

X20IF10A1-1

1 General information

The interface module is equipped with an ASi master interface. This allows third-party components to be integrated in the B&R system and makes it possible to quickly and easily transfer data in both directions.

The interface module can be operated in X20 CPUs or in the expandable POWERLINK bus controller X20BC1083.

- ASi interface master
- Electrically isolated
- 4-pin bus connector

2 Order data


Order number	Short description	Figure
	X20 interface module communication	
X20IF10A1-1	X20 interface module, for DTM configuration, 1 ASi master interface, electrically isolated, order 1x terminal block TB704 separately!	
	Required accessories	
	Terminal blocks	
0TB704.9	Accessory terminal block, 4-pin, screw clamp terminal block 2.5 mm ²	
0TB704.91	Accessory terminal block, 4-pin, push-in terminal block 2.5 mm ²	

Table 1: X20IF10A1-1 - Order data

3 Technical data

Order number	X20IF10A1-1
Short description	
Communication module	ASi interface master
General information	
B&R ID code	0xA718
Status indicators	Module status, network status, data transfer
Diagnostics	
Module status	Yes, using LED status indicator and software
Network status	Yes, using LED status indicator and software
Data transfer	Yes, using LED status indicator
Fieldbus current consumption	Max. 27 mA
Power consumption	
Bus	1.1 W
Fieldbus	0.85 W
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
DNV GL	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
Interfaces	
Interface IF1	
Fieldbus	ASi interface master
Type	ASi master profile M4
Variant	4-pin male multipoint connector
Power supply	ASi power supply ¹⁾
Voltage range	24 to 32 V
Controller	netX100
Max. number of slaves	62
Max. distance	
Standard	100 m
With additional components	500 m
Max. cycle time	5 ms
Response time	Typ. 3 ms
Electrical properties	
Electrical isolation	PLC isolated from AS (IF1)
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

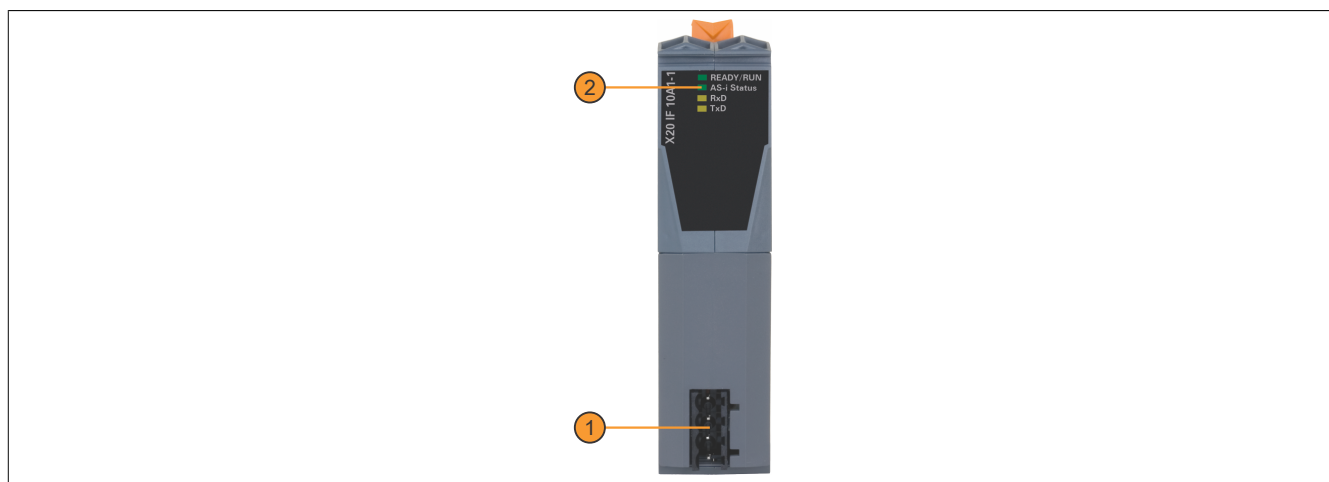
Table 2: X20IF10A1-1 - Technical data

Order number	X20IF10A1-1
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 TB704 separately.
Slot	In the X20 CPU and expandable bus controller X20BC1083

Table 2: X20IF10A1-1 - Technical data


- 1) An ASi-specified power supply unit is required for the power supply of the ASi interface.

4 Operating and connection elements




1	IF1 - ASi interface	2	LED status indicators
---	---------------------	---	-----------------------

4.1 LED status indicators

Figure	LED	Color	Status	Description
	READY/RUN	Green/red	Off	No power to module
		Green	On	PCI bus communication in progress
		Red	Blinking	Error when booting
	ASi status	Green/red	On	Communication on the PCI bus has not yet been started
			Blinking	Configuration not found for this channel
		Green	Blinking quickly	Configuration mode is active
			Blinking	Communication has stopped
			On	Configuration error, data exchange is active
	Rx-D	Red	Blinking	Configuration error-free, data exchange is active
			On	ASi interface power failure
			On	Fatal system error or hardware error
	RxD	Yellow	Flickering or on	The module is receiving data via the ASi interface.
	TxD	Yellow	Flickering or on	The module is transmitting data via the ASi interface.

