0 from the DigitalOutput02 register does not reset the output
2	Reserved	-	
3	DigitalOutput04	0	Writing 0 to the DigitalOutput04 register resets the output
		1	Writing 0 from the DigitalOutput04 register does not reset the output
4 - 7	Reserved	-	

4.5.3 Set mask of the digital channels

Name:
CfO_OutSetMask

The settings in this register only affect the values written to registers "DigitalOutput02 and 04" on page 15.

- 0 allows manual setting of digital outputs using registers DigitalOutput02 and 04
- 1 prevents manual setting of digital outputs using registers DigitalOutput02 and 04

When "1" is used, the output event function can be used to set the outputs.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Reserved	-	
1	DigitalOutput02	0	Writing 1 to the DigitalOutput02 register sets the output
		1	Writing 1 from the DigitalOutput02 register does not set the output
2	Reserved	-	
3	DigitalOutput04	0	Writing 1 to the DigitalOutput04 register sets the output
		1	Writing 1 from the DigitalOutput04 register does not set the output
4 - 7	Reserved	-	

4.5.4 Input states of the channels

Name:
see "Name in the Automation Studio I/O configuration"

This register reads the input status of a physical channel. The polarity settings are accounted for in the value (bit 2 in "CfO_CFGchannel[x]" on page 13 register).

The bits in this register are shown in the Automation Studio I/O mapping table under different names based on the function used in order to improve readability.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Physical input channel	Value	Name in the Automation Studio I/O configuration
0	Channel 1	0 or 1	DigitalInput01
1	Channel 2	0 or 1	DigitalInput02 StatusDigitalOutput02
2	Channel 3	0 or 1	DigitalInput03
3	Channel 4	0 or 1	DigitalInput04 StatusDigitalOutput04 ReferenceEnableSwitch01 ComparatorActualValue01
4 - 7	Reserved	-	

4.5.5 Output states of the channels

Name:
DigitalOutput02 and DigitalOutput04

The output status of a physical channel can be written using this register. In order to configure a channel as an output:

1) Bit 0 "Push" and/or bit 1 "Pull" must be enabled in the "CfO_CFGchannel[x]" on page 13 register.
2) Bits 4 to 7 in the "CfO_CFGchannel[x]" on page 13 register must be set to Direct I/O.
3) 0 must be set for the respective channel in the "CfO_OutClearMask" on page 13 and "CfO_OutSetMask" on page 14 registers.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Reserved	-	
1	DigitalOutput02	0 or 1	Output status of channel 2
2	Reserved	-	
3	DigitalOutput04	0 or 1	Output status of channel 4
4 - 7	Reserved	-	

4.6 Event functions

The module provides configurable event functions. An event function can be connected to physical I/O and the values derived from them (e.g. counters) or be purely used for internal processing.

Every event function has event inputs and outputs. Event functions can also have only inputs or only outputs. Each event output has a unique event ID. It is possible to configure when an event is generated on an event output. The effect of the arrival of an event is specified by the event function.

Event functions can also be linked to one another. The link takes place using the event input. Every event input has a 16-bit register to which the event number of the linked event output is written.

Information:

The module functions that can be configured in the Automation Studio I/O configuration are primarily based on these event functions and their links. Changes in the Automation Studio I/O configuration have multiple effects on event functions and their links.

4.6.1 List of event IDs

Various hardware and software functions send event IDs or require event IDs in order to start. The following table shows all of the IDs available to configure the module.

Event ID	Description	
Direct event inputs		
512	Comparator condition	FALSE
513	Comparator condition	TRUE
Counter comparator functions		
2112	Counter function 1	Event function 1; FALSE
2113		Event function 1; TRUE
2144		Event function 2; FALSE
2145		Event function 2; TRUE
2368	Counter function 2	Event function 1; FALSE
2369		Event function 1; TRUE
2400		Event function 2; FALSE
2401		Event function 2; TRUE
Edge events		
4096	Falling edge on I/O channel	Channel 1
...		...
4099		Channel 4
4112	Rising edge on I/O channel	Channel 1
...		...
4115		Channel 4
4128	Rising or falling edge on I/O channel	Channel 1
...		...
4131		Channel 4
SSI counter events		
7168	SSI valid	
7169	SSI ready	
SSI comparator events		
7232	SSI 1 comparator condition	FALSE
7233		TRUE
Timer events		
208	Timer1	50 µs
209	Timer2	100 µs
210	Timer3	200 µs
211	Timer4	400 µs
212	Timer5	800 µs
213	Timer6	1600 µs
214	Timer7	3200 µs
215	Timer8	3200 µs (time offset to timer 7)
Network functions		
224	SOAISOP (**s**ynchronous **o**ut **a**synchronous **i**n **s**tart of **p**rotocol)	
225	AOSISOP (**a**synchronous **o**ut **s**ynchronous **i**n **s**tart of **p**rotocol)	
226	SOAIEOP (**s**ynchronous **o**ut **a**synchronous **i**n **e**nd of **p**rotocol)	
227	AOSIEOP (**a**synchronous **o**ut synchronous **i**n **e**nd of **p**rotocol)	
Idle event		
192	No-load operation	

Timer

There are 8 timer events that the module can generate.

> **Information:**
>
> **The timers have the highest event priority. All other system functions are interrupted when a timer event occurs, and jitter for the amount of time it takes to process the event.**

Idle event

Idle time is the time that remains after the system has processed all higher priority events and operations. The module performs the following functions during idle time:

- Handling of the asynchronous protocol
- Mechanism for (re-)linking events
- Operation of LEDs
- Execution of event functions linked to the idle function

4.6.2 Edge events

For each physical input channel there are 3 event functions

- Falling edge
- Rising edge
- Falling and rising edge

The respective event is triggered when an edge is detected on the hardware input and the "CfO_EdgeDetectRising" on page 17 and/or "CfO_EdgeDetectFalling" on page 17 register has been configured for the respective channel.

Edges are detected by the hardware and processed for each interrupt. The interrupt handler uses an event distributor, which requires a specific amount of time for each edge to operate the hardware and execute linked event functions. To reduce this time, edge detection can be enabled/disabled individually for each channel. To optimize system load and I/O jitter, it is important to only enable edge detection where it is actually needed.

Information:

Edge detection can also be used for channels that are configured as outputs.

4.6.2.1 Event frequency limitation

To stabilize the system, there is a mechanism that limits the number of events created through edge recognition. After an edge event is processed, at least one idle event must occur before a new event is processed for the same edge.

The "CfO_FallingDisProtection" on page 18 and "CfO_RisingDisProtection" on page 18 registers can be used to disable this limitation for each edge, and then an event will be generated for every edge. However, this can cause a system overload, i.e. I/O operation can fail for up to 100 ms before the module changes to the reset state.

