X20ATB312

1 General information

The module is equipped with 4 inputs for PT100 4-line resistance temperature measurement.

- 4 inputs for resistance temperature measurement
- PT100 sensor
- Direct resistance measurement as well
- 4-wire measurement
- Filter time can be configured
- NetTime timestamp: Moment of measurement

NetTime timestamp of the measurement

For many applications, not only the measured value is important, but also the exact time of the measurement. The module is equipped with a NetTime timestamp function for this that supplies a timestamp for the recorded position and trigger time with microsecond accuracy.

The timestamp function is based on synchronized timers. If a timestamp event occurs, the module immediately saves the current NetTime. After the respective data is transferred to the CPU, including this precise time, the CPU can then evaluate the data using its own NetTime (or system time), if necessary.

2 Order data

Model number	Short description	Figure
	Temperature measurement	
X20ATB312	X20 temperature input module, 4 inputs for resistance measurement, PT100, resolution 0.01°C, 4-wire connections, NetTime function	
	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	
X20TB1F	X20 terminal block, 16-pin, 24 VDC keyed	

Table 1: X20ATB312 - Order data

3 Technical data

Model number	X20ATB312
Short description	
I/O module	4 inputs for PT100 resistance temperature measurement
General information	
B&R ID code	0xE0EF
Status indicators	I/O function per channel, operating state, module status
Diagnostics	
Module run/error	Yes, using LED status indicator and software
Inputs	Yes, using LED status indicator and software
Power consumption	
Bus	0.01 W
Internal I/O	0.5 W (Rev. ≥ D0), 0.6 W (Rev. < D0)
Additional power dissipation caused by actuators (resistive) [W]	-
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
EAC	Yes
Resistance measurement temperature inputs	
Input	Resistance measurement with constant current supply for 4-wire connections
Digital converter resolution	24-bit
Filter time	Configurable between 1 and 200 ms
Conversion time [1]	
1 channel	20 ms with 50 Hz filter
2 channels	40 ms per channel with 50 Hz filter
Conversion procedure	Sigma-delta
Output format	DINT or UDINT for resistance measurement
Temperature measurement range	-200 to 850°C
Resistance measurement range	0.5 to 390 Ω
Temperature sensor resolution	1 LSB = 0.01°C
Resistance measurement resolution	0.001 Ω
Input filter	First-order low-pass filter / cutoff frequency 1050 Hz
Sensor standard	EN 60751
Isolation voltage between channel and bus	500 V_{eff}
Linearization method	Internal
Measurement current	1 mA
Temperature sensor normalization	-200.0 to 850.0°C
Reference	1568 Ω ±0.1%
Permissible input signal	Short-term max. 28.8 V
Max. error at 25°C [2]	
Gain	0.0059% [3]
Offset	0.0015% [4]
Max. gain drift	<0.00065 %/°C [3]
Max. offset drift	<0.000025 %/°C [4]
Nonlinearity	<0.001% [4]
Standardized range of values for resistance measurement	19 to 390 Ω
Temperature measurement monitoring	
Range undershoot	0x80000001
Range overshoot	0x7FFFFFFF
Open circuit	0x7FFFFFFF
General fault	0x80000000
Open inputs	0x7FFFFFFF
Resistance measurement monitoring	
Range undershoot	0x80000001
Range overshoot	0xFFFFFFFF
Open circuit	0xFFFFFFFF
General fault	0x80000000
Electrical properties	
Electrical isolation	Channel isolated from bus Channel not isolated from channel

Table 2: X20ATB312 - Technical data

Model number	X20ATB312
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
Storage	5 to 95%, non-condensing
Transport	5 to 95%, non-condensing
Mechanical properties	
Note	Order 1x terminal block X20TB1F separately. Order 1x bus module X20BM11 separately.
Spacing	12.5 $^{+0.2}$ mm

Table 2: X20ATB312 - Technical data

1) The module is equipped with two independent converters (sensors 1 and 2, sensors 3 and 4). The conversion time applies to the number of channels connected to the respective converter.
2) To guarantee accuracy, modules with a power dissipation < 1.2 W must be connected to the left and right of this module.
3) Based on the current resistance value.
4) Based on the entire resistance measurement range.