4.2 ASi interface (IF1)

Interface	Pinout	
 4-pin male multipoint connector	Terminal	Explanation
	1	ASi+
	2	ASi+
	3	ASi-
	4	ASi-

5 Use in the expandable X20BC1083 POWERLINK bus controller

5.1 Cyclic data

If this module is connected to the expandable POWERLINK bus controller, the amount of cyclic data is limited by the POWERLINK frame to 1488 bytes in each direction (input and output).

When using multiple X20IF10xx-1 interfaces or other X2X modules with a POWERLINK bus controller, the 1488 bytes are divided between all connected modules.

5.2 Operating netX modules

It is important to note the following in order to operate netX modules with the bus controller without problems:

- A minimum revision $\geq E0$ is required for the bus controller.
- netX modules can only be operated with the POWERLINK V2 setting. V1 is not permitted.
- With SDO access to POWERLINK object 0x1011/1 on the bus controller, the netX firmware and the configuration stored on the bus controller are not reset. They can only be overwritten by accessing them again. This affects objects 0x20C0 and 0x20C8, subindexes 92 to 95.

5.3 Timing characteristics

The internal data transfer results in an additional runtime shift of one cycle per direction.

Information:

For additional information about runtime behavior, see section "Runtime shift" in X20BC1083.

6 netX error codes

netX modules return an error code when an error occurs. These error codes are fieldbus-specific. A complete list of all error codes in PDF format is available in Automation Help in section "Communication / Fieldbus systems / Support with FDT/DTM / Diagnostic functions / Diagnostics on the runtime system / Master diagnostics" under item "Communication_Error".

7 Firmware

The module comes with preinstalled firmware. The firmware is part of the Automation Studio project. The module is automatically brought up to this level.

To update the firmware contained in Automation Studio, a hardware upgrade must be performed (see "Project management / Workspace / Upgrades" in Automation Help).

8 The ASi interface

ASi stands for "Actuator Sensor Interface" and is a bus system for the lowest field level of automation technology. Using ASi bus systems provides an easy and affordable way to connect, operate and service sensors and actuators.

ASi is particularly suitable for safety-related components such as safety monitors, emergency stop switches or door locks.

The bus system is composed of a 2-conductor line that transfers both power and information at the same time. This eliminates the need for parallel wiring, where each individual sensor or actuator is connected to the controller's input or output module via a separate line.

ASi is a single master system, meaning only one master can be operated in a network at a time. The communication between master and slave works via the cyclic polling method. The master sends a poll request to each slave, which is answered by the individual slaves with a poll response.

Either 31 or 62 slaves can be operated depending on the type of addressing.

- Standard addressing is suitable for 1 to 32 slaves.
- Extended addressing is suitable for up to 62 slaves. Here, the slaves are grouped as A and B slaves, i.e. 1A to 31A and 1B to 31B).

Information:

Only ASi slaves with the ID code A can be addressed in extended form. With an ID code other than A, only the standard addressing of up to 32 slaves can be used.

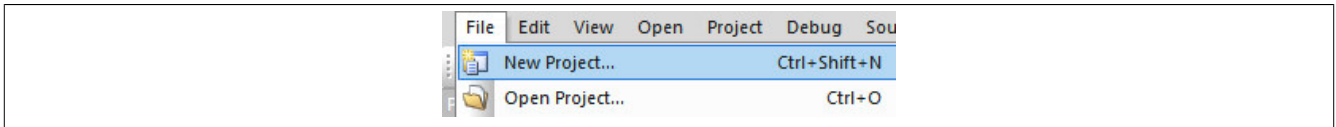
8.1 Settings in Automation Studio

The interface module can be operated in the slot of a CPU or in the slot of an expandable POWERLINK bus controller.

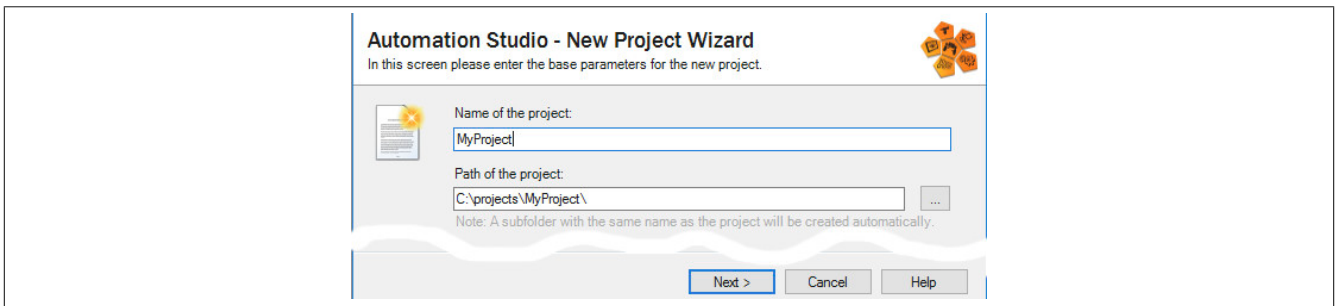
To do this, a new Automation Studio project is created and the suitable settings are made on the module.