4.6.2.2 Generate event on falling edge

Name:
CfO_EdgeDetectFalling

This register defines whether an event is generated on a falling edge.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Channel 1	0	No event generated on falling edge.
		1	Events 4096 and 4128 are generated on falling edge.
...		...	
3	Channel 4	0	No event generated on falling edge.
		1	Events 4099 and 4131 are generated on falling edge.
4 - 7	Reserved	-	

4.6.2.3 Generate event on rising edge

Name:
CfO_EdgeDetectRising

This register defines whether an event is generated on a rising edge.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Channel 1	0	No event generated on rising edge.
		1	Events 4112 and 4128 are generated on rising edge.
...		...	
3	Channel 4	0	No event generated on rising edge.
		1	Events 4115 and 4131 are generated on rising edge.
4 - 7	Reserved	-	

4.6.2.4 Enable limit for falling edges

Name:
CfO_FallingDisProtection

This register can be used to enable/disable the event frequency limit for falling edges on the respective channel.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Channel 1	0	Event frequency limit enabled.
		1	Event frequency limit disabled.
...		...	
3	Channel 4	0	Event frequency limit enabled.
		1	Event frequency limit disabled.
4 - 7	Reserved	-	

4.6.2.5 Enable limit for rising edges

Name:
CfO_RisingDisProtection

This register can be used to enable/disable the event frequency limit for rising edges on the respective channel.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Channel 1	0	Event frequency limit enabled.
		1	Event frequency limit disabled.
...		...	
3	Channel 4	0	Event frequency limit enabled.
		1	Event frequency limit disabled.
4 - 7	Reserved	-	

4.6.3 Direct input function

The module features a direct input function.

This event function is based on comparator functionality. If the event configured in the "CfO_DIREKTIOevent0IDwr" on page 19 register occurs, the event function compares the status of all Direct I/O channels enabled in the "CfO_EvCompMask" on page 20 register to a status defined in the "CfO_DIREKTIOeventcompState" on page 19 register. The event that is generated depends on the results of this comparison.

- If the respective bits are the same, then event number 513 is generated
- If the respective bits are different, then event number 512 is generated

4.6.3.1 Configure event ID for input function

Name:
CfO_DIREKTIOevent0IDwr

This register holds the event ID generated by the direct input function. For a list of all possible event IDs, see "List of event IDs" on page 16

Data type	Value	Information
INT	192 to 7,233	ID of event function

4.6.3.2 Configure the mode of the input function

Name:
CfO_DIREKTIOevent0mode

The mode in which the "direct input function" operates can be set in this register.

Comparator functions can be operated in 4 different modes. For a description, see "Comparator modes" on page 29.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Comparator mode	0	Off
		1	Individual
		2	State change
		3	Continuous
2 - 7	Reserved	-	

4.6.3.3 Comparator status for comparator mask

Name:
CfO_DIREKTIOevent0compState

This register contains the status bits that are compared with the bits specified in the "CfO_Ev0CompMask" on page 20 register, which contain the I/O input status, when an event is received.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Comparator status of channel 1	0 or 1	
...		...	
3	Comparator status of channel 4	0 or 1	
4 - 7	Reserved	-	

4.6.3.4 Configure the comparator mask for the input function

Name:
CfO_Ev0CompMask

If a bit is set, then the input status of the respective channel is compared with that bit in the "CfO_DIREKTIOeventcompState" on page 19 register.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Channel 1	0	Do not compare bit
		1	Compare bit in register
...		...	
3	Channel 4	0	Do not compare bit
		1	Compare bit in register
4 - 7	Reserved	0	

4.6.4 Direct output functions

The module has 2 of these event functions

The effect of executing this event function is similar to writing to the "DigitalOutput02 and 04" on page 15 registers. When this event function is triggered, however, the changed output states are passed on to the hardware immediately, regardless of the X2X cycle.

When this event function is used, the masks of the corresponding outputs (see registers "CfO_OutClearMask" on page 13 and "CfO_OutSetMask" on page 14) must be set to 1. Otherwise, the output state would constantly be overwritten by the values in registers "DigitalOutput02 and 04" on page 15.

4.6.4.1 Configure event ID for output function

Name:
CfO_DIREKTIOevent0IDwr to CfO_DIREKTIOevent1IDwr

These registers hold the event IDs that trigger the direct output function. For a list of all possible event IDs, see "List of event IDs" on page 16

Data type	Value	Information
INT	192 to 7,233	ID of event function

4.6.4.2 Configure channels for resetting

Name:
CfO_DIREKTIOoutclearmask0 to CfO_DIREKTIOoutclearmask1

Writing "1" to the bit position that corresponds to a channel resets the output if the output event function is being executed. This corresponds to writing "0" in registers "DigitalOutput 02 and 04" on page 15.

The bit that corresponds to channels that should be reset should be set to "1" in the "CfO_OutClearMask" on page 13 register.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Reserved	-	
1	Channel 2	0	Reset channel 2
		1	Do not reset channel 2
2	Reserved	-	
3	Channel 4	0	Reset channel 4
		1	Do not reset channel 4
4 - 7	Reserved	-	

4.6.4.3 Configure channels for setting

Name:
CfO_DIREKTIOoutsetmask0 to CfO_DIREKTIOoutsetmask1

Writing "1" to the bit position that corresponds to a channel sets the output if the output event function is being executed. This corresponds to writing "1" in registers "DigitalOutput 02 and 04" on page 15.

The bit that corresponds to channels that should be reset should be set to "1" in the "CfO_OutSetMask" on page 14 register.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Reserved	-	
1	Channel 2	0	Set channel 2
		1	Do not set channel 2
2	Reserved	-	
3	Channel 4	0	Set channel 4
		1	Do not set channel 4
4	Reserved	-	

4.7 Counters and encoders

The module has 2 internal counter functions, each with 2 event counter registers. Each of these 2 counters is permanently assigned to 2 physical inputs. This assignment cannot be changed.

The counter registers perform different functions based on how the event functions are connected. The counter registers can be configured in the following ways:

- ABR counter
- AB counter
- Up/down counters
- Event counters

Different names are used for them in Automation Studio and in the register description to improve clarity.

Channel	Counter function	Counter register	Name in Automation Studio
1	1	1	ABEncoder01 ABREncoder01 Counter01 EventCounter01
2		2	EventCounter02
3	2	1	ABEncoder02 Counter02 EventCounter03
4		2	EventCounter04

4.7.1 Counter value calculation

There are 3 steps for calculating the state of any counter function

1. The counter value is based on the 2 absolute value counters "abs1" and "abs2". They are only used internally in the module and cannot be read. Depending on the mode, these registers show the respective physical input signals.

