OPERATING INSTRUCTIONS

Sensor Integration Gateway - SIG300

Network devices

PROFINET

SICK
Sensor Intelligence.

Described product

SIG - Sensor integration gateway PROFINET

Manufacturer

SICK AG
Erwin-Sick-Str. 1
79183 Waldkirch
Germany

Production location

SICK PCA
55438 Minneapolis, MN
USA

Legal information

Original document

This document is an original document of SICK AG.

Contents

1 About this document

1.1 Information on the operating instructions

Read these operating instructions carefully before starting any work in order to familiarize yourself with the product and its functions.

The operating instructions are an integral part of the product and should remain accessible to the personnel at all times. When handing this product over to a third party, include these operating instructions.

These operating instructions do not provide information on the handling and safe operation of the machine or system in which the product is integrated. Information on this can be found in the operating instructions for the machine or system.

1.2 Further information

You can find the product page with further information via the SICK Product ID:
pid.sick.com/{P/N}/{S/N}
(see "Product identification via the SICK product ID", page 11).

The following information is available depending on the product:

- This document in all available language versions
- Data sheets
- Other publications
- CAD files and dimensional drawings
- Certificates (e.g., declaration of conformity)
- Software
- Accessories

1.3 Symbols and document conventions

Warnings and other notes

DANGER
Indicates a situation presenting imminent danger, which will lead to death or serious injuries if not prevented.

WARNING
Indicates a situation presenting possible danger, which may lead to death or serious injuries if not prevented.

CAUTION
Indicates a situation presenting possible danger, which may lead to moderate or minor injuries if not prevented.

NOTICE
Indicates a situation presenting possible danger, which may lead to property damage if not prevented.

NOTE
Highlights useful tips and recommendations as well as information for efficient and trouble-free operation.

Instructions to action

► The arrow denotes instructions to action.

1. The sequence of instructions is numbered.
2. Follow the order in which the numbered instructions are given.

✓ The tick denotes the results of an action.

2 Safety information

2.1 General safety notes

Please observe the safety notes and the warnings listed here and in other sections of this product documentation to reduce the possibility of risks to health and avoid dangerous situations.

CAUTION
Failure to observe the relevant work safety regulations may lead to physical injury or cause damage to the system.

Integrating the product

NOTICE
The product must be integrated correctly to provide the expected functionality.

- ► Plan the integration of the product in accordance with the machine requirements (project planning).
- ► Implement the integration of the product in accordance with the project planning.

Repairs and modifications

NOTICE
Improper work on the product

A modified product may not provide the expected functionality.

- ► Apart from the procedures described in this document, do not repair, open, manipulate or otherwise modify the product.

2.2 Notes on the product

Table 1: Notes on the product

Symbol	Meaning
	Connection, mounting, and setting may only be performed by skilled person.
	Not a safety component in accordance with the EU Machinery Directive.
	When commissioning, protect the device from moisture and contamination.

2.3 Intended use

The SIG300 is a remote IO-Link input and output module for connecting an Ethernet network.

Intended use requires that the device is used industrially indoors without any specific climatic and atmospheric requirements. Operation of the device according to its intended use and enclosure rating IP67 are only guaranteed if open male and female connectors are sealed with blind plugs. Intended use also includes EMC-compliant electrical installation. If the product is used for any other purpose or modified in any way, all warranty claims against SICK AG will be void.

NOTE
This document is aimed at the trained specialist personnel. Qualified specialist personnel are persons who are familiar with work such as the installation and operation of the product, and who have the necessary qualifications for this activity. All claims against the manufacturer in respect of warranty and liability shall be invalidated in the event of damage resulting from unauthorized manipulation or incorrect use. The operating entity is responsible for ensuring that the work safety regulations and work safety regulations applicable in the specific individual case are observed.

Incorrect use, improper modification or manipulation of the product will invalidate any warranty from SICK; in addition, any responsibility and liability of SICK for damage and secondary damage caused by this is excluded.

2.4 Improper use

- The device is not a “safety-component” as defined in the EC Machinery Directive (2006/42/EC).
- The device must not be used in explosion-hazardous areas.
- Any other use that is not described as intended use is prohibited.
- Any use of accessories not specifically approved by SICK AG is at your own risk.

The device is not suitable for the following applications (this list is not exhaustive)

- As a safety device to protect persons, their hands, or other body parts
- Underwater
- In explosion-hazardous areas
- Outdoors, without additional protection

WARNING
Improper use

Any improper use can result in dangerous situations. Therefore, observe the following information:

- The device should be used only in line with intended use specifications.
- All information in these operating instructions must be strictly complied with.

2.5 Qualification of personnel

Any work on the product may only be carried out by personnel qualified and authorized to do so.

Qualified personnel are able to perform tasks assigned to them and can independently recognize and avoid any potential hazards. This requires, for example:

- technical training
- experience
- knowledge of the applicable regulations and standards

2.6 Notes on UL approval

UL Environmental Rating: Enclosure type 1

Table 2: Notes on Approvals

Product standard	EN 61131-2 Programmable logic controller, Part 2 Compliant
CE	2014/30/EU Compliant 2011/65/EU Compliant
UKCA	Compliant
EMC	2014/30/EU Compliant

REACH	Compliant
WEEE	2012/19/EU Compliant
RoHS	2011/65/EU & 2015/863 – Exception 6c&7a
China RoHS	SJ/T 11364-2014 25 EPUP
KC	Compliant

3 Product description

3.1 Scope of delivery

Included in the scope of delivery of the SIG300:

- SIG300 IO-Link Master module
- Protection caps

NOTE
No screws are included in the scope of delivery.

3.2 Product identification via the SICK product ID

SICK product ID

The SICK product ID uniquely identifies the product. It also serves as the address of the web page with information on the product.

The SICK product ID comprises the host name pid.sick.com, the part number (P/N), and the serial number (S/N), each separated by a forward slash.

For many products, the SICK product ID is displayed as text and QR code on the type label and/or on the packaging.

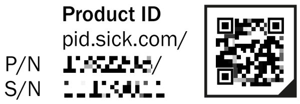


Figure 1: SICK product ID

3.3 General information

The SIG300 IO-Link Master is a gateway for connecting IO-Link devices as well input and/or output digital signals for data integration into an Ethernet network.
The data can be transmitted to a network via the integrated IIOT interfaces (REST API or MQTT).
It is intended for use in industrial environments that require enclosure rating up to IP67.

The module has 8 IO-Link Master channels and offers up to 16 configurable DI/DO. The device is connected to an M12 female connector that can operate Class A and Class B connection type.

In addition, the SIG300 has a powerful user interface that can be accessed via the web interface. This is used to parameterize the SIG300 and the connected devices.

The SIG300 can be commissioned using the following methods:

- Integrated web interface
  It provides direct access for parameterization via a suitable web interface on devices connected to the same Ethernet network as the SIG300 or via the SIG300 USB interface.
- SENTIO CREATOR software application via USB or Ethernet connection.
- Via Ethernet using REST API interfaces that provide direct access for higher-level automation operations.

3.3.1 IO-Link

IO-Link is a standard (IEC 61131-9) that can be used to connect intelligent devices at the sensor and actuator level to an automation system.

The SIG300 complies with IO-Link Interface and System Specification V1.1.3.

Communication

Communication takes place between a master and a device. An IO-Link Master contains one or more ports. One device can be connected per port, which means that IO-Link is point-to-point communication and not a fieldbus. The IO-Link Master forms the interface between the higher-level fieldbus level and the IO-Link system.

IO-Link is functional and enables advanced diagnostics of sensors and actuators or simple and fast parameterization through bidirectional communication. The IO-Link devices are connected to the master via unshielded 3-, 4- or 5-wire standard cables of a maximum length of 20 m.

The SIG300 supports IO-Link communication at the following speeds:

- COM 1 → 4,800 baud
- COM 2 → 38,400 baud
- COM 3 → 230,400 baud

The module automatically selects the communication speed that matches the IO-Link device.

IO-Link modus (IOL)

IO-Link communication (C/Q) is activated at pin 4 of each IO-Link ports, so an IO-Link device can be connected.

Fieldbus communication (IO-Link call)

The following tasks can be performed:

- Parameterization/Configuration of an IO-Link device during operation.
- Diagnosis of an IO-Link device by reading out diagnostic parameters.
- Execution of IO-Link port functions.
- Saving/Restoring of IO-Link device parameters.

Data storage mode

The data storage mode allows IO-Link devices to be exchanged without any configuration in the event of service. Both the IO-Link Master and the IO-Link device store the device parameters. During data storage, these different parameter data memories are synchronized.

In the event of device replacement, the master writes the stored device parameters to the new device. The application can be restarted without further intervention via a configuration tool or the like.

If the IO-Link Master is replaced, the new master reads the parameters from the IO-Link device and saves them. The “Backup and Restore” option in "Validation and backup" menu must be activated for this purpose. The application can also be restarted here without further intervention via a configuration tool or the like.

The data storage mode is only available for devices that comply with IO-Link version V1.1 and higher.

3.3.2 IoT interfaces (dual talk)

NOTE

The interfaces REST API JSON and MQTT JSON require the following firmware version or newer:

V1.8

The SIG300 features Dual Talk functionality, which allows the user to address the module from IT networks and integrate it into Internet-of-Things applications.

The SIG300 contains the following interfaces:

- REST API JSON
- MQTT JSON

REST API

The Representational State Transfer – Application Programming Interface (REST API) is a programmable interface that uses HTTP requests for GET and POST data. This allows access to detailed device information. The format is JSON.

The REST API interface of the SIG300 complies with the JSON Integration for IO-Link standard version V1.0.0 published by the IO-Link community.

MQTT

The MQTT (Message Queuing Telemetry Transport) protocol is an open network protocol for machine-to-machine communication that enables the transmission of telemetric data between devices.

An MQTT client is integrated in the SIG300, which enables the device to publish certain information to an MQTT broker. The format is JSON.

The publication of messages can either take place periodically or based in an event of a value change.

The MQTT interface of the SIG300 complies with the JSON Integration for IO-Link standard version V1.0.0 published by the IO-Link community.

3.3.3 Web interface

The SIG300 has an integrated web server that provides functions for device configuration and diagnostic information via a web interface.

see "Web-Interface", page 54

3.4 Operating elements and status indicators

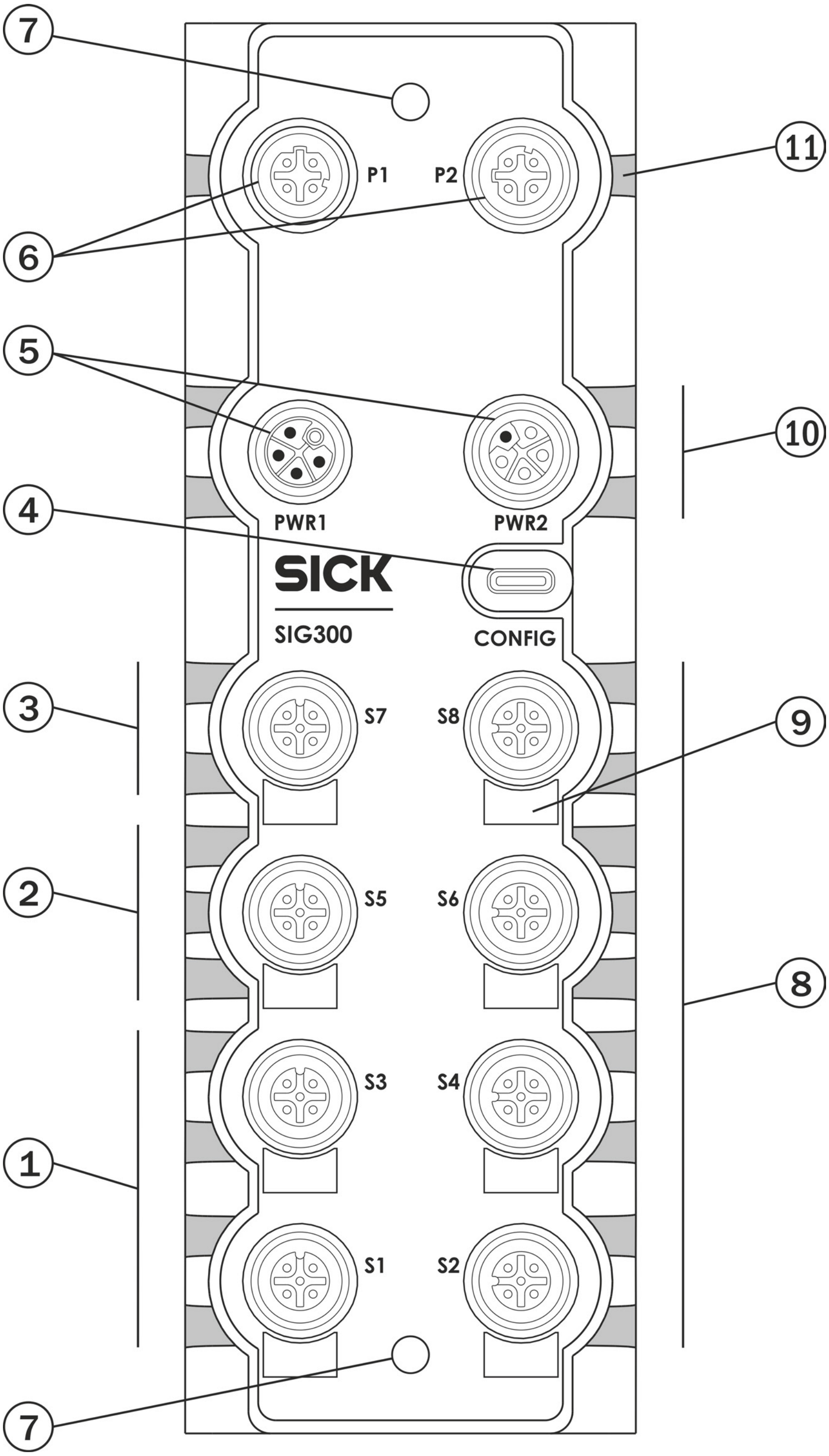


Figure 2: Operating and status indicators

① IO-Link ports (S1...S4) class A1

② IO-Link ports (S5, S6) class A2

③ IO-Link ports (S7, S8) class B

④ USB port for configuration

⑤ Voltage supply (Input/Output)