8.1.1 Creating an Automation Studio project

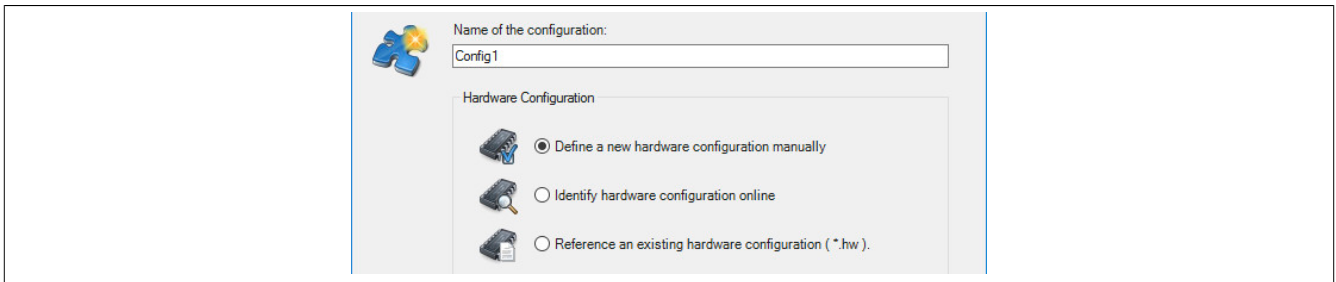
- Create a new Automation Studio project by selecting "New project".



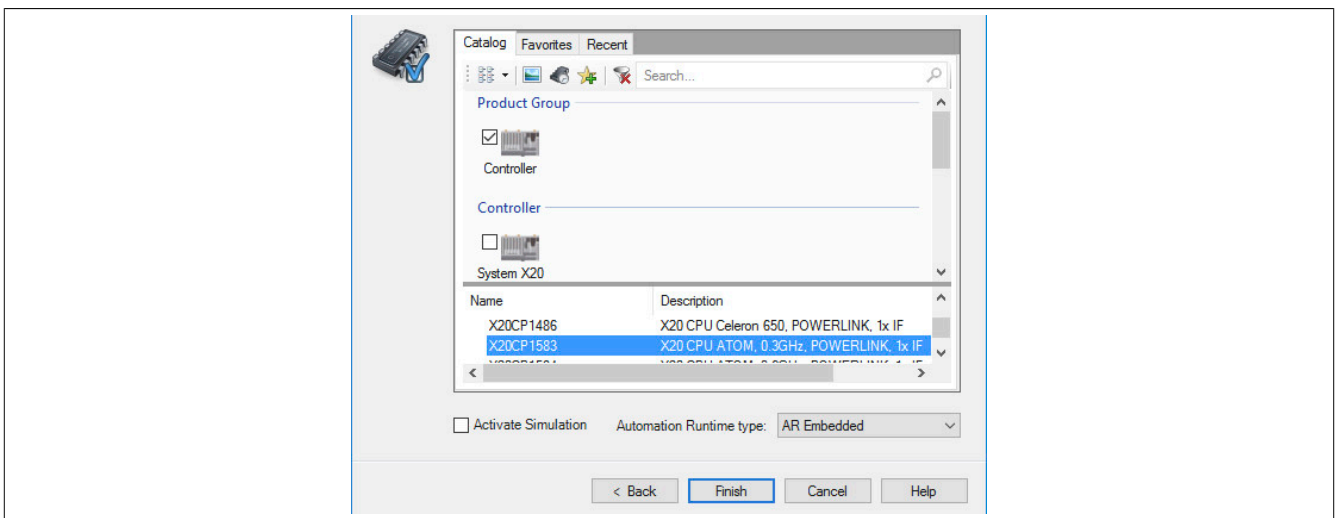
- Assign a project name and set up the project path.



- Assign the hardware configuration type and configuration name.

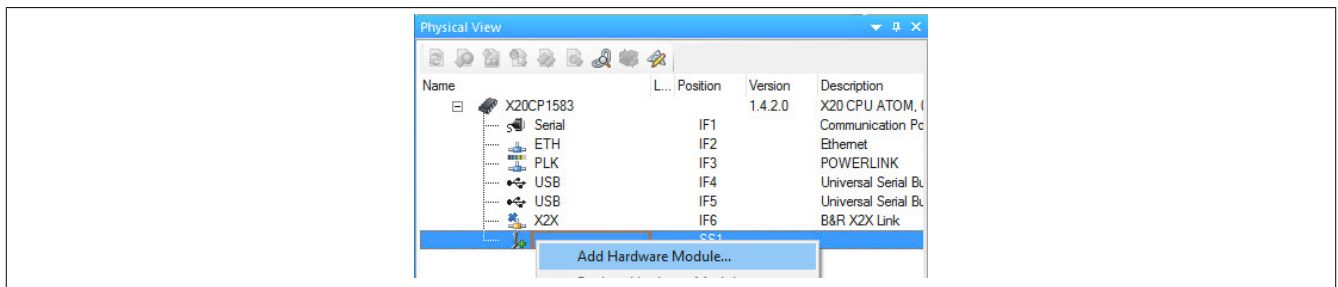


- Select the hardware in the next step if "Define a new hardware configuration manually" was selected. In order to simplify the search, different filters can be set in the Hardware Catalog. Lastly, highlight the required hardware and create the Automation Studio project by clicking on "Finish".