	Mode		
	Edge counters	AB encoders	Up/down counter
abs1	Edges of counter channel 1	Increments in positive direction	Counter channel 2 = 0: Edges of counter channel 1 in up direction
abs2	Edges of counter channel 2	Increments in negative direction	Counter channel 2 = 1 Edges of counter channel 1 in down direction

2. From the absolute value registers "abs1" and "abs2", 2 more counters are formed: "counter 1" and "counter 2". These are only used internally in the module and cannot be read. The following values are used for the calculation:

- Absolute value registers "abs1" and "abs2"
- SW_reference_counter 1 and 2: This reference value can be defined by the "CfO_CounterPresetValue" on page 27 register to allow referencing <> 0.
- HW_reference_counter 1 and 2: In the "CfO_CounterEventMode" on page 30 register, you can configure whether latched values should be copied to these registers when counter events occur.

counter1 = abs1 + SW_reference_counter1 - HW_reference_counter1
counter2 = abs2 + SW_reference_counter2 - HW_reference_counter2

3. The counter registers contain the sum of the two internal counters "counter 1" and "counter 2". The "CfO_CounterConfigReg" on page 26 register allows you to define a sign for each "counter" register and define whether or not it should be used.

Counter register = counter1 + counter2

4.7.2 Sample configurations

All of the settings available in Automation Studio for AB encoders, ABR encoders, up/down counters and event counters are based on the two counter functions.
The following configuration examples show the values with which Automation Studio initializes the module registers in order to implement these functions.

4.7.2.1 I/O configuration - AB encoder

The following table shows how the module's various event functions can be linked in order to configure an AB encoder.

[x] stands for the respective counter function, either 1 or 2

Register	Value	Comment
For the function		
CfO_Counter[x]config	0x01	Mode = Up/down counter
CfO_Counter[x]configReg0	0x0D	Configure the calculation of the internal "counter1" and "counter2" registers (see "Counter value calculation" on page 22 and "Examples of calculation configurations" on page 26)
For the latch		
CfO_Counter[x]event0config	0x000D	Configuration of the calculation of the first value used for the latch
CfO_Counter[x]event0mode	0x03	Mode of the first counter event function - Continuous
CfO_Counter[x]event0IDwr	(any)	Number of the event that should trigger Latch 1 ("Latch 01 - Channel" in the Automation Studio I/O configuration).
CfO_Counter[x]event1config	0x0D	Configuration of the calculation of the second value used for the latch
CfO_Counter[x]event1mode	0x03	Mode of the second counter event function - Continuous
CfO_Counter[x]event1IDwr	(any)	Number of the event that should trigger Latch 2
For the comparator		
CfO_Counter1event1IDwr	0x00D0	Event number of Timer 1 (50 µs) **Information:** **The latch and comparator must not have the same event number!**
CfO_Counter1event1config	0x900D or 0xA00D	Configuration of the comparator for the second counter event
CfO_Counter1event1mode	0x03	Mode of the second counter event function - Continuous
CfO_DIREKTIOoutevent0IDwr	0x0861	TRUE event output of the second counter to trigger the direct output function (set outputs).
CfO_DIREKTIOoutsetmask0	0x08, 0x20, 0x80	Outputs that should be set when comparator condition = TRUE
CfO_DIREKTIOoutevent1IDwr	0x0860	FALSE event output of the second counter to trigger the direct output function (reset outputs).
CfO_DIREKTIOoutclearmask1	0x08, 0x20, 0x80	Outputs that should be reset when comparator condition = FALSE

4.7.2.2 I/O configuration - ABR encoder

The following table shows how the module's various event functions can be linked in order to configure an ABR encoder.

Register	Value	Comment
For the function		
CfO_Counter1PresetValue1	(any)	Desired offset value for referencing
CfO_Counter1event0IDwr	0x0201	Link between the first counter event and the direct input comparator condition TRUE
CfO_Counter1config	0x01	Mode = AB encoder
CfO_Counter1configReg0	0x0D	Configure the calculation of the internal "counter1" and "counter2" registers (see "Counter value calculation" on page 22 and "Examples of calculation configurations" on page 26)
CfO_DIREKTIOevent0IDwr	0x1002 or 0x1012	Selection of the desired input edge as trigger for the ABR encoder function
CfO_Counter1event0config	0x0000	Configuration of the first counter event (for referencing)
CfO_DIREKTIOevent0mode	0x03	Mode of the "direct input function" - Continuous
CfO_DIREKTIOevent0compState	0x00 or 0x08	Comparator status for the "direct input function"
CfO_Ev0CompMask	0x08	Comparator mask for the "direct input function"
For the latch		
CfO_Counter1event0config	0x000D	Configuration of the calculation of the value used for the latch
CfO_Counter1event0mode	0x03	Mode of the first counter event function - Continuous
CfO_Counter1event0IDwr	(any)	Number of the event that should trigger the latch
For the comparator		
CfO_Counter1event1IDwr	0x00D0	Event number of Timer 1 (50 µs) **Information:** **The latch and comparator must not have the same event number!**
CfO_Counter1event1config	0x900D or 0xA00D	Configuration of the comparator for the second counter event
CfO_Counter1event1mode	0x03	Mode of the second counter event function - Continuous
CfO_DIREKTIOoutevent0IDwr	0x0861	TRUE event output of the second counter to trigger the direct output function (set outputs).
CfO_DIREKTIOoutsetmask0	0x08, 0x20, 0x80	Outputs that should be set when comparator condition = TRUE
CfO_DIREKTIOoutevent1IDwr	0x0860	FALSE event output of the second counter to trigger the direct output function (reset outputs).
CfO_DIREKTIOoutclearmask1	0x08, 0x20, 0x80	Outputs that should be reset when comparator condition = FALSE

4.7.2.3 I/O configuration - Up/down counter

The following table shows how the module's various event functions can be linked in order to configure an up/down counter.

[x] stands for the respective counter function, either 1 or 2

Register	Value	Comment
For the function		
CfO_Counter[x]config	0x03	Counter mode = Up/down counter
CfO_Counter[x]configReg0	0x0D, 0x07	Configure the calculation of the internal "counter1" and "counter2" registers (see "Counter value calculation" on page 22 and "Examples of calculation configurations" on page 26)
For the latch		
CfO_Counter[x]event0config	0x0D, 0x07	Configuration of the calculation of the first value used for the latch
CfO_Counter[x]event0mode	0x03	Mode of the first counter function - Continuous
CfO_Counter[x]event0IDwr	(any)	Number of the event that should trigger Latch 1
CfO_Counter[x]event1config	0x0D, 0x07	Configuration of the calculation of the second value used for the latch
CfO_Counter[x]event1mode	0x03	Mode of the second counter function - Continuous
CfO_Counter[x]event1IDwr	(any)	Number of the event that should trigger Latch 2
For the comparator		
CfO_Counter1event1IDwr	0x00D0	Event number of Timer 1 (50 µs) **Information:** **The latch and comparator must not have the same event number!**
CfO_Counter1event1config	0x900D, 0xA00D or 0x9007, 0xA007	Configuration of the comparator for the second counter event
CfO_Counter1event1mode	0x03	Mode of the second counter event function - Continuous
CfO_DIREKTIOoutevent0IDwr	0x0861	TRUE event output of the second counter to trigger the direct output function (set outputs).
CfO_DIREKTIOoutsetmask0	0x08, 0x20, 0x80	Outputs that should be set when comparator condition = TRUE
CfO_DIREKTIOoutevent1IDwr	0x0860	FALSE event output of the second counter to trigger the direct output function (reset outputs).
CfO_DIREKTIOoutclearmask1	0x08, 0x20, 0x80	Outputs that should be reset when comparator condition = FALSE

4.7.2.4 I/O configuration - Event counter

The following table shows how the module's various event functions can be linked in order to configure an event counter.