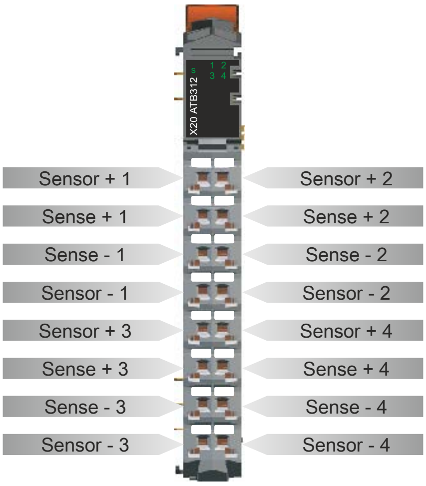
4 Status LEDs

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

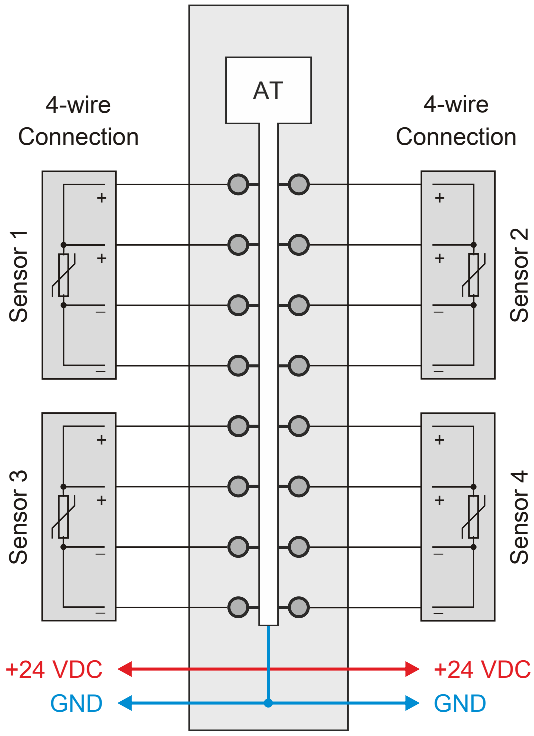
Image	LED	Color	Status	Description
	s	Green	Off	Module supply not connected
			Single flash	Reset mode
			Double flash	Mode BOOT (during firmware update)[1]
			Blinking	PREOPERATIONAL mode
			On	RUN mode
		Red	Off	Module supply not connected or everything OK
			On	Error or reset status
			Single flash	Parameter or conversion error[2]
		Red on / Green single flash		Invalid firmware
	1 - 4	Green	Off	Input turned off or not supplied
			Single flash	Parameter error[2]
			Double flash	Conversion error[2]
			Blinking	Overflow, underflow or open line
			On	A/D converter running, value OK

1) Depending on the configuration, a firmware update can take up to several minutes.
2) Parameter or converter errors are indicated simultaneously on the red "s" LED and the channel LED of the respective output.