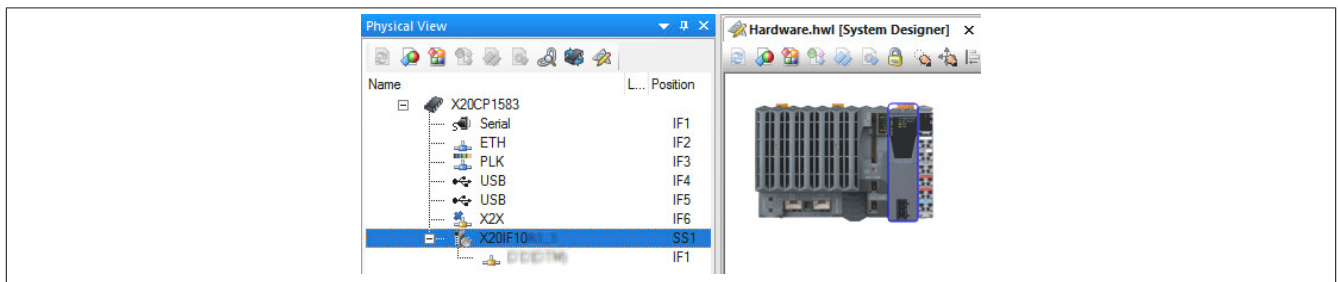


8.1.2 Adding an interface module

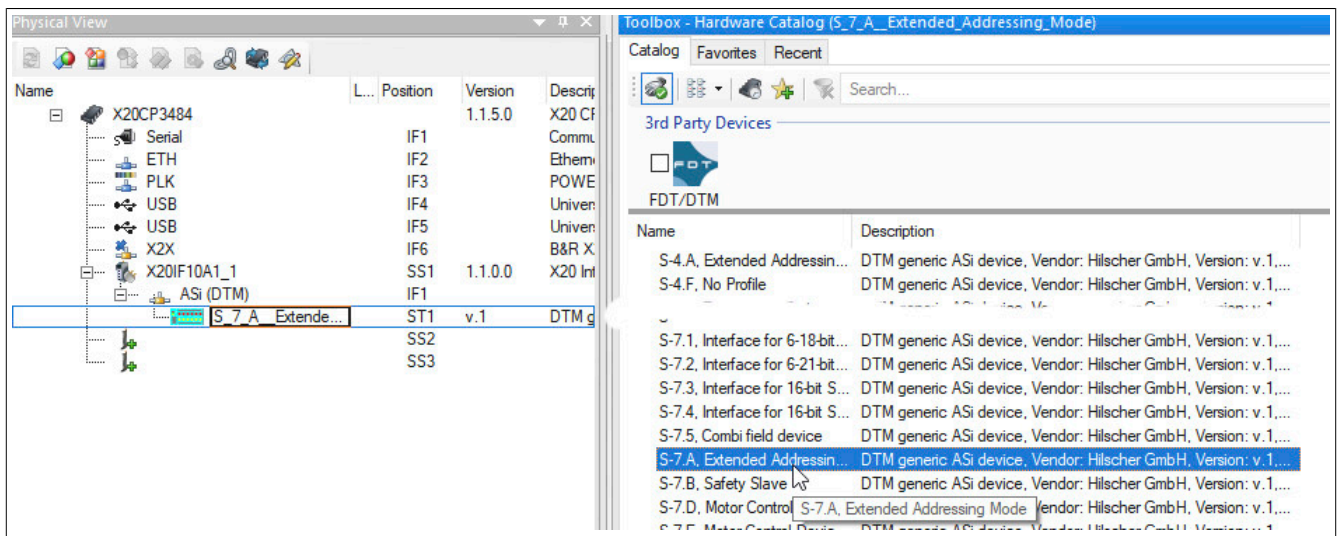
- In this example, the interface card is connected in the slot of a CPU. Right-clicking on the slot and selecting "Add hardware module" opens the Hardware Catalog.



- The module is added to the project via drag-and-drop or by double-clicking on the interface card.