[x] stands for the respective counter function, either 1 or 2

Register	Value	Comment
For event counters on channels 1 and 3		
CfO_Counter[x]configReg0	0x01 or 0x03	Configure the calculation of the internal "counter1" and "counter2" registers (see "Counter value calculation" on page 22 and "Examples of calculation configurations" on page 26)
CfO_Counter[x]event0mode	0x43	Mode of the first counter event function and referencing configuration
CfO_Counter[x]event0IDwr	(any)	Number of the event that should trigger referencing
For event counters on channels 2 and 4		
CfO_Counter[x]configReg1	0x04 or 0x08	Configure the calculation of the internal "counter1" and "counter2" registers (see "Counter value calculation" on page 22 and "Examples of calculation configurations" on page 26)
CfO_Counter[x]event1mode	0x83	Mode of the second counter event function and referencing configuration
CfO_Counter[x]event1IDwr	(any)	Number of the event that should trigger referencing

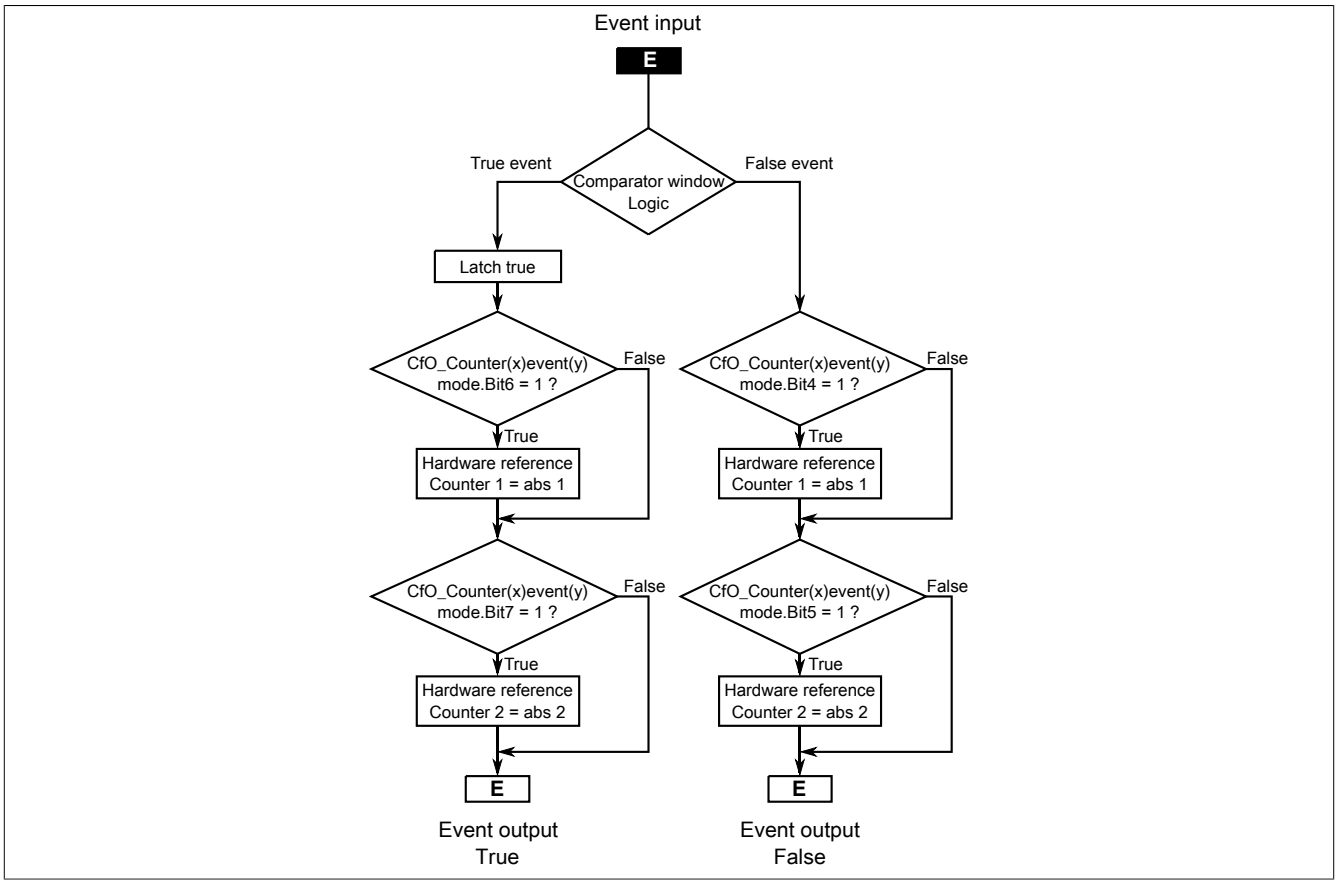
4.7.3 General event functions

Each of the 2 counter functions has 2 counter event functions. These consist of:

- Event ID that triggers the counter event function
- A window comparator
- Latch register for saving the counter value

When the counter event function is complete, a combined event ID in the range 2112 to 2401 (see "List of event IDs" on page 16) is sent.

Each counter event function also has the option to copy the current counter value to the "HW reference counter" when an event occurs (see "Counter value calculation" on page 22).



4.7.3.1 Configure counter mode

Name:
Counter function 1: CfO_Counter1config
Counter function 2: CfO_Counter2config

These registers are used to configure the mode of the counter function. Each counter function can be operated in 3 different modes.

	Counter function mode		
	Edge counters	**AB encoder**	**Up/down counter**
Counter channel 1[1)]	Counting pulses, edge counter 1	A	Metering pulses
Counter channel 2[1)]	Counting pulses, edge counter 2	B	Counting direction (0 = positive, 1 = negative)
Counter register 1	Counter value 1	Position	Counter value
Counter register 2	Counter value 2		

1) Corresponds to the physical channels of the counter functions. See "Description of channel assignments" on page 6.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Counter mode	00	Edge counters
		01	AB encoder
		11	Up/down counter
2 - 7	Reserved	-	

4.7.3.2 Configure calculation of internal counters

Name:
Counter function 1: CfO_Counter1configReg0 to CfO_Counter2configReg0
Counter function 2: CfO_Counter1configReg1 to CfO_Counter2configReg1

The calculation of the internal "counter1" and "counter2" registers can be configured in these registers. For information on using these internal registers, see "Counter value calculation" on page 22.

Data type	Values	Bus controller default setting[1)]
USINT	See the bit structure.	1

1) The bus controller default value applies only to the register numbers specified in function model 254.