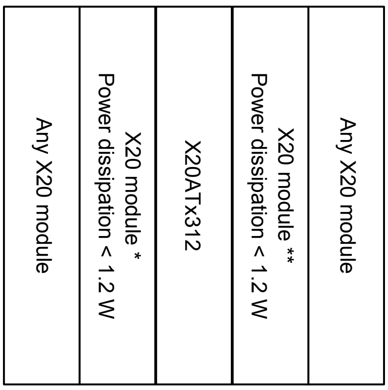
5 Pinout



6 Connection example

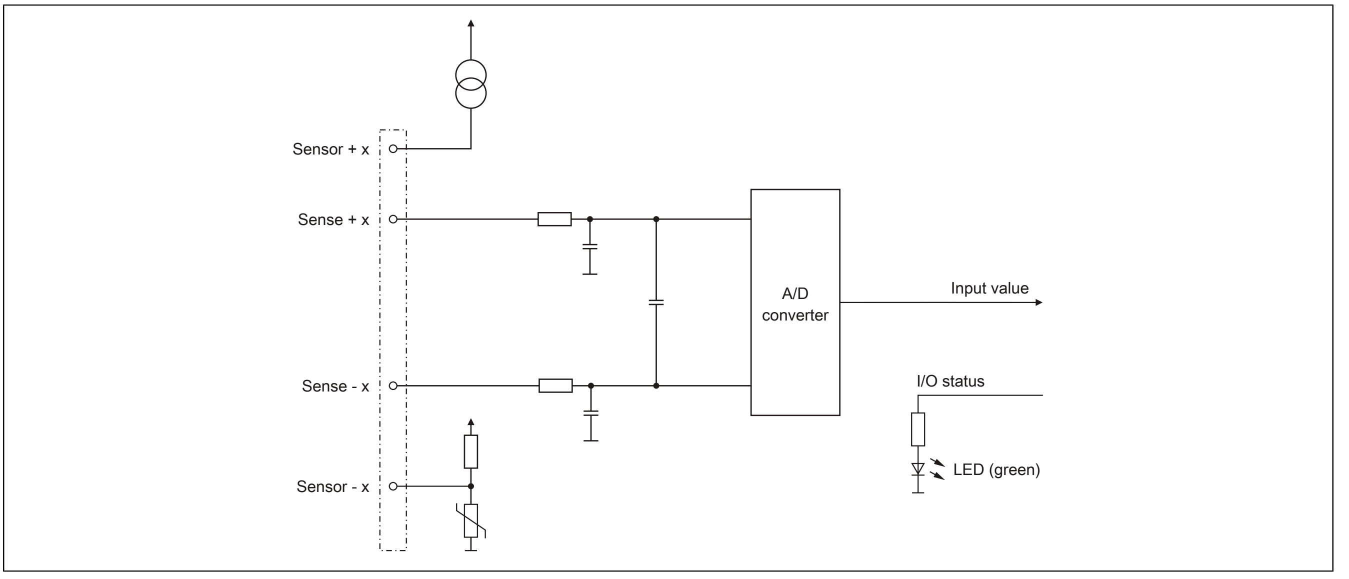


To ensure accuracy, modules with a power consumption <1.2 W must be inserted to the left and right of this modules.



* E.g. X20PS2100
** E.g. X20DO6639

7 Input circuit diagram



8 Register description

8.1 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.

8.2 Function model 0 - Standard

Register	Name	Data type	Read		Write	
			Cyclic	Acyclic	Cyclic	Acyclic
Configuration						
130	InputFilter	UINT				•
134	ModeADC	UINT				•
Index * 64 + 450	SensorType0x (Index x = 1 to 4)	UINT				•
Index * 64 + 502	PreparationInterval0x (Index x = 1 to 4)	UINT				•
Index * 64 + 484	ReplaceUpper0x (Index x = 1 to 4)	DINT				•
Index * 64 + 476	ReplaceLower0x (Index x = 1 to 4)	DINT				•
Index * 64 + 468	UpperLimit0x (Index x = 1 to 4)	DINT				•
Index * 64 + 460	LowerLimit0x (Index x = 1 to 4)	DINT				•
Index * 64 + 490	Hysteresis0x (Index x = 1 to 4)	UINT				•
Index * 64 + 494	ErrorDelay0x (Index x = 1 to 4)	UINT				•
Index * 64 + 498	SumErrorDelay0x (Index x = 1 to 4)	UINT				•
Communication						
Index * 4 - 4	Temperature0x (Index x = 1 to 4)	DINT	•			
	Resistor0x (Index x = 1 to 4)	UDINT				
Index * 64 + 196	Measurand0x (Index x = 1 to 4)	DINT		•		
Index * 64 + 217	IOCycleCounter0x (Index x = 1 to 4)	USINT	•			
Index * 64 + 218	IOCycleCounter0x (Index x = 1 to 4)	UINT	•			
Index * 64 + 210	Sampletime0x (Index x = 1 to 4)	INT	•			
Index * 64 + 212	Sampletime0x (Index x = 1 to 4)	DINT	•			
Index * 64 + 233	Status0x (Index x = 1 to 4)	USINT	•			
	Underrun0x	Bit 0				
	Overrun0x	Bit 1				
	OpenLine0x	Bit 2				
	ConverterFault0x	Bit 4				
	SumFault0x	Bit 5				
	ParameterFault0x	Bit 6				
	IoSupplyFault0x	Bit 7				