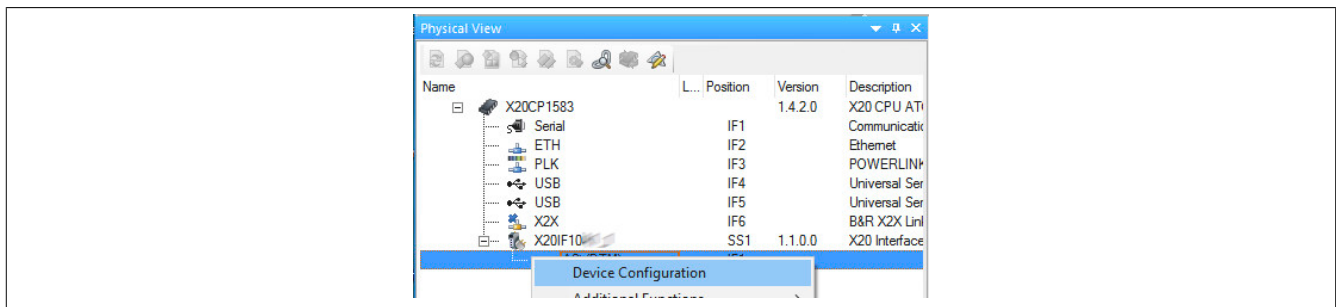


- ASi slaves from the Hardware Catalog containing the required ASi profile can now be connected to the ASi master via drag-and-drop.

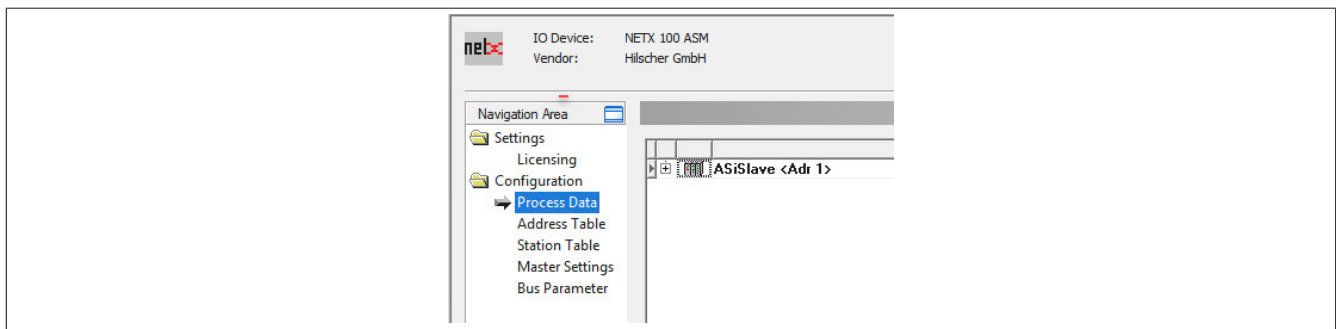


8.1.3 Configuring the ASi master

- Additional module settings can be made under "Device configuration". This configuration environment is opened by right-clicking on the IF interface and selecting "Device configuration".



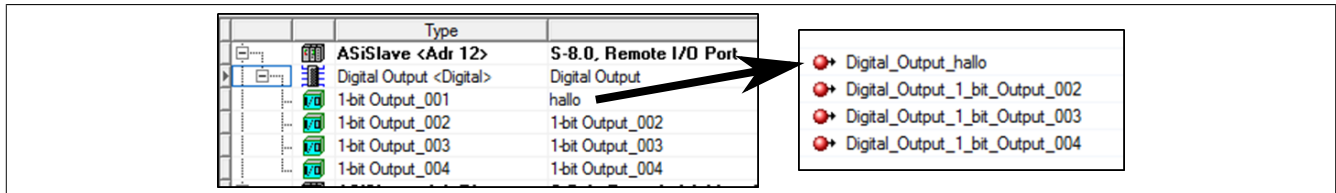
- General settings are made in the device configuration.



Process data

This table lists the process data of all attached slaves (device description files).

Parameter	Explanation
Type	Device designation specified by the hardware. Further description of modules configured on the device or the input or output signals.
Tag	The name of the input and output data can be changed in column "Tag".
SCADA	This parameter is not supported.



Address table

Contains a list of all slaves which are categorized according to their input and output data.

"Display mode" allows toggling between decimal and hexadecimal display.

Parameter	Explanation
Station address	Station address of the assigned slave device.
Device	Current device name of the assigned slave
Name	Device name of the assigned slave
Module	Name of the module
Type	Input or output type
Length	Number of bytes inserted (IB, QB, IW or QW).
Address	Input or output data offset address

The address table can also be exported as a CSV file.

Station table

All ASi slaves are listed here.

Parameter	Explanation
Activate	This allows the slaves to be enabled or disabled. <ul style="list-style-type: none"> Enabled: Process memory is reserved and data is exchanged. Disabled: The master reserves memory in the process data image for the slave, but no data is exchanged.
Station address	Station address of the slave. Valid addresses: <ul style="list-style-type: none"> Standard address range: 1 to 31 Extended address range: 1A to 31A and 1B to 31B. In the application, this corresponds to the ranges 1 to 31 and 32 to 62. A/B slaves contain an "A" in their ID code.
Device	Name of the assigned slave
Name	Name of the assigned slave
Vendor	Vendor information

Master settings

- Start of bus communication

It is possible to select how data exchange is started on the module.

Parameter	Explanation
Automatically by device	Data exchange is started automatically after initializing this module.
Controlled by application	Data exchange is started by Automation Runtime.