Bit structure:

Bit	Description	Value	Information
0	counter 1 - use	0	0 is added instead of "counter 1"
		1	"counter 1" is used for addition
1	counter 1 - sign	0	The sign of the "counter 1" register is not changed for addition
		1	The sign of the "counter 1" register is reversed for addition
2	counter 2 - use	0	0 is added instead of "counter 2"
		1	"counter 2" is used for addition
3	counter 2 - sign	0	The sign of the "counter 2" register is not changed for addition
		1	The sign of the "counter 2" register is reversed for addition
4 - 7	Reserved	-	

Examples of calculation configurations

0b00000001	= 0x01	Only the "counter1 - use" bit is set, entering the contents of the "counter" (edge of counter event channel 1) directly in the counter register.
0b00000011	= 0x03	"counter 1 - use" and "counter1 - sign" bits are set. The sign is changed so that the counter register counts in the negative direction.
0b00001101	= 0x0d	Edges on counter input channel 1 increase the value in the counter register. Edges on counter input channel 2 decrease the value in the counter register. This value is the best setting for modes "AB counter" and "Up/Down counter".

4.7.3.3 Offset value for referencing

Name:

Counter function 1: CfO_Counter1PresetValue1 to CfO_Counter2PresetValue1
Counter function 1: CfO_Counter1PresetValue1_32Bit to CfO_Counter2PresetValue1_32Bit

Counter function 2: CfO_Counter1PresetValue2 to CfO_Counter1PresetValue2
Counter function 2: CfO_Counter1PresetValue2_32Bit to CfO_Counter1PresetValue2_32Bit

"Preset value" in the Automation Studio I/O configuration.

These registers can be used to define an offset value for referencing. This value is copied to the internal SW_reference_counter register of the respective counter register.

Data type	Value
INT	-32768 to 32767
DINT	-2,147,483,648 to 2,147,483,647

4.7.3.4 Counter register

Name:
Different names are used for these 4 registers depending on their function.

These 4 registers show the results of the counter value calculation for the respective register. Depending on the function, this corresponds to either the encoder position or the counter value.

For information on the relationship between physical channels and counter registers, see "Counters and encoders" on page 22 and "Description of channel assignments" on page 6

Counter function 1		
Counter register	**Function**	**Name**
1	**AB encoders**	ABEncoder01
	ABR encoders	ABREncoder01
	Up/down counters	Counter01
	Event counters	EventCounter01
2	**Event counters**	EventCounter02

Counter function 2		
Counter register	**Function**	**Name**
1	**AB encoders**	ABEncoder02
	Up/down counters	Counter02
	Event counters	EventCounter03
2	**Event counters**	EventCounter04

Data type	Value	Information
INT	-32,768 to 32,767	Encoder position or counter value
DINT[1)]	-2,147,483,648 to 2,147,483,647	Encoder position or counter value

1) Only in function model 1

4.7.3.5 Status of the ABR encoder

Name:
StatusABR01

This register contains the homing state of the ABR encoder.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Reserved	0	
2	Bit is always 1 after the first reference pulse.	0	No reference pulses have occurred since the start of referencing.
		1	The first reference pulse has occurred.
3	State change when referencing is complete	0 or 1	
4	Bit is always 1 after the first reference pulse.	0	No reference pulses have occurred since the start of referencing.
		1	The first reference pulse has occurred.
5 - 7	Continuous counter	xxx	Increased with each reference pulse

Examples of possible values

0b00000000	= 0x00	Referencing OFF or homing procedure already active
0b00111100	= 0x3C	First reference complete, reference value applied in the "ABREncoder0" on page 27 register
0bxxx11100	= 0xxB	Bits 5 to 7 are changed with each reference pulse
0bxxx1x100	= 0xxx	Bits changed continuously with the setting continuous referencing. With every reference pulse, the reference value is applied to the "ABREncoder0" on page 27 register

4.7.3.6 Configure ABR referencing mode

Name:
ReferenceModeABR01

The bits in this register are used to configure the reaction to the configured reference pulse.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Sets the referencing mode	00	Referencing OFF
		01	Single shot referencing
		10	Reserved
		11	Continuous referencing
2 - 5	Reserved	-	
6 - 7	Reserved	11	Must always be 11!

This results in the following values:

0b00000000	= 0x00	Referencing OFF
0b11000001	= 0xC1	Single shot referencing → When starting over after the referencing process is complete, the value 0x00 must be written to start again. Wait until the "StatusABR" on page 28 register also takes on the value 0x00, then the value 0xC1 can be written again.
0b11000011	= 0xC3	Continuous referencing → Referencing takes place automatically with every reference pulse

4.7.4 Comparator functions

The ABR and AB counters and the up/down counter have a comparator function. It always works the same and is described here globally for all three.

The comparators are implemented in software form. They do not work actively but rather passively, i.e. the comparison is only carried out when an event is received. The event received is forwarded along the TRUE or FALSE branch depending on the status of the comparator condition. An event function like this generally also offers a latch for the TRUE and FALSE branch to save the value used for the comparator at the time of the event.

4.7.4.1 Comparator modes

Comparator functions can be operated in 4 different modes.

- **Off**
  Events are ignored.
- **Individual**
  The event function is executed once and then disables itself automatically. To re-enable it, the "event function mode" must be changed, preferably to "off" and then to the desired mode. This setting allows a hardware latch to be simulated.
- **State change**
  The event function only responds when the comparator state changes, i.e. from False to True (or vice versa). Only the first event for each status is processed, e.g. the first "true" of a sequence of events with the comparator condition "true". After the event function is enabled, the first incoming event is used to determine the starting state and therefore not forwarded. This setting allows a hardware comparator to be simulated.
- **Continuous**
  Each incoming event is forwarded to the true or false branch depending on the comparator condition. This setting allows event filters to be created.

4.7.4.2 Configure event ID for comparator

Name:
Counter function 1: CfO_Counter1event0IDwr to CfO_Counter1event1IDwr
Counter function 2: CfO_Counter2event0IDwr to CfO_Counter2event1IDwr

This register holds the event ID that should trigger the counter event function. For a list of all possible event IDs, see "List of event IDs" on page 16

Data type	Value	Information
INT	192 to 7,233	ID of counter event function

4.7.4.3 Configure calculation of comparator

Name:
Counter function 1: CfO_Counter1event0config to CfO_Counter1event1config
Counter function 2: CfO_Counter2event0config to CfO_Counter2event1config

These registers are used to configure the counter event function for the respective counter function.