8.3 Function model 254 - Bus Controller

Register	Offset[1]	Name	Data type	Read		Write	
				Cyclic	Acyclic	Cyclic	Acyclic
Configuration							
130	-	InputFilter	UINT				•
134	-	ModeADC	UINT				•
Index * 64 + 450	-	SensorType0x (Index x = 1 to 4)	UINT				•
Index * 64 + 502	-	PreparationInterval0x (Index x = 1 to 4)	UINT				•
Index * 64 + 484	-	ReplaceUpper0x (Index x = 1 to 4)	DINT				•
Index * 64 + 476	-	ReplaceLower0x (Index x = 1 to 4)	DINT				•
Index * 64 + 468	-	UpperLimit0x (Index x = 1 to 4)	DINT				•
Index * 64 + 460	-	LowerLimit0x (Index x = 1 to 4)	DINT				•
Index * 64 + 490	-	Hysteresis0x (Index x = 1 to 4)	UINT				•
Index * 64 + 494	-	ErrorDelay0x (Index x = 1 to 4)	UINT				•
Index * 64 + 498	-	SumErrorDelay0x (Index x = 1 to 4)	UINT				•
Communication							
Index * 4 - 4	Index * 4 - 4	Temperature0x (Index x = 1 to 4)	DINT	•			
		Resistor0x (Index x = 1 to 4)	UDINT				
Index * 64 + 217	-	IOCycleCounter0x (Index x = 1 to 4)	USINT		•		
30	-	Status01To04	USINT		•		

1) The offset specifies where the register is within the CAN object.

8.3.1 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).

8.3.2 CAN I/O bus controller

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

- Upgrade version <1.1.3.0: 1
- Upgrade version ≥1.1.3.0: 2

8.4 Configuration of the A/D converter

8.4.1 Setting the conversion rate

Name:
InputFilter

This register helps configure the sampling time of the A/D converter.

Data type	Value	Filter time in ms	Conversion rate in s^{-1}
UINT	4	1	1000
	9	2	500
	48	10	100
	80	16.33	60
	96	20 (bus controller default setting)	50
	160	33.33	30
	192	40	25
	320	66.7	15
	480	100	10
	960	200	5

Information:

The lower the conversion interval is set, the more precisely the value can be converted. However, this also increases the I/O update time.

8.4.2 Operating mode of the A/D converter

Name:
ModeADC

The operating mode for the A/D converter can be configured in this register.

The individual options allow faster digitalization of the analog values, but this also reduces the precision of the measured values.

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

Bit structure:

Bit	Description	Value	Information
0	Chopper mode	0	Alternating gain of the analog value (bus controller default setting)
		1	Chopper mode off
1	Order of the SINC filter	0	SINC4 (bus controller default setting)
		1	SINC3
2 - 15	Reserved	-	-

The following applies:

ConversionTime(SINC3) = ConversionTime(SINC4) - 1 x ConversionCycle
ConversionTime(without Chop) = 0.5 x ConversionTime(Chop)

8.5 Configuring the measurement channels

Each temperature measurement channel can be configured independently. All the registers required for this purpose by each channel are arranged separately.

8.5.1 Channel parameters

Name:
SensorType01 to SensorType04

This register defines the basic behavior of the channel.

Data type	Values	Bus controller default setting
UINT	See the bit structure.	129

Bit structure:

Bit	Description	Value	Information
0 - 2	Sensor type with unit and resolution	001	PT100 [10 mK/bit] - Temperature measurement (bus controller default setting)
		010	PT100 [1 mΩ/bit] - Resistance measurement
		011 to 111	Reserved
3 - 4	Reserved	-	
5	Replacement value strategy	0	Replace statically
		1	Retain last valid value
6	Monitoring the user-defined limit values	0	Switch off additional limits
		1	Switch on additional limits
7	Channel (on/off)	0	Switch off the entire channel
		1	Switch on channel (bus controller default setting)
8 - 15	Reserved	-	

8.6 Configuring the replacement value strategy

If a measured value is detected that is outside the permitted value range, the behavior of the input register must still remain clearly defined. The module provides the user two different options for this purpose.

Retain last valid value

With this strategy, the determined measured value is stored temporarily for a specific time and written to the input register after a delay. If an invalid measured value is detected, this value and all values that have been stored temporarily are discarded. The last valid input register value is retained. To update the value in the input register, there must be enough valid values stored in the temporary buffer. The number needed is determined by the time period specified in "PreparationInterval0x".