- Module alignment

The addressing mode is defined by the process image here. The addresses (offsets) of the process data are always interpreted as byte addresses.

Addressing mode	Explanation
Byte boundaries	The module address can start on any offset.
2 byte boundaries	The module address can only start on even byte offsets.

Information:

This configuration is automatically managed by Automation Runtime and is not permitted to be changed (default setting).

- Application monitoring

The module-internal watchdog time can be set here. If the watchdog has been enabled (watchdog time not equal to 0), the hardware watchdog must be reset after the set time at the latest.

Parameter	Explanation	Values
Watchdog time	Watchdog software disabled	0 ms
	Permissible range of values	20 to 65535 ms
	Default value: 1000 ms	

Information:

The watchdog time is reset automatically by Automation Runtime.

- Process image storage format

This is used to define how data is stored in the process image (I/O mapping). The storage format is only applied to data type "Word". This change has no effect on other data types.

Storage format	Explanation
Big-endian	MSB/LSB = Higher/Lower byte (Motorola format)
Little-endian	MSB/LSB = Higher/Lower byte (Intel format)

Information:

This configuration is automatically managed by Automation Runtime and is not permitted to be changed (default setting).

- Process data handshake

This parameter configures the handshake for the data exchange between application and device. Only "Buffered, host-controlled" is supported here.

Bus parameters

— Behavior in case of defective slave device

The selection depends on the settings under "Behavior during startup sequence".

- **Don't take care about the status of any connected slave devices** - The slave status is disregarded but the communication remains intact.
- **Stop communication if a device is missing** - Communication is stopped if a slave is missing.
- **Stop communication if a device is reporting a periphery failure** - Communication is stopped if a slave reports a peripheral fault.
- **Stop communication if a device is missing or reporting a periphery failure** - Communication is stopped if a slave is missing or a peripheral fault is reported.

— Behavior during startup sequence

Determines the procedure of the master during startup in correlation with the connected slaves.

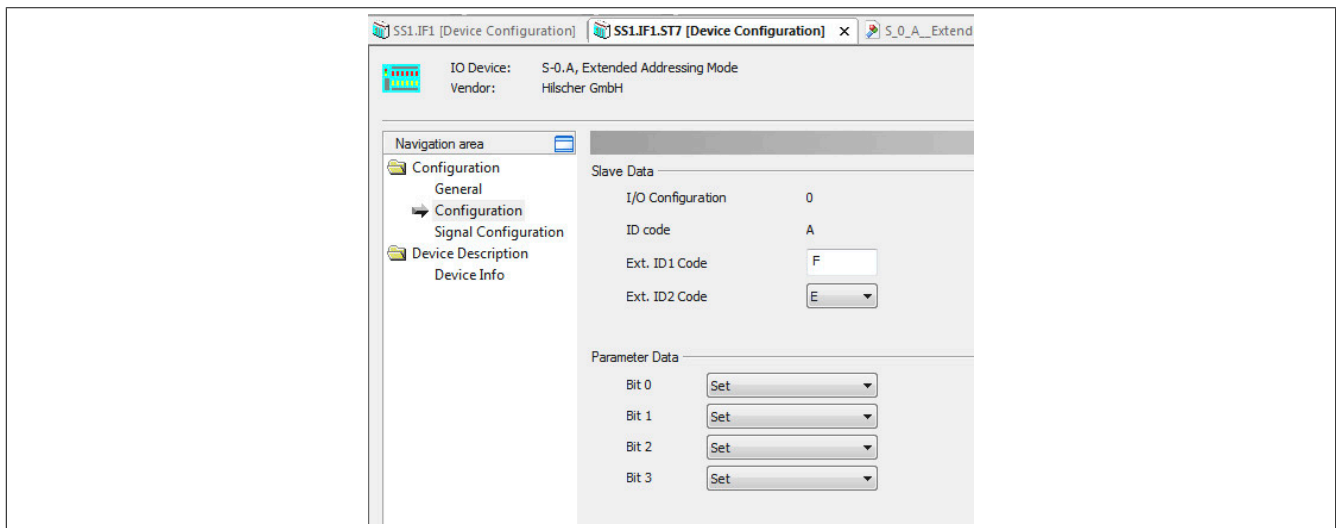
- **Protected mode** - If enabled, the master enters protected data exchange mode. If not enabled, the master enters configuration mode.
- **Auto address assignment** - If enabled, the master assigns a slave the address of a missing slave if it has an identical I/O, ID, ID1 and ID2 code and address 0.

— Overwrite configuration database

This parameter is not supported.

8.1.4 Configuring the ASi slave

- General settings can be made on the ASi slave using "Device configuration."



General

The description can be adjusted here. However, this is only used by the configuration dialog boxes and not by Automation Studio.

The slave address set on the master is also displayed. See "Station table " on page 9 for setting the address.