Bits 0 to 3 configure the calculation of the comparison or to latch the value. This calculation is similar to the calculation of the counter register (see "Counter value calculation" on page 22)

Bits 8 to 13 can be used to limit the number of bits used for the comparison. A mask is calculated as 2^n - 1 and linked with an "AND" operation. This makes it possible to generate a comparator pulse every 2^n increments.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	counter 1 - use	0	0 is added instead of "counter 1"
		1	"counter 1" is used for addition
1	counter 1 - sign	0	The sign of the "counter 1" register is not changed for addition
		1	The sign of the "counter 1" register is reversed for addition
2	counter 2 - use	0	0 is added instead of "counter 2"
		1	"counter 2" is used for addition
3	counter 1 - sign	0	The sign of the "counter 2" register is not changed for addition
		1	The sign of the "counter 2" register is reversed for addition
4 - 7	Reserved	-	
8 - 13	Number of bits for comparator mask	x	The mask value is calculated as 2^n-1, where n is value set in these bits. Default: 0
14	Reserved	-	
15	Margin comparator mode	0	MarginComparator01 >= (Current position - OriginComparator01)
		1	MarginComparator01 > (Current position - OriginComparator01)

4.7.4.4 Configure mode and latching of comparator function

Name:
Counter function 1: CfO_Counter1event0mode to CfO_Counter1event1mode
Counter function 2: CfO_Counter2event0mode to CfO_Counter2event1mode

It is possible to set the mode of the comparator function as well as possible copying of the latched registers in this register.

Comparator functions can be operated in 4 different modes. For a description, see "Comparator modes" on page 29.

Bits 4 to 7 can be used to define hardware referencing actions.

Based on these bits, the values of the internal absolute value counters "abs1" and "abs2" can be copied to the respective "HW_reference_counter" register at every counter event (see "Counter value calculation" on page 22). This function can be used to reference the counter values directly in the hardware.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Comparator mode	0	Off
		1	Individual
		2	State change
		3	Continuous
2 - 3	Reserved	-	
4	Copy abs1 counter value	0	No action
		1	When event is FALSE → hardware reference counter 1 = abs1
5	Copy abs2 counter value	0	No action
		1	When event is FALSE → hardware reference counter 2 = abs2
6	Copy abs1 counter value	0	No action
		1	When event is TRUE → hardware reference counter 1 = abs1
7	Copy abs2 counter value	0	No action
		1	When event is TRUE → hardware reference counter 2 = abs2

4.7.4.5 Comparator origin

Name:
OriginComparator01

This register is available for the comparator function of the ABR encoder, AB counter and up/down counter.

It defines the position value at which the respective configured comparator output channel is set.

Data type	Value	Information
INT	-32,768 to 32,767	Comparator window origin, 16-bit
DINT	-2,147,483,648 to 2,147,483,647	Comparator window origin, 32-bit

4.7.4.6 Width of the comparator

Name:
MarginComparator01

This register is available for the AB and ABR encoders and the up/down counters.

It defines the width of the comparator window in the positive direction.

Data type	Value	Information
INT	-32768 to 32767	Width of comparator window, 16-bit
DINT	-2,147,483,648 to 2,147,483,647	Width of comparator window, 32-bit

4.7.4.7 Read latch position or counter value

Name:
Different names are used for these 4 registers depending on their function.

If the comparator returns TRUE, then the current counter value is latched and copied to these registers. The calculation of the comparison value used for the latch can be configured in register "Configure calculation of comparator" on page 30.

Counter function 1		
Event function	**Function**	**Name**
1	**AB encoders**	Latch01AB01
	Up/down counters	Latch01Counter01
2	**ABR encoders**	Latch01ABR01
	AB encoders	Latch02AB01
	Up/down counters	Latch02Counter01

Counter function 2		
Event function	**Function**	**Name**
1	**AB encoders**	Latch01AB02
	Up/down counters	Latch01Counter02
	Event counters	Latch02AB02
2	**Event counters**	Latch02Counter02

Data type	Value	Information
INT	-32,768 to 32,767	Latched encoder position or counter value
DINT[1)]	-2,147,483,648 to 2,147,483,647	Latched encoder position or counter value

1) Only in function model 1

4.8 SSI encoder interface

The module has 1 SSI encoders available, supported directly in the hardware. Two 24 V output channels are set for the SSI encoder and cannot be changed. (See also "Description of channel assignments" on page 6)

When using the SSI encoder, the corresponding clock channel can be configured in the "CfO_CFGchannel" on page 13 register as "Channel-specific" and "Push/Pull".

SSI encoders	Channel number
Data channel	1
Clock channel	2

4.8.1 SSI event functions

The SSI counter consists of an event function and an event input. The SSI cycle is started when an event is received on this input.

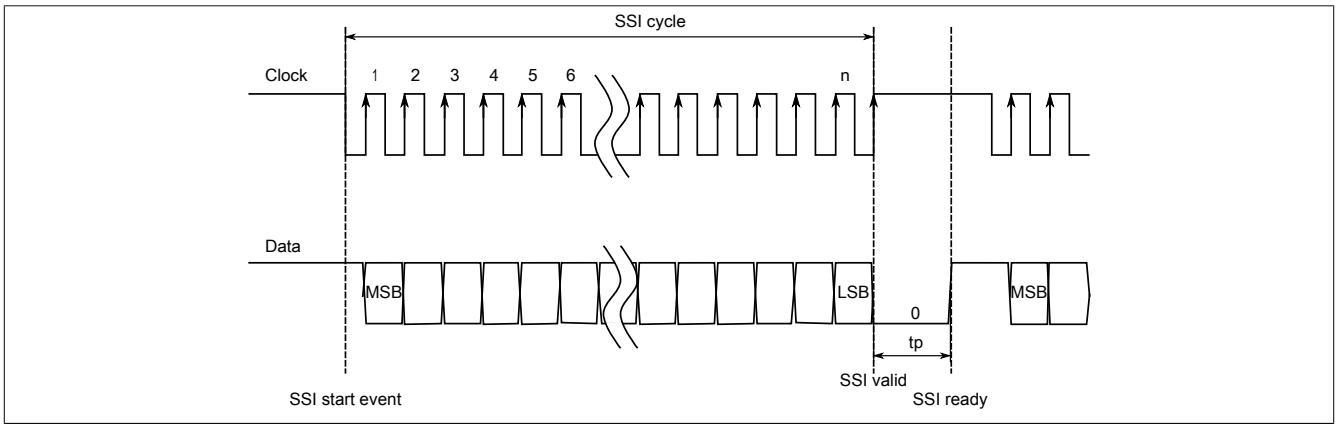
Information:

The SSI event function is not linked to an event by default, i.e. SSI functions are disabled.

2 events are transmitted from the SSI encoder interface.

- An "SSI valid" event is triggered immediately after the end of the SSI cycle if a new counter value is available.
- The "SSI ready" event then shows when the monoflop time has expired (tp in SSI encoder timing diagram). This is the earliest that the next SSI cycle can be started.

SSI encoder - Timing diagram



4.8.1.1 Configure event ID for SSI

Name:
CfO_SSI1eventIDwr

This register holds the event ID that should start the SSI cycle. For a list of all possible event IDs, see "List of event IDs" on page 16

Normally this register is set to network event 225 "AOSISOP"- This ensures that the new encoder position is available at the next "I/O → Synchronous Frame" transfer. Check the SSI transfer time and the X2X cycle time, because the SSI cycle must be completed within this time.