Replace with static value

With this strategy, the measured value is written to the input register without delay. If an invalid value occurs, it is replaced by a static value that has been predefined by the user.

8.6.1 Preparation interval

Name:
PreparationInterval01 to PreparationInterval04

This register defines the time interval in which the measured value is checked before being passed on.

Data type	Value	Information
UINT	0 to 65,535	Unit in 0.1 ms. Bus controller default setting: 0

Information:

This register must be defined if the replacement value strategy "Retain last valid value" was selected in register "SensorType0x" on page 9.

8.6.2 Static replacement value when exceeding the upper limit

Name:
ReplaceUpper01 to ReplaceUpper04

This register is used to defined a replacement value that is output in place of the invalid measured value if the upper limit is violated.

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

Information:

This register must be defined if the replacement value strategy "Replace with static value" was selected in register "SensorType0x" on page 9.

8.6.3 Static replacement value when falling below the lower limit

Name:
ReplaceLower01 to ReplaceLower04

This register is used to defined a replacement value that is output in place of the invalid measured value if the lower limit is violated.

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

Information:

This register must be defined if the replacement value strategy "Replace with static value" was selected in register "SensorType0x" on page 9.

8.7 Configuring the user-defined limit values

This module provides the user the option to specify user-defined limits. If the valid measurement range is reduced in this way, the behavior of the replacement value strategy is more likely to be applied.

Valid measurement range

The valid range is derived from the properties of the sensor being used or the hardware and firmware of the respective B&R module. These values cannot be changed by the user.

Valid range of values

The range of values is always within the valid measurement range. The range of values can be adapted to the requirements of the application by specifying the upper and lower limit value.

8.7.1 Upper limit value

Name:
UpperLimit01 to UpperLimit04

This register specifies the upper limit value. The values entered should be within the valid measurement range.

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

8.7.2 Lower limit value

Name:
LowerLimit01 to LowerLimit04

This register specifies the lower limit value. The values entered should be within the valid measurement range.

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

8.7.3 Hysteresis

Name
Hysteresis01 to Hysteresis04

A hysteresis can be set in order to avoid frequent status changes in the measurement range close to the limit value. Here, a small section is defined at the edge of the valid range of values where the measured values retain the status (valid or invalid) of the previous measured value.

Data type	Values	Information
UINT	0 to 65,535	Bus controller default setting: 16

8.8 Configuring status messages

Errors are detected by the module and sent to the application. When using Function model 0 - Standard, the trigger behavior of these error messages can be influenced by the "Delay" register.

In Automation Studio, an error message can be read either packed as the entire register or individually as bits.

8.8.1 Delaying error messages

Name:
ErrorDelay01 to ErrorDelay04

In order to avoid false alarms due to short-term measurement variations, the status messages transmitted to the PLC can be delayed. This register determines the number of A/D conversions in which an error must exist before an error message is transmitted.

Data type	Values	Information
UINT	0 to 65,535	AD conversions. Bus controller default setting: 2

8.8.2 Delaying the sum error message

Name:
SumErrorDelay01 to SumErrorDelay04

This register can be used to set the delay used when sending bit 5 of the "Status0x" on page 13 register to the PLC independent of the other status messages.

Data type	Values	Information
UINT	0 to 65,535	Bus controller default setting: 4000

8.9 Communication

The received temperature data is stored with a timestamp and, depending on the configuration, is made available under various register names and data types.

8.9.1 Measured value – Temperature

Name:
Temperature01 to Temperature04

If the channel is configured for resistance measurement, the current temperature value is made available in this register.

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

8.9.2 Measured value – Resistance

Name:
Resistor01 to Resistor04

If the channel is configured for resistance measurement, the current resistance value is made available in this register.

Data type	Values
UDINT	0 to 4,294,967,295

8.9.3 Measured value – Unweighted

Name:
Measurand01 to Measurand04

When using the AsIoAcc library, the unweighted measurement can be accessed via this register. This refers to a measured value that is within the valid measurement range and has not yet been compared with the user-defined limits.

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

Information:

If no user-defined limits are configured, the value of this register does not differ from the temperature or resistance value.

8.9.4 Cycle counter

Name:
IOCycleCounter01 to IOCycleCounter04

This register is used to provide a continuous counter for the application that is incremented each time a temperature value is read.