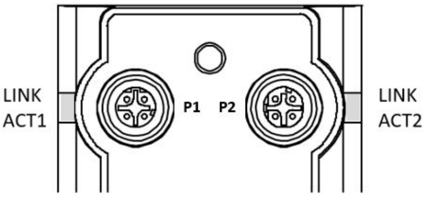
⑥ Ethernet Port (P1) and Ethernet Port (P2).

⑦ Mounting hole
⑧ Port Pin status LEDs
⑨ Port designation
⑩ Power status LEDs
⑪ Communication status LEDs

3.4.1 Status indicators

3.4.1.1 Network indication LEDs

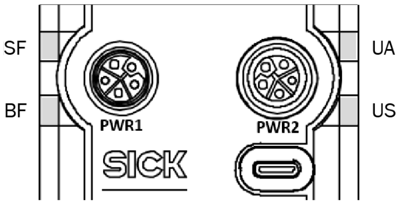
Table 3: Network indication LEDs

Picture	LED	Signal	Description
	LINK ACT 1 (Link / Activity 2)	○ Off	No network connection on port 1
		● Green	Network connection on port 1
		✹ Green	Data exchange activity
	LINK ACT 2 (Link / Activity 2)	○ Off	No network connection on port 2
		● Green	Network connection on port 2
		✹ Green	Data exchange activity

○ LED off. ✹ LED flashes. ● LED illuminates.

3.4.1.2 Power supply indication LEDs

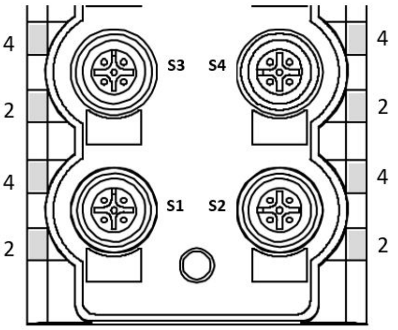
Table 4: Power supply indication LEDs

Picture	LED	Signal	Description
	SF System Failure	○ Off	• System not ok • Hardware or configuration error • Firmware problem • Configured IO-Link device not connected • Module error
		● Green	System ok
	BF Bus Failure	○ Off	Device connected and communicates successfully
		● Red	• no bus connection / connection to PLC lost • no ip adress • wrong name
	UA Actuator power supply (Only available for Port class B)	○ Off	No voltage: UA < 11.5V
		● Green	Voltage UA > 11.5V
	US Sensor and electronic power supply	○ Off	No voltage: UA < 11.5V
		● Green	Voltage US > 11.5V

○ LED off. ✹ LED flashes. ● LED illuminates.

3.4.1.3 IO-Link ports (S1 ... S4) class A1 LEDs

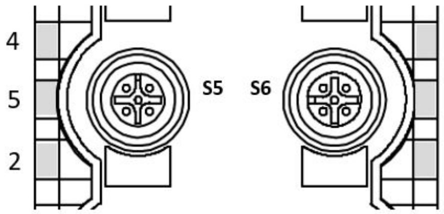
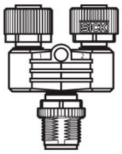
Table 5: IO-Link ports (S1 ... S4) class A1 LEDs

Picture	LED	Signal	Description
	4 IO-Link Or DI/DO LED for Pin 4	🟢 Green	IO-Link communication active or DI/DO state = HIGH
		✺ Green	No IO-Link communication
		○ Off	DI/DO state = Low
	2 DI/DO: LED for Pin 2	🟡 Yellow	DI/DO state = High
		○ Off	DI/DO state = Low

○ LED off. ✺ LED flashes. ● LED illuminates.

3.4.1.4 IO-Link ports (S5, S6) class A2 LEDs

Table 6: IO-Link ports (S5, S6) class A2 LEDs

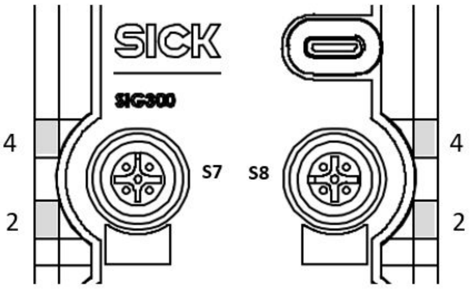
Picture	Picture legend	LED	Color	Description
	4	IO-Link Or DI/DO LED for Pin 4	🟢 Green	IO-Link communication active or DI/DO state = HIGH
			✺ Green	No IO-Link communication
			○ Off	DI/DO state = Low
	5[1]	DI/DO: LED for Pin 5	🟡 Yellow	DI/DO state = High
			○ Off	DI/DO state = Low
	2[1]	DI/DO: LED for Pin 2	🟡 Yellow	DI/DO state = High
			○ Off	DI/DO state = Low
	Note: Disable “Pin 5” if the “T-Connector” (YM2A15-000S01FY2A4) is used to connect two DI/DO devices.			

○ LED off. ✺ LED flashes. ● LED illuminates.

[1] Requires firmware version 1.5.3 or higher to be used.

3.4.1.5 IO-Link ports (S7, S8) class B LEDs

Table 7: IO-Link ports (S7, S8) class B LEDs

Picture	Picture legend	LED	Color	Description
	4	IO-Link Or DI/DO LED for Pin 4	● Green	IO-Link communication active or DI/DO state = HIGH
			✹ Green	No IO-Link communication
			○ Off	DI/DO state = Low
	2	UA (power supply actuator) on Pin 2	● Yellow	UA (actuator power supply) on Pin 2 active
			○ Off	UA (actuator power supply) on Pin 2 not active

○ LED off. ✹ LED flashes. ● LED illuminates.

4 Transport and storage

4.1 Transport

For your own safety, please read and observe the following notes:

NOTICE
Damage to the device due to improper transport.

- The device must be packaged for transport with protection against shock and moisture.
- Recommendation: Use the original packaging as it provides the best protection.
- Transport should be performed by specialist staff only.
- The utmost care and attention is required at all times during unloading and transportation on company premises.
- Note the symbols on the packaging.
- Do not remove packaging until immediately before you start mounting.

4.2 Transport inspection

Immediately upon receipt at the receiving work station, check the delivery for completeness and for any damage that may have occurred in transit. In the case of transit damage that is visible externally, proceed as follows:

- Do not accept the delivery or only do so conditionally.
- Note the scope of damage on the transport documents or on the transport company’s delivery note.
- File a complaint.

NOTE
Complaints regarding defects should be filed as soon as these are detected. Damage claims are only valid before the applicable complaint deadlines.

4.3 Storage

Store the device under the following conditions:

- Recommendation: Use the original packaging.
- Do not store outdoors.
- Store in a dry, dust-protected place.
- To allow any residual dampness to evaporate, do not package in airtight containers.
- Do not expose to aggressive substances.
- Protect from sunlight.
- Avoid mechanical shocks.
- Temperature and relative humidity, see "General technical data", page 104
- For storage periods longer than 3 months, regularly check the general condition of all components and the packaging.

5 Mounting

5.1 Prerequisites

The following requirements must be met when mounting the SIG300:

- Level mounting surface free of mechanical stress.
- Select suitable mounting location with regard to vibration and impact load, temperature and humidity see "General technical data", page 104.
- Protected to prevent the connecting cables from being torn off by personnel or the device.
- For proper installation keeping a minimum distance of 3 mm between two modules is recommended.
- When using angled plug connectors, a minimum distance of 50 mm must be maintained between two modules.
- Mount modules in such a way that they cannot be used as climbing aids.

NOTE
To ensure IP67 protection, all unused connections must be covered with sealing caps. These are included in the scope of delivery and can be also ordered separately. Suitable sealing caps can be found at SICK.com (“Protective caps kit” part number 2136242). In addition, connected cables and sealing caps must be fastened with the appropriate torque (see manufacturer’s specifications).

5.2 Module mounting

The SIG300 is mounted using two screws (M4) and two washers.

The fixing screws and tightening torques depend on the substrate of the mounting location. Always tighten the screws carefully and observe the maximum permissible tightening torque of 1 Nm.

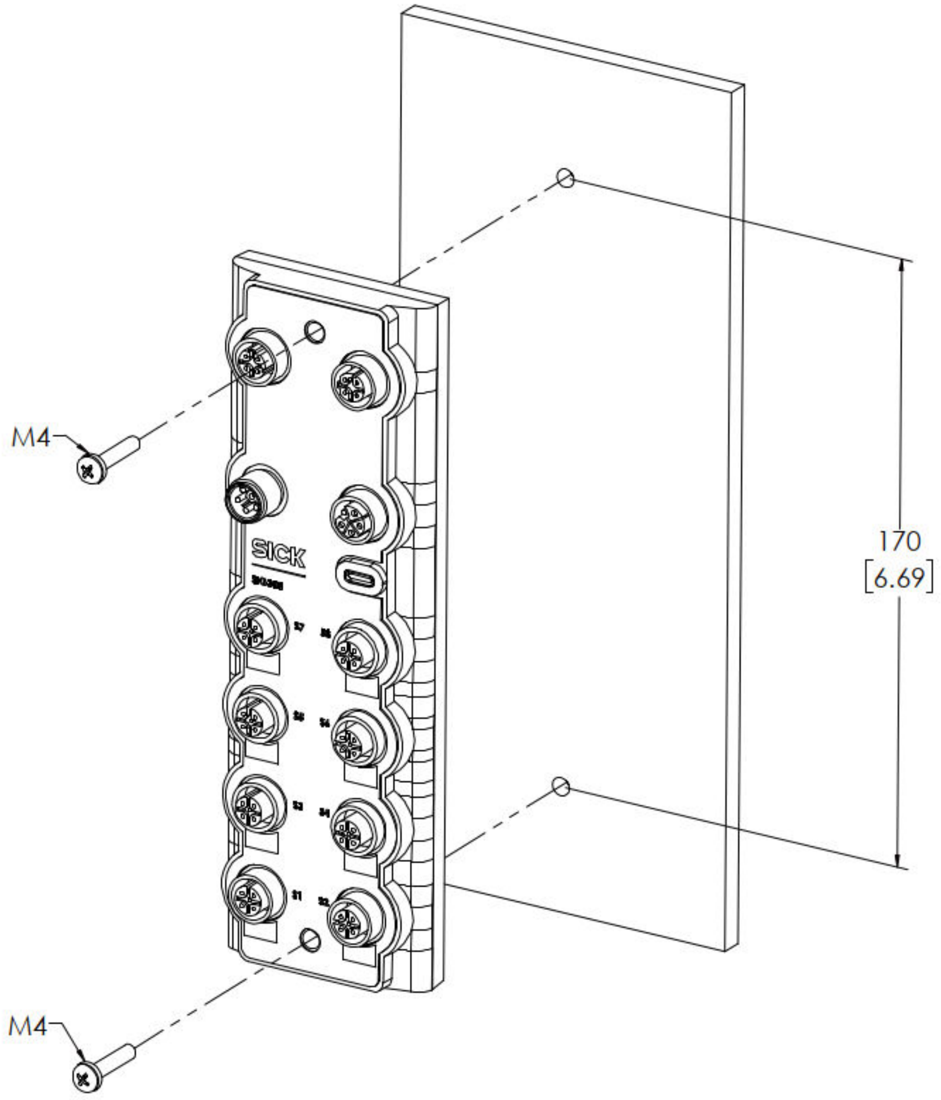


Figure 3: Mounting the module

Mounting

1. Position M4 screw in the upper mounting opening and tighten lightly.
2. Align housing.
3. Position another M4 screw in the lower mounting opening and tighten lightly.
4. Tighten both screws with a max. tightening torque of 1 Nm.