Data type	Value	Information
INT	192 to 7,233	ID of event function

4.8.1.2 Configure SSI

Name:
CfO_SSI1cfg

This configuration register sets the encoding, clock rate and number of bits. Default = 0. This must be set once using an acyclic write command.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Name	Value	Information
0 - 5	SSI value valid bits	x	
6 - 7	Clock rate	00	1 MHz
		01	500 kHz
		10	250 kHz
		11	125 kHz
8 - 13	SSI number of bits	x	Number of bits including leading zeros
14	Reserved	0	
15	Keying	0	Binary coding
		1	Gray coding

4.8.1.3 SSI advanced configuration

Name:
ConfigAdvanced

This configuration register is used to set the encoding, clock rate, bit count and monostable multivibrator check settings. This must be set once using an acyclic write command.

It only differs from "CfO_SSI1cfg" on page 33 by data length and additional monostable multivibrator testing.

Data type	Values
UDINT	See the bit structure.

Bit structure:

Bit	Name	Value	Information
0 - 5	SSI value valid bits	x	
6 - 7	Clock rate	00	1 MHz
		01	500 kHz
		10	250 kHz
		11	125 kHz
8 - 13	SSI number of bits	x	Number of bits including leading zeros
14	Reserved	0	
15	Keying	0	Binary coding
		1	Gray coding
16 - 17	Monostable multivibrator check	00	Check OFF, no additional clock bit
		01	Check set to High level
		10	Check set to Low level
		11	Level is clocked but ignored
18 - 31	Reserved	0	

4.8.1.4 Enable SSI event function

Name:
CfO_SSI1control

The two SSI encoder events can be enabled/disabled using this register.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Event: "SSI valid"	0	Not sent
		1	Sent
1	Event: "SSI ready"	0	Not sent
		1	Sent
2 - 7	Reserved	-	

4.8.1.5 Read SSI position

Name:
SSIEncoder01

The last transferred SSI position can be read out from this register. The SSI encoder value is displayed as a 32-bit position value. This position value is generated synchronously with the X2X cycle.

Data type	Value	Information
UDINT	0 to 4,294,967,295	Last SSI position transferred

4.8.2 SSI comparator condition

The module has an assigned comparator function for the SSI function. These consist of:

- Event ID that triggers the comparator function
- The window comparator
- Latch register for saving the counter value

When the comparator function is complete, event ID 7232 or 7233 (see "List of event IDs" on page 16) is sent.

4.8.2.1 Configure event ID for SSI comparator

Name:
CfO_SSI1event0IDwr

This register holds the event ID that should start the SSI comparator function. For a list of all possible event IDs, see "List of event IDs" on page 16

Data type	Value	Information
INT	192 to 7,233	ID of comparator function

4.8.2.2 Configure the mode of the SSI comparator function

Name:
CfO_SSI1event0mode

This register can be used to configure the mode of the comparator function.

Comparator functions can be operated in 4 different modes. For a description, see "Comparator modes" on page 29.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 1	Comparator mode	0	Off
		1	Individual
		2	State change
		3	Continuous
2 - 7	Reserved	-	

4.8.2.3 Configure calculation of SSI comparator

Name:
CfO_SSI1event0config

The calculation of the position value used for the comparator can be configured in this register.

The window comparator condition is calculated as follows:

```
counter_window_value = ssi_counter & (2^ssi_data_bits - 1)
diff = counter_window_value – origin_comparator
if ((diff & (2^(comparator_mask)-1)) <= margin_comparator)
condition = True;
else
condition = False;
```

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 5	SSI data bits	x	Number of data bits used for masking
6 - 7	Reserved	-	
8 - 13	Comparator mask	x	The mask value is calculated from 2^n-1, where n is the value configured in SSI data bits. Default: 0
14	Comparator mode	0	MarginComparator >= SSI position - OriginComparator
		1	MarginComparator > SSI position - OriginComparator

4.8.2.4 Origin of the SSI comparator

Name:
OriginComparator01_SSI

This register contains the origin of the window comparator.

Data type	Value	Information
UDINT	0 to 4,294,967,295	Origin of the window comparator.

4.8.2.5 Width of the SSI comparator

Name:
MarginComparator01_SSI

This register provides the width of the window comparator.

Data type	Value	Information
UDINT	0 to 4,294,967,295	Width of the SSI window comparator

4.8.2.6 Read SSI latch position

Name:
Latch01SSI01

If the SSI window comparator returns "True", then the current SSI position is latched and saved in this register.

Data type	Value	Information
UDINT	0 to 4,294,967,295	Latched SSI position

4.9 PWM - Pulse width modulation

The module has 2 PWM functions available, supported directly by the hardware. A 24 V output channel is set for each PWM encoder and cannot be changed. (See also "Description of channel assignments" on page 6)

When using the PWM function, the corresponding channel can be configured in the "CfO_CFGchannel" on page 13 register as "Channel-specific".

PWM function	Channel
PWM1	2
PWM2	4

4.9.1 Configure PWM prescaler

Name:
CfO_PWM0prescaler to CfO_PWM1prescaler

The length of the PWM cycle is configured using this register. The base is a 48 MHz clock, which can be changed (divided) using the setting in this register. One PWM cycle consists of 1000 of the resulting clocks after they have been divided. The period duration of the PWM cycle is calculated as follows:

$$PWM\_cycle = 1000 \frac{prescale}{48000000} [s]$$

Data type	Value	Information
UINT	2 to 65535	Prescaler for PWM cycle

4.9.2 Output PWM values

Name:
PWMOutput02 and PWMOutput04

In this register, a configuration is made for the percentage of the PWM cycle (in 1/10% steps) that the PWM output is logical 1, i.e. ON.

Data type	Value	Information
UINT	0	PWM output always off
	1 to 999	Turn on time in 1/10% steps
	1000	PWM output always on

4.10 Time measurement function

The module has a time measurement function for each I/O channel. It can be configured separately for rising and falling edges on each channel.
A starting edge can be configured for each time measurement function. When a configured starting edge occurs, the value of the internal timer is saved in a FIFO buffer. This FIFO buffer holds up to 16 elements. When the actual trigger edge occurs, the difference in time between the starting edge and the triggered edge is copied to the respective register.
Bits 8 to 11 "Previous start edge" of registers "CfO_EdgeTimeFallingMode" on page 37 and "CfO_EdgeTimeRisingMode" on page 38 can be used to define which detected starting edge from the FIFO buffer should be used to calculate the difference. In addition, when the trigger edge occurs, the current counter value of the counter internally clocked by bits 12 to 15 "Resolution of time measurement" is copied to registers "TimeStampFallingCH" on page 40 and "TimeStampRisingCH" on page 40.

Information:

The time measurement function is an extension of edge detection, so all of the channels used must be configured there.