Data type	Value	Information
USINT	0 to 32,767	AD conversion.
UINT	0 to 65,535	AD conversion.

8.9.5 Sampling time

Name:
Sampletime01 to Sampletime04

This register provides the application with the NetTime at the time of temperature recording.

For more information about NetTime and timestamps, see "NetTime technology" on page 14.

Data type	Value	Information
INT	-32,768 to 32,767	NetTime timestamp in µs
DINT	-2,147,483,648 to 2,147,483,647	NetTime timestamp in µs

Information:

The SDC library requires a 16-bit value for the sampling time. It is therefore also prepared as a 16-bit value.

8.9.6 Status messages

Name:
Status01 to Status04

The register bits are set if an error has been diagnosed and the error remains longer than the delay configured in the "ErrorDelay0x" on page 11 register.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Name	Value	Information
0	Underrun01 to Underrun04	0	No error
		1	Value below the permitted range
1	Overrun01 to Overrun04	0	No error
		1	Value above the permitted range
2	OpenLine01 to OpenLine04	0	No error
		1	Sensor is not connected correctly
3	Reserved	-	
4	ConverterFault01 to ConverterFault04	0	No error
		1	Invalid A/D converter output
5	SumFault01 to SumFault04	0	No error
		1	Composite error
6	ParameterFault01 to ParameterFault04	0	No error
		1	The "SensorType0x" on page 9 register is faulty
7	IoSupplyFault01 to IoSupplyFault04	0	No error
		1	The supply voltage (I/O) is faulty

8.9.7 Status messages for function model 254

Name:
Status01To04

The bits in this register are set if an error has been detected.

Data type	Values
USINT	See the bit structure.

Bit structure:

Bit	Name	Value	Information
0	Underrun on channel 01	0	No error
		1	Value below the permitted range
1	Overrun on channel 01	0	No error
		1	Value above the permitted range
...		...	
6	Underrun on channel 04	0	No error
		1	Value below the permitted range
7	Overrun on channel 04	0	No error
		1	Value above the permitted range

Information:

If an open line is detected on a channel, then both error messages will be displayed at the same time.

8.10 NetTime technology

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

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

NetTime TECHNOLOGY

8.10.1 Time information

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

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

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

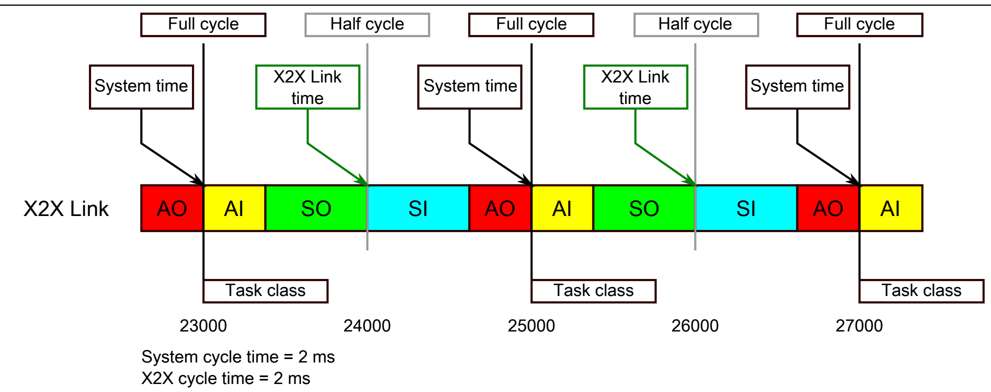
The initialization of the times is based on the system time during the startup of the X2X Link, the I/O modules or the POWERLINK interface.