5.3 Functional earth

SIG300 does not require a functional earth connection.

6 Electrical installation

6.1 Electrical connection

The SIG300 is used in electrical installations. When working on the module or the plant, the safety rules of electrical engineering must be observed.

The following information must be observed, depending on the connection type:

- Power and IO-Link cables must be connected in a voltage-free state (U_V = 0 V).
- The connection of the network and IO-Link cable of the SIG300 must be voltage-free (U_S = 0 V; U_A = 0 V). Only apply voltage/switch on the voltage supply (U_S > 0 V; U_A > 0 V) once all electrical connections have been established.
- All unused male and female connectors must be sealed with sealing caps to ensure enclosure rating IP67.
- If possible, sensor/bus (U_S) and actuator power (U_A) supplies should be drawn from different sources.
- In case of separate actuator and sensor supply, always switch on the sensor voltage first and then the actuator voltage to ensure error-free function of the digital inputs and outputs.
- An incorrect supply voltage may result in damage to the device.
- Cables and/or modules damaged by short-circuits can overheat and cause fires. Provide sensible current monitoring or fuse. The fuse protection must be designed to protect the supply cable.
- During operation of the module, the device surface may heat up. If necessary, wear suitable thermal gloves.
- Only install cables and accessories that comply with the requirements and regulations for safety, electromagnetic compatibility and, if applicable, telecommunications terminal equipment and the specification requirements.
- Observe the derating when using the product. The ambient temperature and the current have an influence on the heating of the product.

WARNING
Only operate the product with DC 24 V PELV (Protective Extra-Low Voltage) or SELV (Safety Extra-Low Voltage) voltage sources.

There is a risk of electric shock if this is not observed.

Only use a power supply unit that allows max. 60 V DC or 25 V AC in the event of a fault.

6.2 Pin assignment

NOTE
You will find a large selection of connection cables at www.sick.com

Explanation of the connection diagrams

- DI = Digital input
- DO = Digital output
- FE = Functional earth
- n. c. = Not connected
- Rx+ = Receiver +
- Rx- = Receiver -
- Tx+ = Sender +
- Tx- = Sender –
- U_S = Sensor voltage
- U_A = Actuator voltage

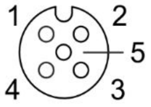
IO-Link ports

Tightening torque = 0.6 Nm

The length of cable of the sensor and actuator lines is generally limited to 30 m. If an IO-Link connection is active, the length of cable is limited to max. 20 m.

Table 8: IO-Link ports (S1 – S8): M12 female contact, A-coded, port class A1 / A2 / B

PIN	IO-Link port class A1 – S1 … S4	IO-Link port class A2 – S5-S6	IO-Link port class B[1] – S7-S8
Pin 1	L+ (U_s+)	L+ (U_s+)	L+ (U_s+)
Pin 2	DI/DO	DI/DO	U_A+
Pin 3	L- (U_s-)	L- (U_s-)	L- (U_s-)
Pin 4	IO-Link/DI/DO	IO-Link/DI/DO	IO-Link/DI/DO
Pin 5	n.c.	DI/DO	U_A-



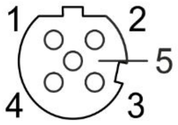
[1] Ports of class B can supply more power to connected devices due to their second power supply. Pins 1, 3, and 4 are used for standard IO-Link communication, with pins 1 and 3 serving as the power supply. Pins 2 and 5 are galvanically isolated from the others and powered by the actuator power supply. This additional power supply allows ports of class B to support higher current loads for connected devices. To maintain galvanic isolation and minimize electrical noise, the isolated pins must be powered by a separate power supply.
Ports of class B are specifically designed for connecting actuators such as valves or motors.

Ethernet ports

Tightening torque = 0.6 Nm

Table 9: Ethernet ports (P1 – P2): M12 female contact, D-coded

PIN	Description
Pin 1	Tx +
Pin 2	Rx +
Pin 3	Tx -
Pin 4	Rx -
Pin 5	n. c.



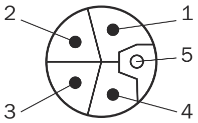
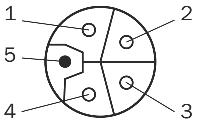
Supply ports

For electrical data, see "General technical data", page 104

Tightening torque = 0.6 Nm

Table 10: Power ports (PWR1 – PWR2): M12 male connector/female contact, L-coded

PIN	Description
Pin 1	U_S Sensor power +. Powers the module and devices connected to S1-S8 (pin 1).
Pin 2	U_A Return, Actuator power -

PIN	Description
Pin 3	U_S Return, Sensor power -
Pin 4	U_A Actuator power + Additional power for IO-Link Class B devices connected to S7 ... S8 (pin 2).
Pin 5	FE (not used)
 *Figure 4: PWR1*	 *Figure 5: PWR2*

6.3 Supply concept

NOTICE
Product damage and/or damage to other connected products can occur if the maximum permissible current is exceeded. Follow the current derating specified in the electrical data table.

Module supply

The module is powered by two supply lines that are galvanically isolated. They are fed in via port PWR1 and fed out via port PWR2 for powering subsequent modules.

Sensor supply

The devices connected to the module are powered via IO-Link ports S1 ... S8. When designing the supply, the requirements of the connected sensors and actuators must be considered.

6.4 Short-circuit and overload-protection on IO-Link ports

The SIG300 IO-Link ports S1 ... S8 support short-circuit and overload protection. This protection is implemented on the power supply in the ports (Pin 1 and Pin 3) and is self-restoring, meaning the ports will automatically return to normal operation once the short-circuit condition is resolved. When a short circuit occurs, the Web Interface of the SIG300 will display the event in the event log, and it will update once the short circuit is resolved. Additionally, information about the short circuit is provided via REST (see the OpenAPI documentation of the device for a more detailed description) and via the available fieldbus, if the SIG300 type in use supports fieldbus.

The short-circuit function is paired between ports 1 and 2, ports 3 and 4, ports 5 and 6, and ports 7 and 8. This means the short-circuit event messages via the Web Interface, REST, and Fieldbus will not differentiate between the paired ports, but will only indicate that an event occurred or was resolved, e. g., on Port 1/2.

7 Commissioning

When the supply voltage is switched on for the first time, the SIG300 starts with the factory settings.

To enable parameterization, the module must be configured to suit the network environment. For this purpose, various preparatory measures must be taken before parameterization can be started.

7.1 IP address

Ethernet

The module requires an IP address so that it can be addressed via the Ethernet network.

ℹ **NOTE**
When delivered, the SIG300 REST has the below default IP addresses:

P1/P2 port: 0.0.0.0 DHCP

PROFINET device type when delivered is "**SIG**".

To change the IP adress: see "Connection options", page 59

The SIG300 supports the following methods for IP address assignment:

1 DHCP
2 Static

USB

USB interface: 169.254.0.1

DHCP

The SIG300 IO-Link Master supports the Dynamic Host Configuration Protocol for assigning IP addresses.

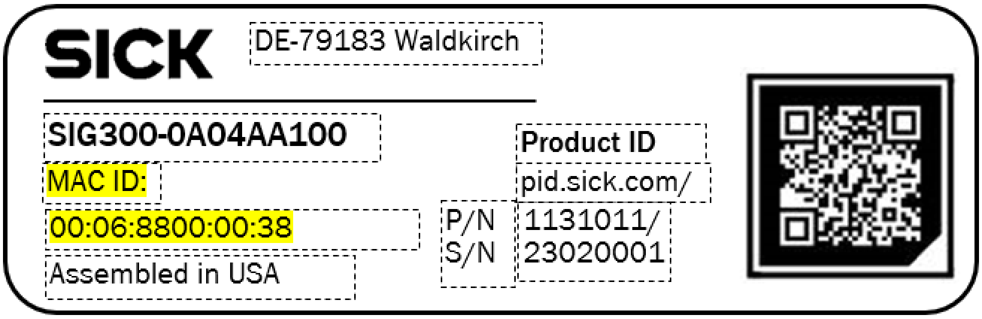
DHCP is deactivated by default. To activate DHCP, open the web browser and change the address mode from static to DHCP in “Configuration/Connection options/Addressing mode”. As soon as DHCP is activated, the IO-Link Master attempts to obtain an address from a DHCP server. When a new IP address is assigned by a DHCP server, the module immediately switches to the new IP address.

Static

In the Ethernet configuration area of the web interface, the default IP address can also be changed statically.

7.2 MAC address

Each device has a uniquely assigned MAC address that cannot be changed by the user. The assigned MAC address is printed on the label.



7.3 Data security

Proper project planning is an important prerequisite for ensuring the confidentiality, availability and integrity of data.

Products are intended for use in local networks. Observe the following notes when using products in your plant:

- Do not connect control components and control networks to an open network such as the Internet or an office network.
- Protect the control components and control networks with the use of a firewall.
- Close all services not required by your application to reduce the risk of cyber attacks and thus increase cyber security.
  (see "Cybersecurity", page 66)
- Restrict physical and electronic access to all automation components to an authorized group of persons.
- To reduce the risk of unauthorized persons gaining access to your system, be sure to change the default passwords and IP addresses before initial commissioning.

7.3.1 Gateway Access Rights

Access to the gateway can be realized:

- Via web UI (Ethernet / USB)
- Via REST interface (Ethernet)
- Via MQTT interface (Ethernet)

The access via web UI, REST and MQTT is controlled by an overriding access rights setting.

For each sensor port, the permission for the REST / MQTT interface and for the Logic Editor can be set separately.

See the gateway section for the REST path to read/write the gateway access rights.

7.3.2 Port Access Rights

SIG300 manages each IO-Link port's access rights and allows each so called “client” to have different access rights.

On the SIG300, there are the following clients:

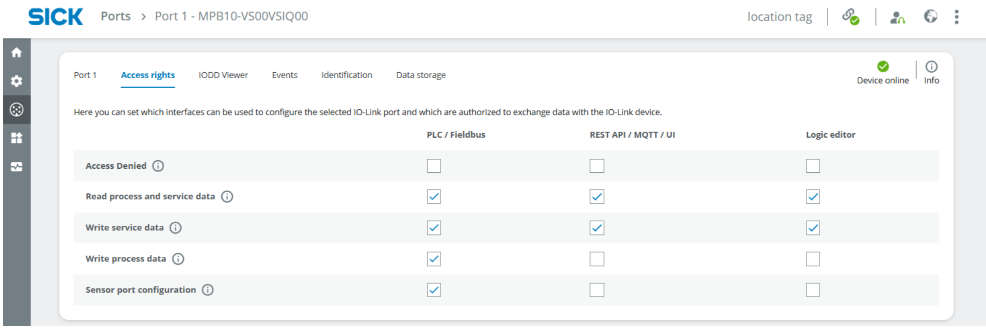
- REST client (includes webserver and MQTT)
- “Logic Editor” client (see "Logic Editor ", page 70).
- Fieldbus

The SIG300 user can define the access rights for each IO-Link port separatly.

The following rules apply when selecting the user access rights for a specific port.

Number	Effect
1	Only one client can control the device process data outputs.
2	Only one client can configure a port.
3	More than one client can write IO-Link service data.
4	More than one client can monitor IO-Link functionality (read service and process data).
5	It is possible to prevent a client from device access (access denied).

The user interface provides the below options related to the IO-Link user roles.



Here you can set which interfaces can be used to configure the selected IO-Link port and which are authorized to exchange data with the IO-Link device.