4.10.1 Enable time measurement function

Name:
CfO_EdgeTimeglobalenable

This register enables/disables the time measurement function for the entire module.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	Time measurement function	0	Disabled for entire module
		1	Enabled for entire module
1 - 7	Reserved	-	

4.10.2 Configure time measurement function for the falling edge

Name:
CfO_EdgeTimeFallingMode01 to CfO_EdgeTimeFallingMode04

These registers can be used to configure the time measurement function for the falling edge of the respective channel.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 3	Selects the channel for the starting edge	0	Channel 1
		...	
		3	Channel 4
4	Selects the edge for the starting edge	0	The falling edge of the channel configured in bits 0 to 3 serves as the starting edge.
		1	The rising edge of the channel configured in bits 0 to 3 serves as the starting edge.
5 - 6	Reserved	-	
7	Trigger	0	Triggered[1)]
		1	Continuous[2)]
8 - 11	Previous start edge	0 to 15	The value determines which entry in the starting edge FIFO should be used to calculate the time difference.
12 - 15	Time measurement resolution	0	8 Mhz
		1	4 Mhz
		2	2 Mhz
		3	1 Mhz
		4	500 kHz
		5	250 kHz
		6	125 kHz
		7	625 kHz

1) The time measurement is triggered by the corresponding bit in the "TriggerRisingCH" on page 39 register.
2) Time measurement runs continuously and is triggered at every edge.

4.10.3 Configure time measurement function for the rising edge

Name:
CfO_EdgeTimeRisingMode01 to CfO_EdgeTimeRisingMode04

These registers can be used to configure the time measurement function for the rising edge of the respective channel.

Data type	Values
UINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0 - 3	Selects the channel for the starting edge	0	Channel 1
		...	
		3	Channel 4
4	Selects the edge for the starting edge	0	The falling edge of the channel configured in bits 0 to 3 serves as the starting edge.
		1	The rising edge of the channel configured in bits 0 to 3 serves as the starting edge.
5 - 6	Reserved	-	
7	Trigger	0	Triggered[1)]
		1	Continuous[2)]
8 - 11	Previous start edge	0 to 15	The value determines which entry in the starting edge FIFO should be used to calculate the time difference.
12 - 15	Time measurement resolution	0	8 Mhz
		1	4 Mhz
		2	2 Mhz
		3	1 Mhz
		4	500 kHz
		5	250 kHz
		6	125 kHz
		7	625 kHz

1) The time measurement is triggered by the corresponding bit in the "TriggerRisingCH" on page 38 register.
2) Time measurement runs continuously and is triggered at every edge.

4.10.4 Trigger falling edge detection

Name:
TriggerFallingCH01 to TriggerFallingCH04

If bit 7 "Trigger" is cleared in register "CfO_EdgeTimeFallingMode" on page 37, then detection of a falling edge on the respective input can be triggered using the respective bit in this register. After a bit has been set, the next falling edge on the corresponding channel is detected.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	TriggerFallingCH01	0	Falling edges on channel 1 are not detected
		1	The next falling edge on channel 1 will be detected
...		...	
3	TriggerFallingCH04	0	Falling edges on channel 4 are not detected
		1	The next falling edge on channel 4 will be detected
4 - 7	Reserved	-	

4.10.5 Trigger rising edge detection

Name:
TriggerRisingCH01 to TriggerRisingCH04

If bit 7 "Trigger" is cleared in register "CfO_EdgeTimeRisingMode" on page 38, then detection of a rising edge on the respective input can be triggered using the respective bit in this register. After a bit has been set, the next rising edge on the corresponding channel is detected.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	TriggerRisingCH01	0	Rising edges on channel 1 are not detected
		1	The next rising edge on channel 1 will be detected
...		...	
3	TriggerRisingCH04	0	Rising edges on channel 4 are not detected
		1	The next rising edge on channel 4 will be detected
4 - 7	Reserved	-	

4.10.6 Show first falling trigger edge

Name:
BusyTriggerFallingCH01 to BusyTriggerFallingCH04

If edges are triggered via the bits in the "TriggerFallingCH" on page 38 register, then a set bit in this register indicates that no falling edges have been detected on the respective channel since the corresponding bit was set in the "TriggerFallingCH" register. If a falling edge occurs on the respective channel, then the corresponding BusyTriggerFalling bit is cleared.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	BusyTriggerFallingCH01	0	Falling edge detected on channel 1
		1	Module waiting for a falling edge on channel 1
...		...	
3	BusyTriggerFallingCH04	0	Falling edge detected on channel 4
		1	Module waiting for a falling edge on channel 4
4 - 7	Reserved	-	

4.10.7 Show first rising trigger edge

Name:
BusyTriggerRisingCH01 to BusyTriggerRisingCH04

If edges are triggered via the bits in the "TriggerRisingCH" on page 39 register, then a set bit in this register indicates that no rising edges have been detected on the respective channel since the corresponding bit was set in the "TriggerRisingCH" register. If a rising edge occurs on the respective channel, then the corresponding BusyTriggerRising bit is cleared.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Description	Value	Information
0	BusyTriggerRisingCH01	0	Rising edge detected on channel 1
		1	Module waiting for a rising edge on channel 1
...		...	
3	BusyTriggerRisingCH04	0	Rising edge detected on channel 4
		1	Module waiting for a rising edge on channel 4
4 - 7	Reserved	-	

4.10.8 Count falling trigger edges

Name:
CountFallingCH01 to CountFallingCH04

These registers contain cyclic counters that are incremented with every detected falling edge on the respective channel.

Data type	Value	Information
USINT	0 to 255	Counter for falling edges

4.10.9 Count rising trigger edges

Name:
CountRisingCH01 to CountRisingCH04

These registers contain cyclic counters that are incremented with every detected rising edge on the respective channel.

Data type	Value	Information
USINT	0 to 255	Counter for rising edges

4.10.10 Timestamp of falling edge

Name:
TimeStampFallingCH01 to TimeStampFallingCH04

When a falling edge occurs on the respective channel, the current counter value of the module timer is copied to these registers.

Data type	Value	Information
UINT	0 to 65535	Timestamp for rising edges

4.10.11 Timestamp of the rising edge

Name:
TimeStampRisingCH01 to TimeStampRisingCH04

When a rising edge occurs on the respective channel, the current counter value of the module timer is copied to these registers.

Data type	Value	Information
UINT	0 to 65535	Timestamp for rising edges

4.10.12 Time difference of falling edge

Name:
TimeDiffFallingCH01 to TimeDiffFallingCH04

When a falling edge occurs on the respective channel, the time difference compared to the starting edge configured in bit 4 of the "CfO_EdgeTimeFallingMode" on page 37 register is copied to this register.

Data type	Value	Information
UINT	0 to 65,535	Time difference from starting edge

4.10.13 Time difference of rising edge

Name:
TimeDiffRisingCH01 to TimeDiffRisingCH04

When a rising edge occurs on the respective channel, the time difference compared to the starting edge configured in bit 4 of the "CfO_EdgeTimeRisingMode" on page 38 register is copied to this register.

Data type	Value	Information
UINT	0 to 65,535	Time difference from starting edge

4.11 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
128 µs

4.12 Maximum cycle time

The maximum cycle time specifies the time up to which the bus cycle can be increased without internal counter overflows causing module malfunctions.

Maximum cycle time
16 ms

4.13 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
128 µs