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

8.10.1.1 PLC/Controller data points

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

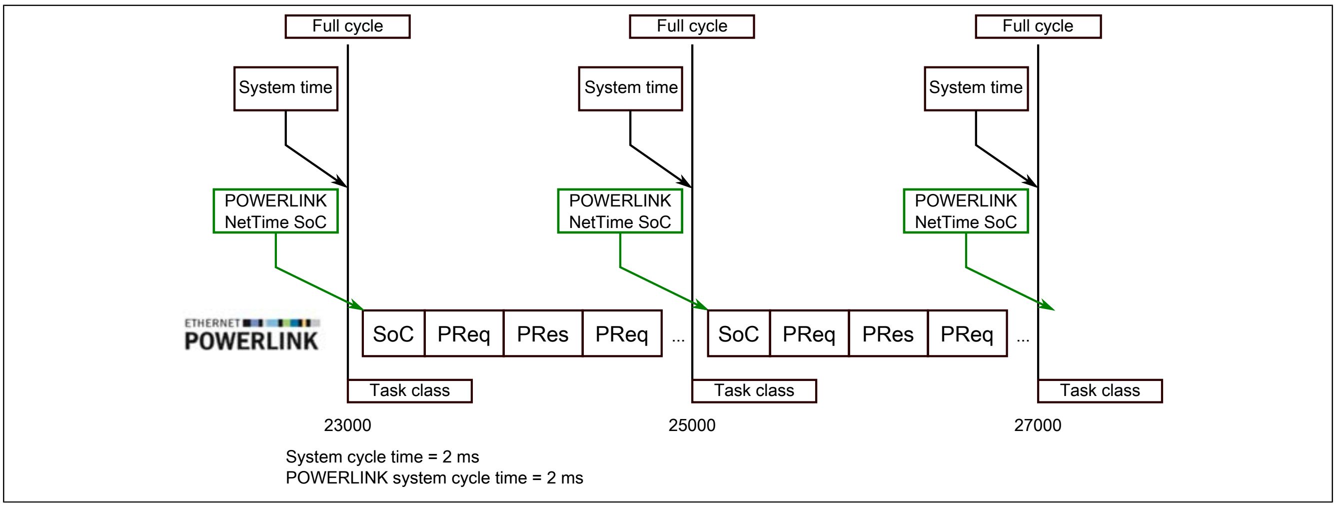
8.10.1.2 X2X Link reference time



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

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

8.10.1.3 POWERLINK reference time

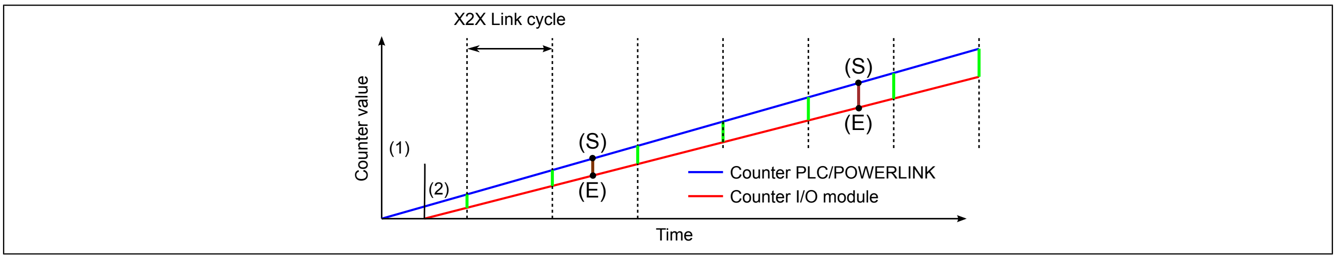


The reference time at POWERLINK is always formed at the SoC (Start of Cycle) of the POWERLINK network. The SoC starts 20 µs after the system tick. This results in the following difference between the system time and the POWERLINK reference time:

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

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

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



At startup, the internal counters for the PLC/POWERLINK (1) and the I/O module (2) start at different times and increase the values at µs intervals.

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

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

Note

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

8.10.2 Timestamp functions

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

8.10.2.1 Time-based inputs

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

> **Information:**
>
> **The determined time always lies in the past.**

8.10.2.2 Time-based outputs

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

> **Information:**
>
> **The specified time must always be in the future and the set X2X Link cycle time must be taken into account for the definition of the time.**

8.10.2.3 Time-based measurements

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

8.11 Minimum cycle time

The minimum cycle time defines how far the bus cycle can be reduced without causing a communication error or impaired functionality. It should be noted that very fast cycles decrease the idle time available for handling monitoring, diagnostics and acyclic commands.

Minimum cycle time
200 µs

8.12 Minimum I/O update time

The minimum I/O update time defines how far the bus cycle can be reduced while still allowing an I/O update to take place in each cycle.

Minimum I/O update time
1 ms