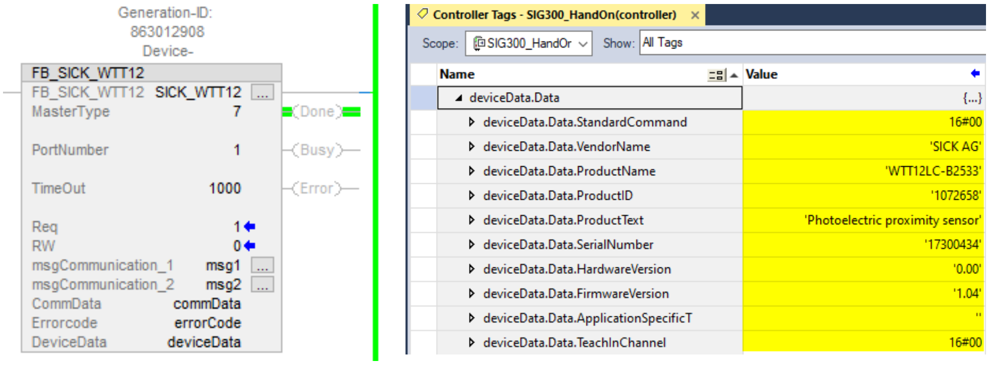
	PLC / Fieldbus	REST API / MQTT / UI	Logic editor
Access Denied ⓘ	□	□	□
Read process and service data ⓘ	☑	☑	☑
Write service data ⓘ	☑	☑	☑
Write process data ⓘ	☑	□	□
Sensor port configuration ⓘ	☑	□	□

8 Operation

8.1 SICK Function Block Factory

With the digital software service **Function Block Factory**, SICK provides a unique opportunity to generate specific PLC function modules.

The Function Block Factory can be used to generate function blocks / AOIs individually for any IO-Link device (manufacturer independent). These function blocks are fully tested and documented. They make PLC programming much easier and faster and help to avoid errors. This can save time and money in the production process.



Generation-ID:
863012908
Device-

FB_SICK_WTT12	
FB_SICK_WTT12	SICK_WTT12
MasterType	7
PortNumber	1
TimeOut	1000
Req	1
RW	0
msgCommunication_1	msg1
msgCommunication_2	msg2
CommData	commData
Errorcode	errorCode
DeviceData	deviceData

Controller Tags - SIG300_HandOn(controller)

Scope: SIG300_HandOr Show: All Tags

Name	Value
deviceData.Data	{...}
deviceData.Data.StandardCommand	16#00
deviceData.Data.VendorName	'SICK AG'
deviceData.Data.ProductName	'WTT12LC-B2533'
deviceData.Data.ProductID	'1072658'
deviceData.Data.ProductText	'Photoelectric proximity sensor'
deviceData.Data.SerialNumber	'17300434'
deviceData.Data.HardwareVersion	'0.00'
deviceData.Data.FirmwareVersion	'1.04'
deviceData.Data.ApplicationSpecificT	''
deviceData.Data.TeachInChannel	16#00

Further information on SICK's Function Block Factory can be found on the following website.

https://fbf.cloud.sick.com

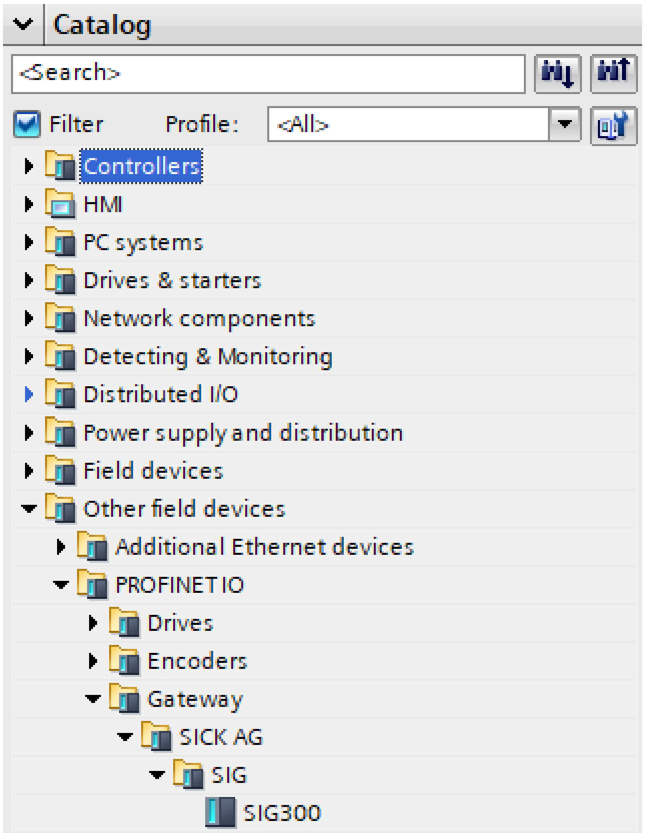
8.2 PROFINET

The SIG300 can exchange process data and parameters via PROFINET. For this purpose, the IO-Link Master must be connected to a suitable programmable logic controller (PLC).

These instructions describe an example integration of the SIG300 into a PROFINET network. The system integration and parameterization described in the following shows a good example of how the SIG300 is used together with the TIA Portal V19 project planning software from Siemens. If you use other controllers and project planning software, observe the corresponding documentation. SICK does not assume any liability for the correctness and completeness of the contents.

8.2.1 Install GSDML file

The device data required for project planning is stored in GSDML (“Generic Station Description Markup Language”) files. The GSDML file makes the possible data module available with input or output of different data widths. Download the corresponding GSDML file under www.sick.com and save the GSDML file at a place where you have access with the control software. Open your application program and import the GSDML file via “Manage general station description files (GSD)” to make the SIG300 available in the hardware catalog of the TIA-Portal.



Catalog

<Search>

☑ Filter Profile: <All>

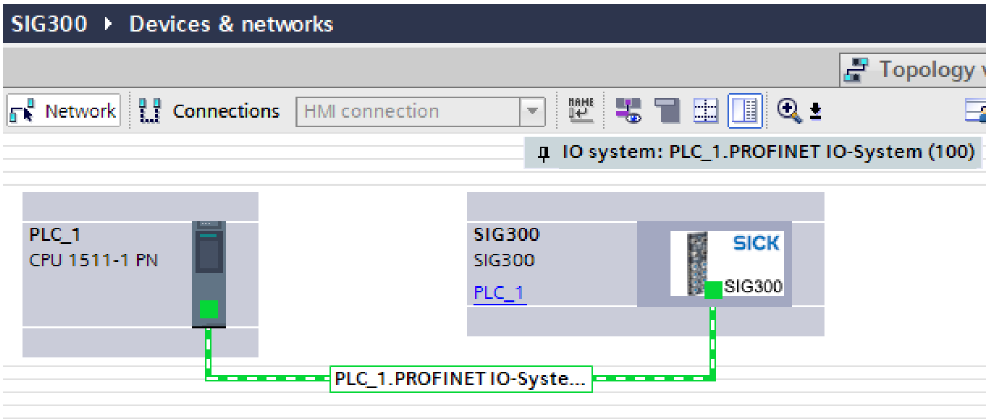
- Controllers
- HMI
- PC systems
- Drives & starters
- Network components
- Detecting & Monitoring
- Distributed I/O
- Power supply and distribution
- Field devices
- Other field devices
  - Additional Ethernet devices
  - PROFINET IO
    - Drives
    - Encoders
    - Gateway
      - SICK AG
        - SIG
          - SIG300

The SIG300 can be found in the hardware catalog under “**Other field devices > PROFINET IO > Gateway > SICK AG > SIG > SIG300**”.

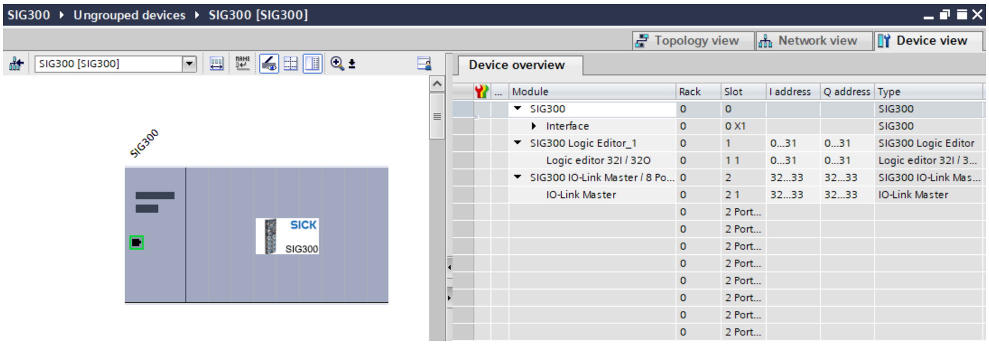
8.2.2 Adding device to the PROFINET network

To include the IO-Link Master into the project, select the corresponding SIG300 entry in the hardware catalog and drag and drop it to a free position in the “Devices & networks” topology view.

Once the SIG300 is inserted in the network view, the network of the module must be assigned to a PROFINET controller (PLC). Click “Not assigned” on the device tile and select the PLC to be assigned as a PROFINET controller.


SIG300 ▸ Devices & networks
Topology v
Network
Connections
HMI connection
IO system: PLC_1.PROFINET IO-System (100)
PLC_1
CPU 1511-1 PN
SIG300
SIG300
PLC_1
SICK
SIG300
PLC_1.PROFINET IO-Syste...

Enter the “Device view” with a double click and continue the configuration.



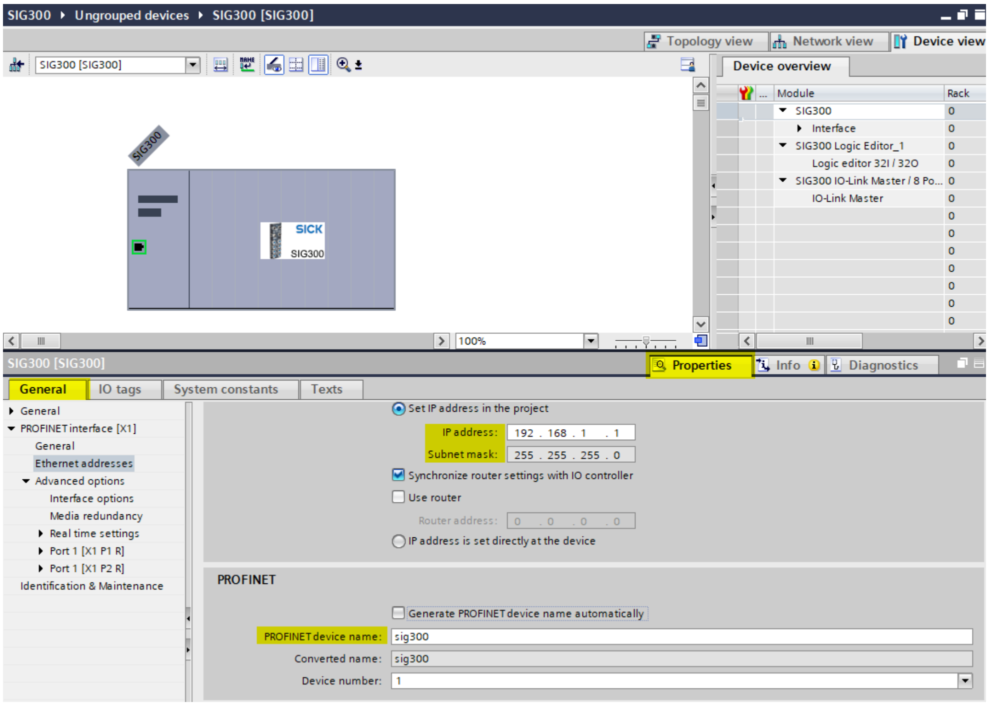
	...	Module	Rack	Slot	I address	Q address	Type
		SIG300	0	0			SIG300
		Interface	0	0 X1			SIG300
		SIG300 Logic Editor_1	0	1	0...31	0...31	SIG300 Logic Editor
		Logic editor 32I / 32O	0	1 1	0...31	0...31	Logic editor 32I / 3...
		SIG300 IO-Link Master / 8 Po...	0	2	32...33	32...33	SIG300 IO-Link Mas...
		IO-Link Master	0	2 1	32...33	32...33	IO-Link Master
			0	2 Port...			
			0	2 Port...			
			0	2 Port...			
			0	2 Port...			
			0	2 Port...			
			0	2 Port...			
			0	2 Port...			
			0	2 Port...			

8.2.3 Configuring and assigning device name and IP address

PROFINET devices are addressed via a unique device name. This can be freely assigned by the user but may only be used once in the network.

8.2.3.1 Configuring PROFINET device name and IP address

To configure the PROFINET device name and the IP address, click on the SIG300 tile in the “Device view”. In the lower part, please select the "Properties" tab and navigate to the "Ethernet addresses" in the navigation tree.