Configuration

- Slave data

Parameter	Explanation
I/O configuration	I/O configuration used in the EDS file. This corresponds to the first position of the slave profile.
ID code	I/O code used in the EDS file. This corresponds to the second position of the slave profile.
Ext. ID1 code	Extended ID code, user-specific
Ext. ID2 code	Extended ID code, user-specific

- Parameter data

Parameter	Explanation	Values
Bit 0 to bit 3	Parameter data to be set by the user. If these are contained in the EDS file, they are displayed here.	Set Reset

Signal configuration

The name and data type of the I/O data point can be adjusted here.

The signal configuration can be set to the default values for the specified slave profile by clicking on "Default".

Parameter	Explanation
Modules	Input or output modules of the signal configuration
Tag	Editable name of the individual input or output signals. The name can also be changed via the master (see "Process data" on page 8)
Data type	Editable data type of the individual input or output signals. The data type can be selected from a list according to the AS slave profile defined in the EDS file.
Type	Type of input or output signal

Device description

General device information can be read here.

8.2 EDS device description file

All possible ASi slave variants have already been imported into Automation Studio as device description files and can be taken from the Hardware Catalog.

Each ASi slave is assigned an ASi profile. The ASi profile is composed of 4 components:

- **I/O configuration**
Contains information about the configuration of individual ASi slave ports: output, input or bidirectional input/output.
- **ID code**
Contains the ID code of the slave.
- **Extended ID code 1**
Length: 4 bits
In extended address mode, the MSB indicates whether it is an A (MSB = 0) or B (MSB = 1) slave (starting with specification 3.0). In the Automation Studio Hardware Catalog, the extended address mode can be identified by an "A" in the slave address (e. g. "S-0. A"). The lower 3 bits contain additional slave-specific information.
- **Extended ID code 2**
Used for slave-specific settings. See description of the corresponding slave.

8.3 Slave addressing

There are 3 ways to assign an address to an ASi slave.

- With the programming device

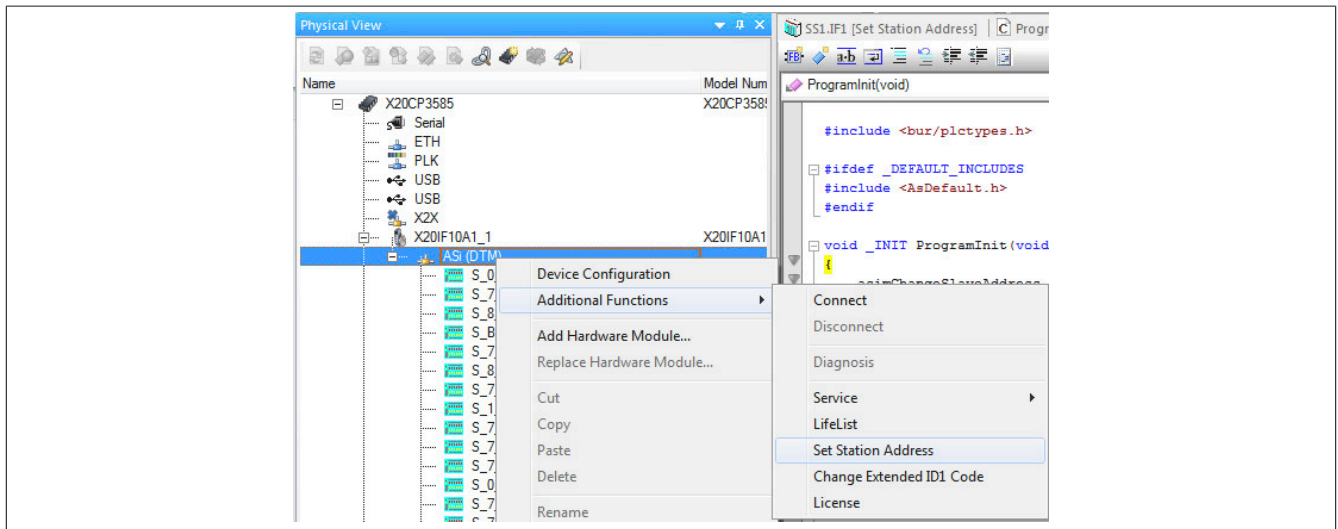
A programming interface is required for this, which does not exist for every slave. If a programming interface is installed in the ASi slave, the slave address can be adjusted via the programming device and the ASi profile can also be read.

- Assignment via function block

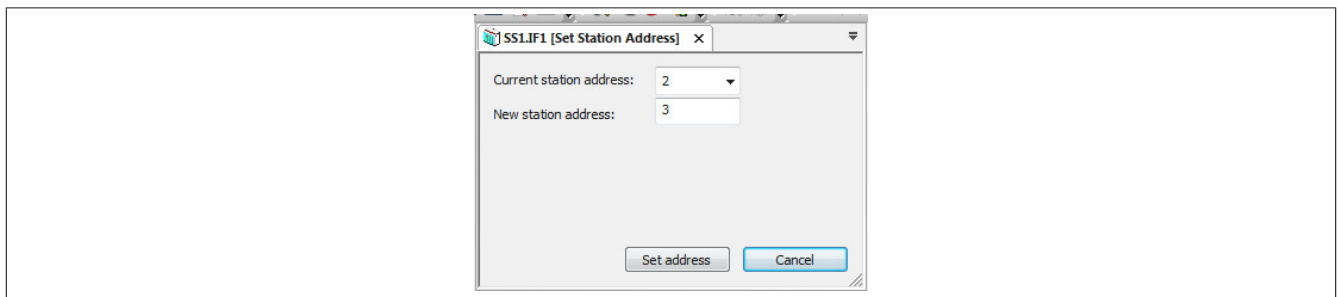
To do this, a program must be created and function block **AsNxAsiM - asimChangeSlaveAddress()** must be programmed. In order to change the slave address using this function block, the current address must be known. In most cases, address 0 (delivery state) is applied to a new ASi slave since no slave is permitted to have address 0.