	...	Module	Rack
		SIG300	0
		Interface	0
		SIG300 Logic Editor_1	0
		Logic editor 32I / 32O	0
		SIG300 IO-Link Master / 8 Po...	0
		IO-Link Master	0
			0
			0
			0
			0
			0
			0
			0

● Set IP address in the project

IP address: 192 . 168 . 1 . 1

Subnet mask: 255 . 255 . 255 . 0

☑ Synchronize router settings with IO controller

□ Use router

Router address: 0 . 0 . 0 . 0

○ IP address is set directly at the device

PROFINET

□ Generate PROFINET device name automatically

PROFINET device name: sig300

Converted name: sig300

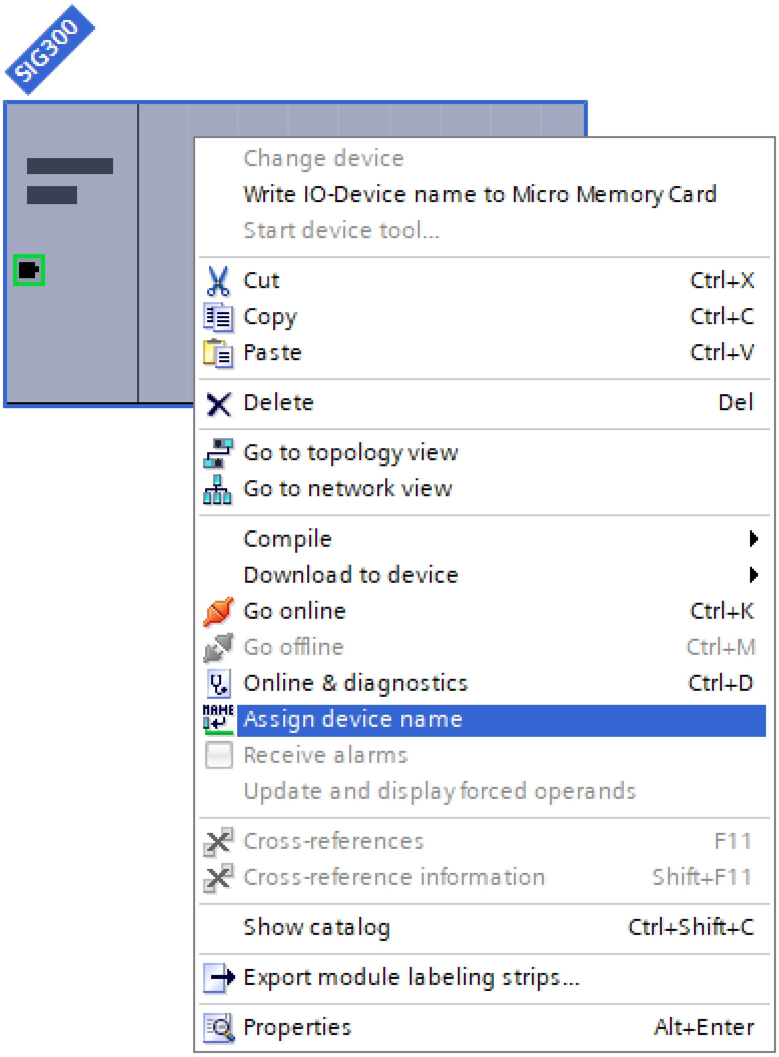
Device number: 1

When assigning the IP, make sure that the device and the controller are in the same ethernet subnet.

In the next step, the device must be assigned the configured PROFINET name.

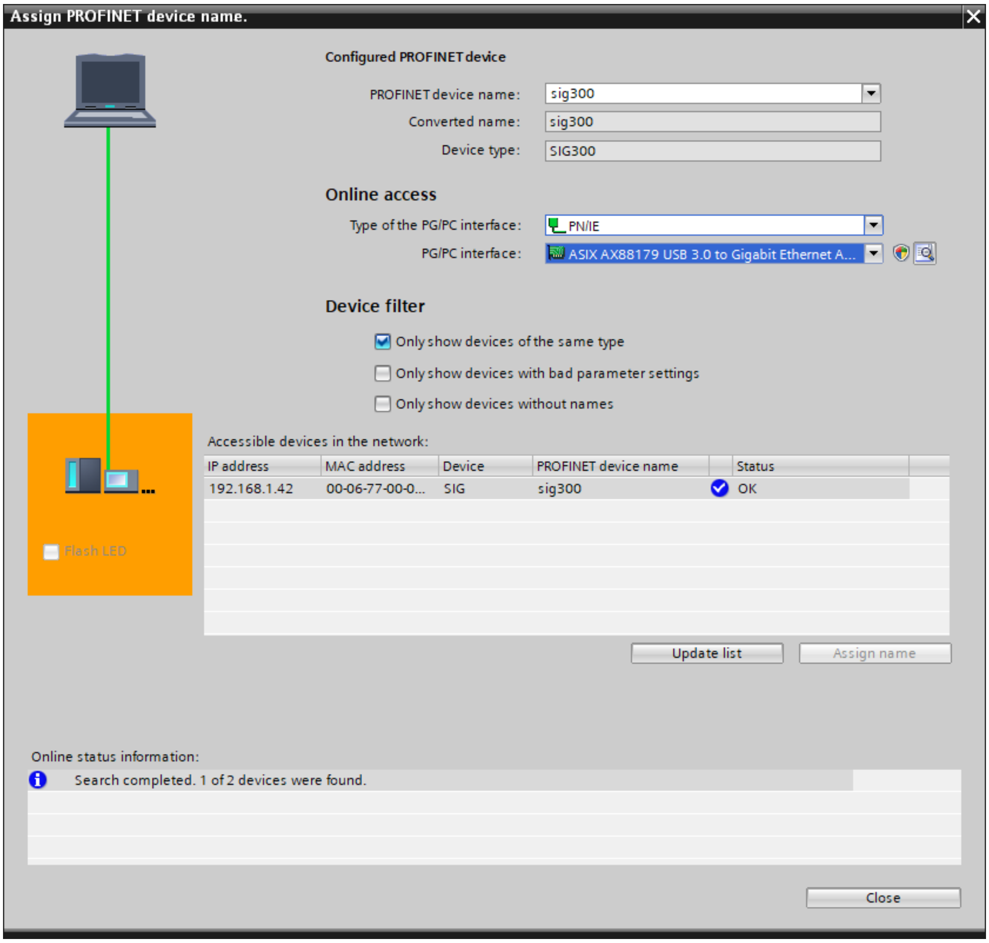
8.2.3.2 Assign device name

After the PROFINET device name has been configured, it must be assigned to the device. Click with the right mouse button on the SIG300 tile and select "Assign device name".



The device name must correspond to the name previously configured under “Properties”.

To make sure that the name is assigned to the correct PROFINET device, it can be identified via the "Flash LED" function or via the MAC address.



Assign PROFINET device name.

Configured PROFINET device

PROFINET device name: sig300

Converted name: sig300

Device type: SIG300

Online access

Type of the PG/PC interface: PN/IE

PG/PC interface: ASIX AX88179 USB 3.0 to Gigabit Ethernet A...

Device filter

☑ Only show devices of the same type

☐ Only show devices with bad parameter settings

☐ Only show devices without names

☐ Flash LED

Accessible devices in the network:

IP address	MAC address	Device	PROFINET device name	Status
192.168.1.42	00-06-77-00-0...	SIG	sig300	✓ OK

Online status information:

Search completed. 1 of 2 devices were found.

8.2.4 PROFINET network configuration

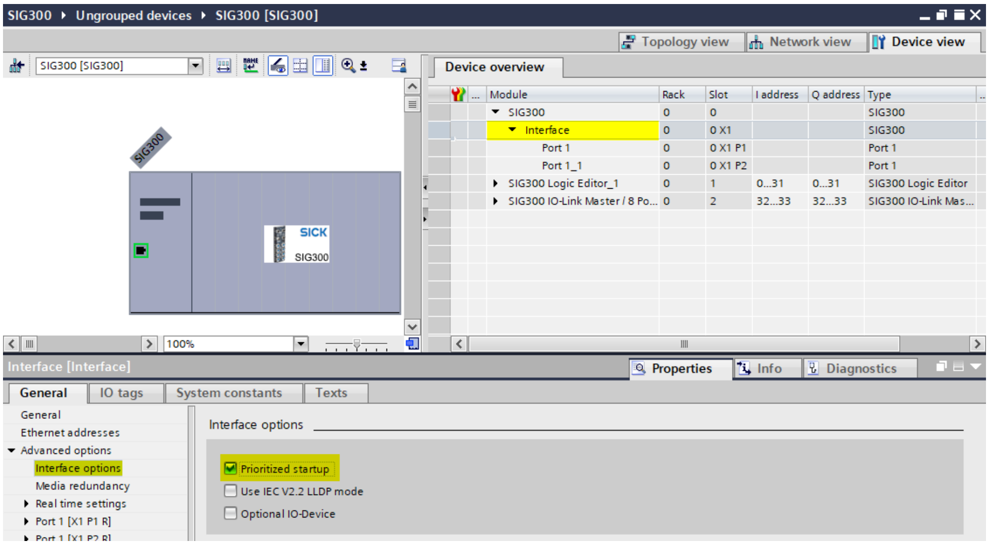
<text GSDML params>

8.2.4.1 Prioritized startup (FSU)

If the prioritized start-up, also called “Fast startup” (FSU), is activated, the modules start up within < 2 s.

To activate the function, several settings must be made in the hardware configuration. The corresponding parameters are set directly in the module.

- Select module (Interface) in the “Device overview”
- Under Advanced options > Interface options, activate “Prioritized startup”.



	...	Module	Rack	Slot	I address	Q address	Type
		▾ SIG300	0	0			SIG300
		▾ Interface	0	0 X1			SIG300
		Port 1	0	0 X1 P1			Port 1
		Port 1_1	0	0 X1 P2			Port 1
		▸ SIG300 Logic Editor_1	0	1	0...31	0...31	SIG300 Logic Editor
		▸ SIG300 IO-Link Master / 8 Po...	0	2	32...33	32...33	SIG300 IO-Link Mas...

- General
- Ethernet addresses
- ▾ Advanced options
  - Interface options
  - Media redundancy
  - ▸ Real time settings
  - ▸ Port 1 [X1 P1 R]
  - ▸ Port 1 [X1 P2 R]

Interface options

☑ Prioritized startup
☐ Use IEC V2.2 LLDP mode
☐ Optional IO-Device

During conventional connection setup, connection parameters are negotiated between the individual PNIO devices, which results in a time delay.

To prevent this delay and to ensure optimal startup in < 2 s, the connection parameters for each port (participating in the FSU) must be permanently parameterized.

- The transmission speed on all connected ports must be permanently set to 100 Mbit.
- This eliminates time-consuming negotiation of the connection parameters during module startup.
- Auto-negotiation must not be activated.
  This eliminates time-consuming negotiation of the connection wire pairs during module start-up.
- Observe port direction: Port 1 IN Port 2 OUT
- Unassigned PNIO ports (e.g. last module in the line topology) do not have to be changed over.

This completes the necessary settings for using FSU.

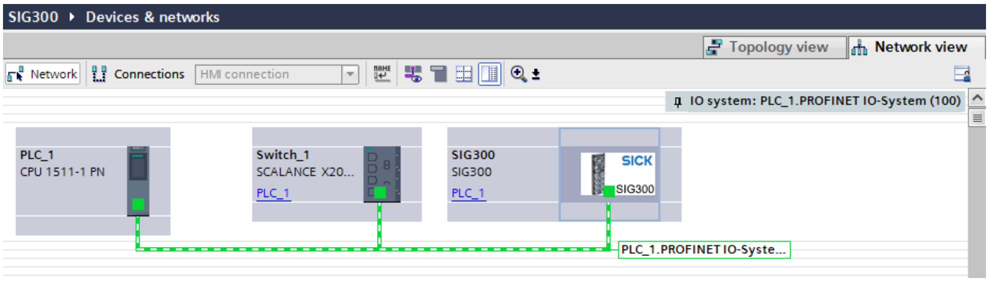
8.2.4.2 Media Redundancy Protocol (MRP) configuration

The SIG300 supports the ring topology with media redundancy, which is enabled by means of the Media Redundancy Protocol (MRP).

Setting up MRP manager

For a ring configuration, 2 ports, a managed switch or a CPU must be set as MRP manager. The remaining participants must be set up as clients.

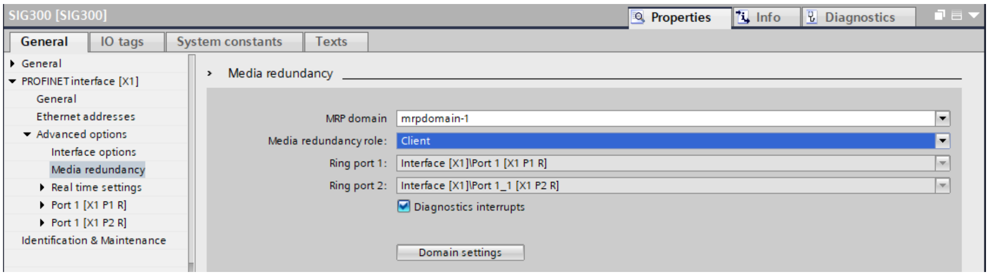
Select the according managed switch and navigate to PROFINET Interface > Advanced options > Media redundancy. Set **Manager** (Auto) role in the media redundancy settings. In addition, check the Diagnostics interruption box.



Setting up MRP client

Select the SIG300 in the module tree and navigate to Advanced options > Media redundancy. For the SIG300, the media redundancy role must be set to “Client”. In addition, the diagnostic interrupts must be activated.

The SIG300 must be in the same MRP domain as the MRP master.



› Media redundancy

MRP domain: mrpdomain-1

Media redundancy role: Client

Ring port 1: Interface [X1]\Port 1 [X1 P1 R]

Ring port 2: Interface [X1]\Port 1_1 [X1 P2 R]

☑ Diagnostics interrupts

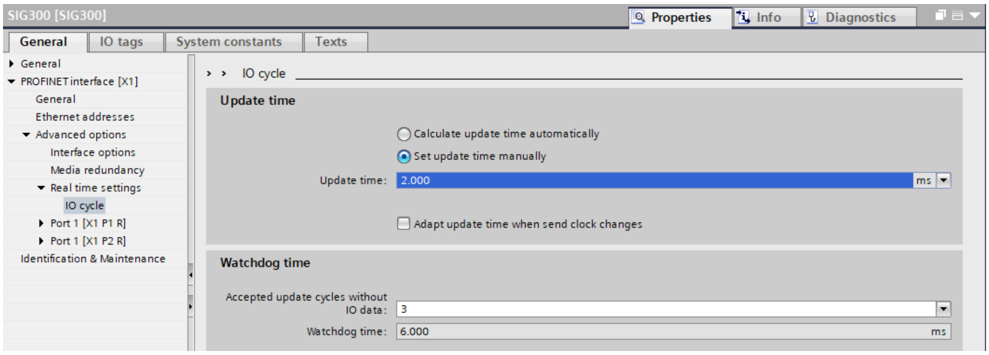
Figure 6: Setting up MRP client

With the ring topology, it is possible to build up a redundant system. That means in normal operation, one side of the ring line is deactivated by the MRP master. If the line is damaged/cut at a point in the ring, the deactivated branch is reactivated and two linear topologies are created.

MRP cycle times

To ensure uninterrupted operation, the response monitoring time should be < 200 ms. This is because the MRP master needs a certain time to activate the second string. If the response monitoring time is smaller than the switchover time of the MRP master, this leads to a communication abort.

This response monitoring time is calculated from the “Update time” and the “Accepted update time without IO data” factor:



› › IO cycle

Update time

○ Calculate update time automatically

● Set update time manually

Update time: 2.000 ms

□ Adapt update time when send clock changes

Watchdog time

Accepted update cycles without IO data: 3

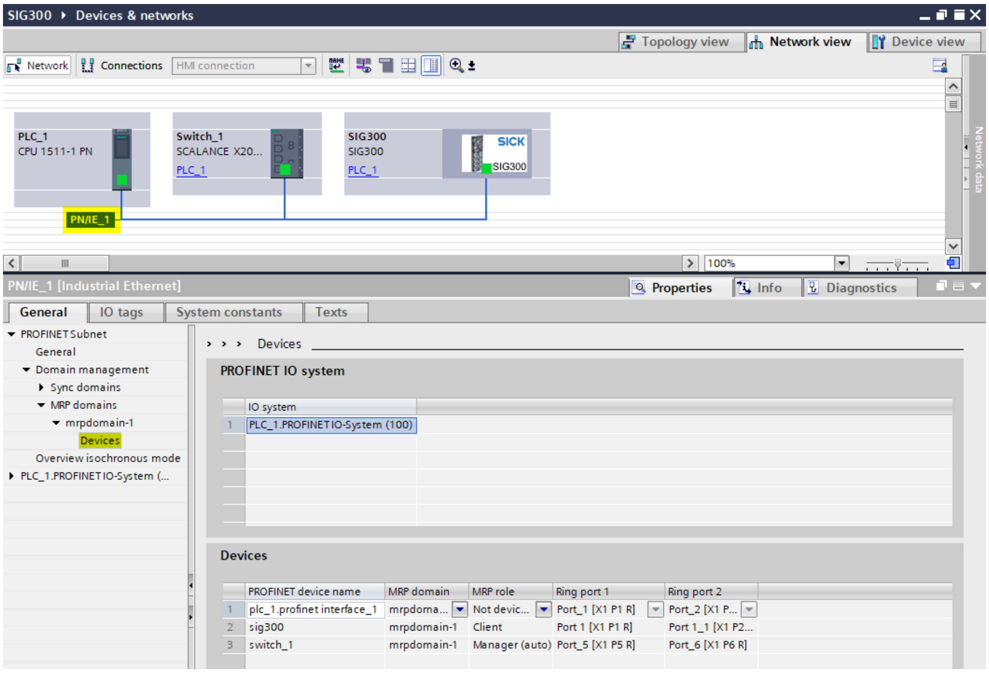
Watchdog time: 6.000 ms

MRP domain

Call up the overview of ring participants via Domain Management > MRP Domains > mrpdomain-1.

Assign the PROFINET Managed Switch to the IO controller and save the created project. Click on the PNIE subnet. The bus “PLC_1.PROFINET IO system (100)” is displayed.

Assign the SIG300 module to the IO controller. All participants of the ring configuration are displayed under Devices.



PROFINET IO system

	IO system
1	PLC_1.PROFINET IO-System (100)

Devices

	PROFINET device name	MRP domain	MRP role	Ring port 1	Ring port 2
1	plc_1.profinet interface_1	mrpdoma...	Not devic...	Port_1 [X1 P1 R]	Port_2 [X1 P...
2	sig300	mrpdomain-1	Client	Port 1 [X1 P1 R]	Port 1_1 [X1 P2...
3	switch_1	mrpdomain-1	Manager (auto)	Port_5 [X1 P5 R]	Port_6 [X1 P6 R]

8.2.5 PROFINET module overview

The device model represents process data and acyclic data in the following slots.

Table 11: PROFINET slots overview

Slot	Sub slot	Designation	Description
0		SIG300	Device access point (DAP): Main module
	X1	Interface	PROFINET interface
	X1 P1	Port 1	Ethernet port P1
	X1 P2	Port 2	Ethernet port P2
1		SIG Logic Editor	Logic Editor
	1	Logic Editor Submodule	To exchange data with the Logic Editor
2		SIG300 IO-Link Master	IO-Link Master ports S1 … S8
	1	IO-Link Master	For data exchange with the digital inputs and outputs (Pin 2 + Pin 5)
	2 … 9	IO-Link Master Submodule	IO-Link process data and digital input data (Pin 4). These submodules provide the client access point required for IO-Link service data access (acyclic data).

8.2.6 Process data (SIG300 Logic Editor)

The plugged-in submodules can be used to define how much data should be exchanged with the Logic Editor via the process data channel.

The process data can be assigned to the SIG300 variables via the Web UI, which can be used in the Logic Editor.

Table 12: Available submodules to exchange data with the Logic Editor

Submodule name	Process data size
Logic editor 0I / 20	0 Byte input / 2 Byte output
Logic editor 0I / 40	0 Byte input / 4 Byte output

Submodule name	Process data size
Logic editor 0I / 80	0 Byte input / 8 Byte output
Logic editor 0I / 160	0 Byte input / 16 Byte output
Logic editor 0I / 320	0 Byte input / 32 Byte output
Logic editor 0I / 640	0 Byte input / 64 Byte output
Logic editor 0I / 1280	0 Byte input / 128 Byte output
Logic editor 2I / 00	2 Byte input / 0 Byte output
Logic editor 4I / 00	4 Byte input / 0 Byte output
Logic editor 8I / 00	8 Byte input / 0 Byte output
Logic editor 16I / 00	16 Byte input / 0 Byte output
Logic editor 32I / 00	32 Byte input / 0 Byte output
Logic editor 64I / 00	64 Byte input / 0 Byte output
Logic editor 128I / 00	128 Byte input / 0 Byte output
Logic editor 2I / 20	2 Byte input / 2 Byte output
Logic editor 4I / 40	4 Byte input / 4 Byte output
Logic editor 8I / 80	8 Byte input / 8 Byte output
Logic editor 16I / 160	16 Byte input / 16 Byte output
Logic editor 32I / 320	32 Byte input / 32 Byte output
Logic editor 64I / 640	64 Byte input / 64 Byte output
Logic editor 128I / 1280	128 Byte input / 128 Byte output

NOTE

If the Logic Editor functionality is not used, it is possible to remove the entire module "SIG300 Logic Editor" to save PROFINET bandwidth.

8.2.7 Process data (SIG300 Logic Master / 8 Port)

The SIG300 has 8 sensor / actuator ports that can be assigned to specific IO-Link devices. For each IO-Link port (S1 - S8), it must be defined which device type is connected to the port.

- Deactivated
- IO-Link device
- Digital input
- Digital output

8.2.7.1 Submodule (IO-Link Master)

The IO-Link master module provides the digital signals from pin 2 and pin 5. Access to the digital signals depends on the behavior set in the respective port configuration.

Table 13: Digital input (Pin 2 / Pin 5)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Reserved	Reserved	S6DI2	S5DI2	S4DI2	S3DI2	S2DI2	S1DI2
1	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	S6DI5	S5DI5

Table 14: Digital output (Pin 2 / Pin 5)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	S8DO2	S7DO2	S6DO2	S5DO2	S4DO2	S3DO2	S2DO2	S1DO2
1	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	S6DO5	S5DO5

8.2.7.2 Submodule (Deactivated)

If a port of the SIG300 is not used, the corresponding submodule can be assigned a "deactivated" module.

8.2.7.3 Submodule (IO-Link xI / yO + PQI)

The IO-Link specification defines three types of data that are exchanged between the IO-Link Master and IO-Link device:

- Cyclic process data (access via corresponding submodule address).
- Acyclic data in the form of device data, ISDUs.
- Acyclic data as events.

The process data and its status information (port qualifiers) are transmitted cyclically after communication has been established. For each IO-Link port, the transmission time can be parameterized individually as the IO-Link cycle time.

The process data of the connected device can be 0 to 32 bytes in size (input and output) and are transmitted via the set address ranges.

Port S1...S8 must be configured according to the process data length of the connected device. The information on the required process data lengths can be found in the documentation of the connected IO-Link device. Contact the manufacturer of the IO-Link device to do so.

If no suitable configuration module is available for the device, the next largest data length must be selected.

Table 15: Available submodules to exchange IO-Link data

Submodule name	Process data size
IO-Link 0I / 1O + PQI	0 Byte input / 1 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 0I / 2O + PQI	0 Byte input / 2 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 0I / 4O + PQI	0 Byte input / 4 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 0I / 8O + PQI	0 Byte input / 8 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 0I / 16O + PQI	0 Byte input / 16 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 0I / 32O + PQI	0 Byte input / 32 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 1I / 0O + PQI	1 Byte input / 0 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 2I / 0O + PQI	2 Byte input / 0 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 4I / 0O + PQI	4 Byte input / 0 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 8I / 0O + PQI	8 Byte input / 0 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 16I / 0O + PQI	16 Byte input / 0 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 32I / 0O + PQI	32 Byte input / 0 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 1I / 1O + PQI	1 Byte input / 1 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 2I / 2O + PQI	2 Byte input / 2 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 4I / 4O + PQI	4 Byte input / 4 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 8I / 8O + PQI	8 Byte input / 8 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 16I / 16O + PQI	16 Byte input / 16 Byte output (IO-Link) + 1 Byte input (PQI)
IO-Link 32I / 32O + PQI	32 Byte input / 32 Byte output (IO-Link) + 1 Byte input (PQI)

PQI

The Port Qualifier Information (PQI) provides information on the status and process data of the IO-Link port and is transferred together with the input data of the IO-Link device.

Table 16: Port Qualifier Information (PQI)

Bit	Description
Bit 0	Reserved, always FALSE
Bit 1	Reserved, always FALSE
Bit 2	New parameter FALSE = No change of the device parameter detected TRUE = Change of device parameter detected: Master has performed a data memory upload, and a new IOL-D backup object (0xB904) is available.
Bit 3	Substitute device detected FASLE = No substitute device detected (identical serial number) TRUE = Substitute device detected (different serial number)
Bit 4	Port activation FALSE = Port deactivated via port function TRUE = Port activated
Bit 5	Device communication FALSE = No device available TRUE = Device detected and in the PREOPERATE or OPERATE state
Bit 6	Displaying a port / device error FALSE = No error / no warning TRUE = Error / warning for device or port
Bit 7	Validity of device process data (PQ) FALSE = Invalid IO process data from the device TRUE = Valid IO process data from the device

8.2.7.4 Submodule (Digital Input / Output)

Use this module if the connected device is to be operated as a standard input or a standard output device.