- Direct mapping via the master

The current slave address is also required here. In the Physical View (Automation Studio), the slave address can be adjusted with a right-click on the ASi interface of the master → Additional functions → Set station address.



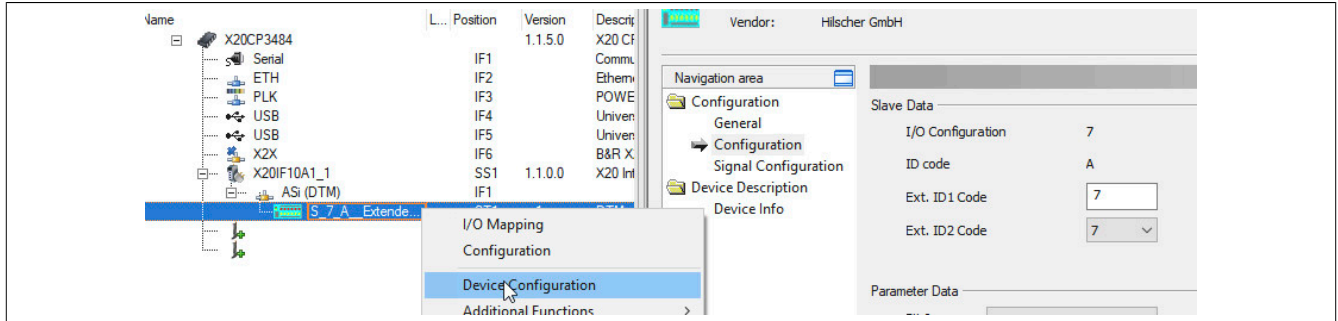
First, the slave to be addressed is selected and the desired new address is specified. A new address can be assigned using "Set address".



8.4 Configuration example

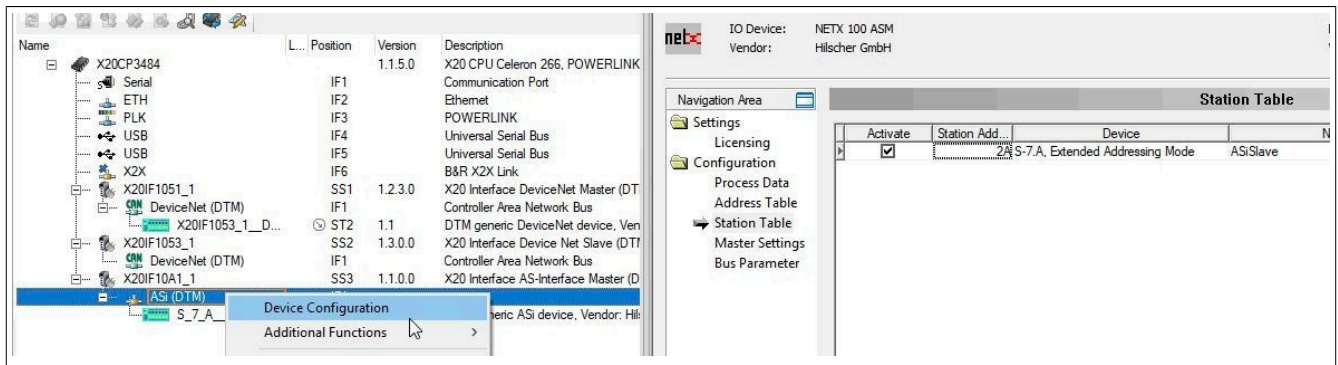
In this example, an ASi slave with the profile S-7.A is configured.

- The desired profile is first selected from the Hardware Catalog and added to the ASi master. See ["Adding an interface module" on page 7](#).
- The ID code settings of the connected device are specified by the ASi slave manufacturer and must be set in the profile. The values under "Ext. ID1 code" and "Ext. ID2 code" are therefore entered via the device configuration of the device description file.



- The address of the ASi slave is then set in the device configuration on the master under "Station table".

The address on the device itself is set differently depending on the ASi slave used, e.g. via a programming device, function block or the master. See ["Slave addressing" on page 12](#).



- Confirm and save all settings with OK. The configuration is transferred to the CPU and, if correct, a connection between master and slave is automatically established.

The connection status can be checked with the ModulOK bit in the I/O mapping of the device description file. When ModulOK = True, data can be exchanged between master and slave.