Table 17: Digital input / output byte

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
false	false	false	false	false	false	false	C/Q (Pin 4) signal input / output

8.2.8 Module parameters

The module parameters can be found in the properties of the respective modules.

8.2.8.1 Parameters (IO-Link Master module)

Table 18: IO-Link module parameters

Parameter name	Information	
Input values with module failure	0: Input value 0	Behavior of input process data when the module has a failure.
	1: Keep last value	

8.2.8.2 Parameters (IO-Link module)

Table 19: IO-Link module parameters

Parameter name	Information
Port configuration	

Parameter name	Information	
Enable Port diagnosis	0: Disable	IO-Link device events of the type “Warning” and “Errors” are transferred to the PROFINET diagnostics.
	1: Enable	
Enable Process Alarm (device notification)	0: Disable	Activision of IO-Link device events of the “Notification” type as process alarm.
	1: Enable	
Configuration Source	0: PDCT	Parameterization with IO-Link tool: Two-stage commissioning is used at this IO-Link port. Port and device parameterization can be performed using tools that support the standardized PDCT interface (Port and Device Configuration Tool).
	1: PNIO	
Enable Input fraction	0: Disable	Activates monitoring of the process data length of the submodule against process data length of the connected device.
	1: Enable	
Enable Pull/Plug	0: Disable	Connection termination / Connection setup on the device is signaled via the PROFINET diagnostics.
	1: Enable	Connection termination / Connection setup on the device is signaled via the PROFINET “Hardware component removed” message.
Port Mode	1: IOL Manual	IO-Link port parameterization active: Single-stage commissioning is used at this IO-Link port. The explicit port parameterization for inspection level, port cycle time, manufacturer ID and device ID is done in the PLC via the PROFINET engineering tool.
	2: IOL Autostart	Automatic IO-Link parameterization: No explicit port parameterization is used at this IO-Link port. Basic assignment such as inspection level, port cycle time, vendor ID, and parameters are determined by the connected IO-Link device.

Parameter name	Information	
Validation / Backup (is only considered in port mode IOL Manual)	0: No device check	No device inspection
	1: Type compatible device (V1.0)	Type-compatible device: An inspection concerning the revision ID (V1.0), device ID and vendor ID is done.
	2: Type compatible device (V1.1)	Type-compatible device: An inspection concerning the revision ID (V1.1), device ID and vendor ID is done.
	3: Type compatible device with Backup + Restore	Type-compatible device with backup and restore: An inspection concerning the revision ID (V1.1), device ID and vendor ID is done. Data storage (reading and writing) is permitted.
	4: Type compatible device with Restore	Type-compatible device with restore: An inspection concerning the revision ID (V1.1), device ID and vendor ID is done. Data storage (writing) is allowed.
I/Q Behavior (Pin 2)	0: Deactivated	Configuration of pin 2 behavior depending on the port class.
	1: Digital Input (only class A)	
	2: Digital Output (only class A)	
	5: Auxiliary power In (only Class B)	
Port cycle time (is only considered in port mode IOL Manual)	0: as fast as possible	IO-Link port cycle time in milliseconds.
	1: 2ms	
	2: 4ms	
	3: 8ms	
	4: 16ms	
	5: 32ms	
	6: 64ms	
	7: 128ms	
Vendor ID (is only considered in port mode IOL Manual)	Expected manufacturer ID / vendor ID of the IO-Link device connected to the IO-Link port (unsigned integer 16). Example: 26 (SICK AG)	
Device ID (is only considered in port mode IOL Manual)	Expected device ID of the IO-Link device connected to the IO-Link port (unsigned integer 16). Example: 8389010 (SIG100)	
Digital configuration		
I/Q Behavior (Pin 5)	0: Deactivated	Configuration of pin 5 behavior depending on the port class (A2).
	1: Digital Input (only class A2)	
	2: Digital Output (only class A2)	
I/Q Output mode (Pin 5)	0: NPN	Operating mode of the IQ Output pin 5 if it is configured as digital output.
	1: PNP	
	2: Push Pull	

Parameter name	Information	
I/Q Output mode (Pin 2)	0: NPN	Operating mode of the IQ Output pin 2 if it is configured as digital output.
	1: PNP	
	2: Push Pull	

8.2.8.3 Parameters (Digital Input module)

Table 20: Digital input module parameters

Parameter name	Information	
Port configuration		
Enable Port diagnosis	0: Disable	IO-Link device events of the type “Warning” and “Errors” are transferred to the PROFINET diagnostics.
	1: Enable	
Configuration Source	0: PDCT	Parameterization with IO-Link tool: Two-stage commissioning is used at this IO-Link port. Port and device parameterization can be performed using tools that support the standardized PDCT interface (Port and Device Configuration Tool).
	1: PNIO	
I/Q Behavior (Pin 2)	0: Deactivated	Configuration of pin 2 behavior depending on the port class.
	1: Digital Input (only class A)	
	2: Digital Output (only class A)	
	5: Auxiliary power In (only Class B)	
Digital configuration		
I/Q Behavior (Pin 5)	0: Deactivated	Configuration of pin 5 behavior depending on the port class (A2).
	1: Digital Input (only class A2)	
	2: Digital Output (only class A2)	
I/Q Output mode (Pin 5)	0: NPN	Operating mode of the IQ Output pin 5 if it is configured as digital output.
	1: PNP	
	2: Push Pull	
I/Q Output mode (Pin 2)	0: NPN	Operating mode of the IQ Output pin 2 if it is configured as digital output.
	1: PNP	
	2: Push Pull	

8.2.8.4 Parameters (Digital Output module)

Table 21: Digital output module parameters

Parameter name	Information	
Port configuration		
Enable Port diagnosis	0: Disable	IO-Link device events of the type “Warning” and “Errors” are transferred to the PROFINET diagnostics.
	1: Enable	

Parameter name	Information	
Configuration Source	0: PDCT	Parameterization with IO-Link tool: Two-stage commissioning is used at this IO-Link port. Port and device parameterization can be performed using tools that support the standardized PDCT interface (Port and Device Configuration Tool).
	1: PNIO	
I/Q Behavior (Pin 2)	0: Deactivated	Configuration of pin 2 behavior depending on the port class.
	1: Digital Input (only class A)	
	2: Digital Output (only class A)	
	5: Auxiliary power In (only Class B)	
Digital configuration		
I/Q Behavior (Pin 5)	0: Deactivated	Configuration of pin 5 behavior depending on the port class (A2).
	1: Digital Input (only class A2)	
	2: Digital Output (only class A2)	
I/Q Output mode (Pin 5)	0: NPN	Operating mode of the IQ Output pin 5 if it is configured as digital output.
	1: PNP	
	2: Push Pull	
I/Q Output mode (Pin 2)	0: NPN	Operating mode of the IQ Output pin 2 if it is configured as digital output.
	1: PNP	
	2: Push Pull	
C/Q Output mode (Pin 4)	0: NPN	Operating mode of the CQ Output pin 4 if it is configured as digital output.
	1: PNP	
	2: Push Pull	

8.2.8.5 Parameters (Deactivated module)

Table 22: Deactivated module parameters

Parameter name	Information	
Port configuration		
Enable Port diagnosis	0: Disable	IO-Link device events of the type “Warning” and “Errors” are transferred to the PROFINET diagnostics.
	1: Enable	

8.2.9 PROFINET diagnostics

The SIG300 sends diagnostic information in the form of alarms according to PROFINET specification V2.3. Diagnostics are reported to the PLC in the form of “coming” and “going” alarms. If an alarm is only present for a short time, it is advantageous if a diagnostic buffer is available in the PLC. With this diagnostic buffer, the alarm details can be evaluated later. If the PLC does not provide a diagnostic buffer, it should be created as user software.

Please consult the documentation of the device for manufacturer-specific IO-Link events.

The IO-Link events are mapped to the fieldbus as follows:

- Warning > PROFINET diagnostics message
- Error > PROFINET diagnostics message
- Notification > PROFINET process alarm

8.2.10 Factory reset via PROFINET

The reset to factory settings function can be executed with the following steps:

- Click on “Connect online” in the TIA-Portal.
- After the connection, select the relevant module.
- Open the SIG300 in the project navigation. Double click on “Online & diagnostics”
- In the Functions tab, click on “Reset to factory settings”. Another window opens here.
- If necessary, activate the “Retain I&M data” selection if it should not be deleted.
- Press the “Reset” button. The module is reset to the factory settings.

After the factory reset, besides the PROFINET device name, the IP address and the SNMP parameters are reset. The following default values are then stored in the module:

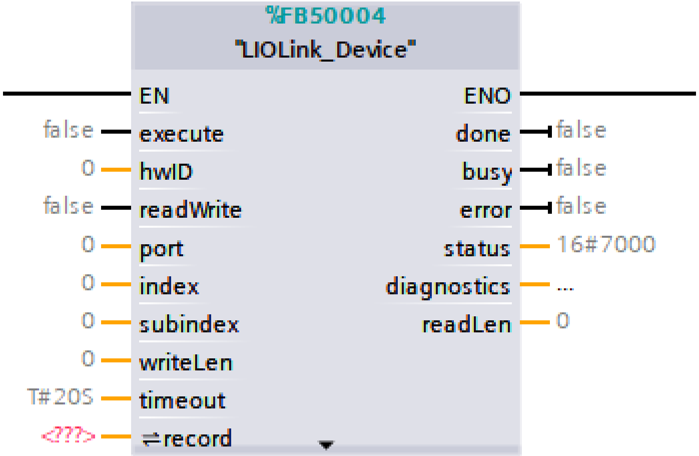
Table 23: Standard values after factory reset

Settings	Value
IP address	0.0.0.0
Subnet mask	0.0.0.0
Router address	0.0.0.0
PROFINET device name	Non-existent

8.2.11 Service data access via IOL_Call

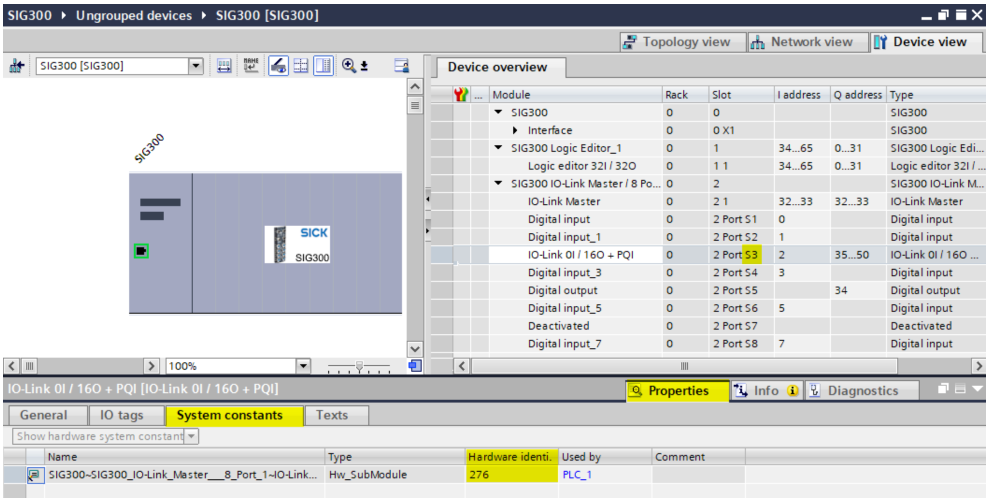
The IOL_Call is an add-on to the PROFINET protocol that can be used to access IO-Link On-Request data (ISDU parameters). It is specified in the document “IO-Link Integration - Edition 2, Guideline for PROFINET” Version 1.0 – June 2017 (order no. 2.832) of the PROFIBUS user organization (PNO).

The PLC manufacturers offer various generic function blocks for reading data from the IO-Link Master as easily as possible. For this use-case, Siemens provides the “LIOLink_Device” function block. This function block used the index 46080 (0xB400) to send ISDU read/write commands via the acyclic PROFINET channel.



NOTE
Please use the hardware identifier (hwID) of the PROFINET Submodule, which represents the corresponding IO-Link port.

In the TIA portal, the hardware identifier can be read here (example for port S3).



Device overview

Module	Rack	Slot	I address	Q address	Type
SIG300	0	0			SIG300
Interface	0	0 X1			SIG300
SIG300 Logic Editor_1	0	1	34...65	0...31	SIG300 Logic Edi...
Logic editor 32I / 32O	0	1 1	34...65	0...31	Logic editor 32I / ...
SIG300 IO-Link Master / 8 Po...	0	2			SIG300 IO-Link M...
IO-Link Master	0	2 1	32...33	32...33	IO-Link Master
Digital input	0	2 Port S1	0		Digital input
Digital input_1	0	2 Port S2	1		Digital input
IO-Link 0I / 16O + PQI	0	2 Port S3	2	35...50	IO-Link 0I / 16O ...
Digital input_3	0	2 Port S4	3		Digital input
Digital output	0	2 Port S5		34	Digital output
Digital input_5	0	2 Port S6	5		Digital input
Deactivated	0	2 Port S7			Deactivated
Digital input_7	0	2 Port S8	7		Digital input

IO-Link 0I / 16O + PQI [IO-Link 0I / 16O + PQI]

General | IO tags | System constants | Texts

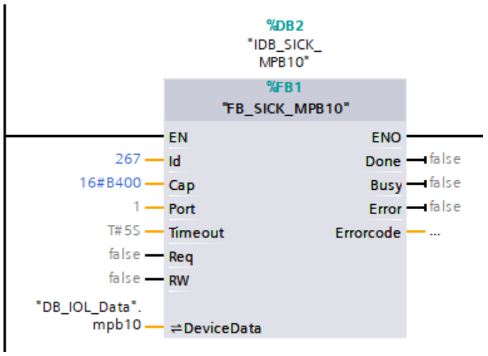
Show hardware system constant

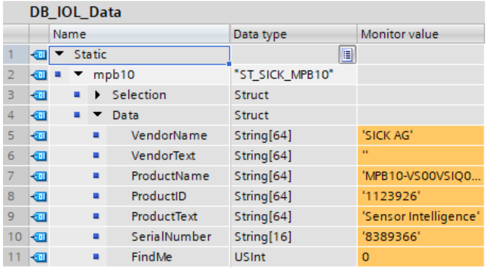
Name	Type	Hardware identi.	Used by	Comment
SIG300~SIG300_IO-Link_Master___8_Port_1~IO-Link...	Hw_SubModule	276	PLC_1	

8.2.12 SICK Function Block Factory

With the digital software service **Function Block Factory**, SICK provides a unique opportunity to generate specific PLC function modules.

The Function Block Factory can be used to generate function blocks individually for any IO-Link device (manufacturer independent). These function blocks are fully tested and documented. They make PLC programming much easier and faster and help to avoid errors. This can save time and money in the production process.




DB_IOL_Data

	Name	Data type	Monitor value
1	Static		
2	mpb10	"ST_SICK_MPB10"	
3	Selection	Struct	
4	Data	Struct	
5	VendorName	String[64]	'SICK AG'
6	VendorText	String[64]	''
7	ProductName	String[64]	'MPB10-VS00VSIQ0...
8	ProductID	String[64]	'1123926'
9	ProductText	String[64]	'Sensor Intelligence'
10	SerialNumber	String[16]	'8389366'
11	FindMe	USInt	0

Further information on SICK's Function Block Factory can be found on the following website.

https://fbf.cloud.sick.com

8.3 IOT Interfaces

NOTE
The interfaces REST API JSON and MQTT JSON require the following firmware version: V1.8

8.3.1 Communication via REST API

The SIG300 provides a REST API with JSON data format for accessing the data of the connected devices. These operating instructions provide an overview of the available device functions and the access mechanisms.

The REST API interface corresponds to the standard of the IO-Link community, which was defined in the document “JSON Integration for IO-Link” with version 1.0.0 (as of March 2020, part number 10.2020).

A connected laptop and appropriate software (e. g. Insomnia, Postman) can be used to read and write data in an IO-Link master via REST using JSON format commands.

8.3.1.1 General description of the interface

The REST API is a client-server interface and enables the client to request data from the server via defined resources. The REST API is stateless, which means that no information about the connection status and no information about the server or client is required.

The operation is based on HTTP methods. Common HTTP methods are GET, POST, PUT and DELETE. For the SIG300, the GET and POST request methods are particularly relevant, where the request and response data is represented in JSON format. JSON, or JavaScript Object Notation, is a minimal, visually readable format for structuring data. It is mainly used to transmit data between a server and a web application as an alternative to XML.

Table 24: Supported HTTP methods

HTTP method	Description
GET	Requests the specified data from the server (= data is only read and not changed)
POST	The payload is transmitted to the server (= write data)
DELETE	Deletes the specified resources on the server (= data is deleted)

8.3.1.2 API documentation

The current documentation (also called Open API) of the API interface can be downloaded at www.sick.de/SIG300. This document defines all available variables and methods of the API interface of the SIG300.

The OpenAPI documentation/specification is in JSON format and can be read out directly from device with appropriate software tools such as Swagger, Postman or Insomnia.

Address to get the OpenAPI document is:
http://[IP-ADDRESS]/iolink/v1/openapi

The list of SIG300 profiles can be found at:
[IP-ADDRESS]/capabilities/profile.

The two api profiles are:
IO-Link Master: "http:// [IP-ADDRESS]:8080/iolink/v1/openapi"
Rest Logic Blocks: "http:// [IP-ADDRESS]:8080/ac/restblocks/v1/flow/openapi"

8.3.1.3 API structure

Request

To request data, a command must be sent to the server, in this case the SIG300. This command is structured in HTTP format:

URL scheme: http://[IP-Adress]/[BasePath]/[Resource]

- The [**IP address**] corresponds to the valid IP address of the module. corresponds to the valid IP address of the module.
- The [**BasePath**] is defined in the standard of the IO-Link community and it is: "**/iolink/v1**"
- The **[Resource]** is used to address the corresponding parameter which is to be read or written. An example for “Resource” string is “masters/{masterNumber}/ports/{portNumber}/configuration”

An overview of the available variables and methods can be found in chapter see "Available variables and methods", page 45 and in the standard of the IO-Link community.

Thus, an example URL for reading the port configuration of port S3 is as follows:

http://192.168.0.1/iolink/v1/masters/1/ports/3/configuration

NOTE
The {masterNumber} variable is always 1, because the SIG300 gateway has only one master.

Response

For each request, the SIG300 responds with status information and data or only status information if no data is available. Depending on the request, this response can contain several pieces of information. In case of an error, the corresponding error code (see section XXX) is returned.

The response corresponds to the following format:

```
{
 “name1”: value1,
 “name2”: value2,
 “name…”: value…
}
```

- The “**name**” corresponds to the name of the object: e. g. **vendorId**
- The “**value**” corresponds to the value of the object: e. g.: 26.

Thus, an example response is as follows:

```
{
 “vendorId”: 26,
 “deviceId”: 8389238,
 “deviceAlias”: “Flow_control_1”
}
```

NOTE
No specific response time can be guaranteed for the use of the REST API interface since the HTTP requests are based on a standard TCP mechanism. The response time also depends on the system environment and system load. When using the web UI at the same time, the response time increases.

8.3.1.4 Available variables and methods

The SIG mainly supports the GET and POST methods. For selected variables, the DELETE function is also supported.

All API calls are executed synchronously. This means that every request is followed by a response. A minimum response time cannot be defined for the REST API commands, as this is also dependent in particular on the system environment and load used.

The following gives an overview of the available variables and methods. Detailed information and examples can be downloaded in the Open API documentation via the SICK website, see www.sick.com.

IO-Link REST API resources

Table 25: Overview of REST API HTTP resources
Input in the browser:
[SIG300 IP address]`/iolink/v1/gateway/identification`

Resource	Description
.../gateway/...	Addressing the gateway
.../masters	List of all available master variables and identification information
.../masters/{masterNumber}/...	Addressing of a specific master. The {masterNumber} variable is always 1, because the SIG300 gateway has only one master.
.../masters/{masterNumber}/ports	List of all available ports at a specific master including port number, status information and deviceAlias
.../masters/{masterNumber} /ports/{portNumber}/...	Addressing of a specific port connection at a specific master
.../devices	Addressing of all connected devices at all masters
.../devices/{deviceAlias}/...	Addressing a specific device via the device name
.../iodds	Managing IODD files stored on the device
.../mqtt/...	Configuration of the MQTT client defining publishing topics and configurations of the MQTT broker.

/gateway

Additional resources are defined for the individual HTTP methods:

Table 26: Gateway Resource overview
Input in the browser:
[SIG300 IPaddress]`/iolink/v1/gateway/identification`

Resource	HTTP Method	Description
.../gateway/identification	GET	Read out of identification data (e. g. MAC address or serial number of the device)
.../gateway/capabilities	GET	Information on device functions
.../gateway /configuration	GET	Read out of device or Ethernet configuration (e. g. DHCP or IP address)
.../gateway /configuration	POST	Writing of device or Ethernet configuration
.../gateway /events	GET	Read the event log containing all events from Gateway, Masters, Ports and Devices
.../gateway/reset	POST	Resetting to factory settings
[SIG300IP address]/iolink/v1/**vendor**/gateway/findme	POST	Activate/Deactivate “Find me function”
SIG300IP address]/iolink/v1/vendor/gateway /access-rights	GET	Read the gateway access rights
SIG300IP address]/iolink/v1/vendor/gateway /access-rights	POST	Write the gateway access rights

/masters

Further resources are also available here:

Table 27: Overview of the master resource
Input in the browser
[SIG300 IPaddress]`/iolink/v1/masters/1/capabilities`

Resource	HTTP Method	Description
.../masters	GET	List of all available master variables and identification information
.../masters/capabilities	GET	Read out of functions of the specific IO-Link Master (e. g. the number of ports or the maximum voltage supply)
.../masters /identification	GET	Read out of identification data of the specific IO-Link Master (e. g. vendor ID, serial number or firmware version)
.../masters /identification	POST	Writing of identification data of the specific master (e. g. LocationTag, functionTag)

/masters/1/ports

With the following resources, additional information of the individual ports at a specific master can be called up:

Table 28: Overview of the port resource
Input in browser:
[SIG300 IPaddress]/iolink/v1/masters/1/ports/[portNumber]/status

Resource	HTTP Method	Description
.../masters/1/ports	GET	List of all available ports at a specific master including port number, status information and deviceAlias
.../masters/1/ports/[portNumber]	GET	Read out of functions of the specific port (e. g. port type or max. voltage supply)
.../masters/1/ports/[portNumber]/status	GET	Read out of current status of a specific port (e. g. deactivated or the IO-Link version of the connected IO-Link device)
.../masters/1/ports/[portNumber]/configuration	GET	Read out of configuration of a specific port (e. g. cycle time or deviceAlias of the connected IO-Link device)
.../masters/1/ports/[portNumber]/configuration	POST	Writing of configuration on a specific port (e. g. deviceAlias)
.../masters/1/ports/[portNumber]/datastorage	GET	Read out of Data Storage object
.../masters/1/ports/[portNumber]/datastorage	POST	Writing of Data Storage object
.../masters/1/ports/[portNumber]/power	POST	Enable/disable the port power (pins 1 and 3) on the Class B ports
.../masters/1/ports/[portNumber]/accessrights	GET	Read the access rights for the specified port
.../masters/1/ports/[portNumber]/accessrights	POST	Write the access rights for the specified port

/devices

The connected devices are addressed via the" deviceAlias".

With "deviceAlias" can be assigned a custom name to the specific port. If no deviceAlias is defined, then the default value (e. g. master1port6) is used.

Example:
An IO-Link device with deviceAlias "sensor34" is connected to port 6. The addressing is performed with the following request:

`GET` [SIG300 IPaddress]`/iolink/v1/devices/sensor34/identification`

However, if no deviceAlias is defined and thus the default value is used, then the request changes as follows:

`GET` [SIG300 IPaddress]`/iolink/v1/devices/master1port6/identification`

The description "master1port6" therefore stands for the sixth port on the first master.

All connected devices are listed via the request:
`GET` [SIG300 IPaddress]`/iolink/v1/devices`

The following resources can be used to call up additional information on the connected devices:

Table 29: Gateway Resource overview
Input in browser
[SIG300 IPaddress]`/iolink/v1/devices/`[deviceAlias]`/events`

Resource	HTTP Method	Description
.../devices	GET	Addressing of all connected devices at all masters
.../devices/[deviceAlias]/capabilities	GET	Read out of device information of the connected device (e. g. min. cycle time)
.../devices/[deviceAlias]/identification	GET	Read out of device identification data
.../devices/[deviceAlias]/identification	POST	Writing of device identification data
Process data		
.../devices/[deviceAlias]/processdata/value	GET	Read out of process data (input and output data) of a specific IO-Link device
.../devices/[deviceAlias]/processdata /getdata/value	GET	Read out of process input data of a specific IO-Link device
.../devices/[deviceAlias]/processdata /setdata/value	GET	Read out of process output data of a specific IO-Link device
.../devices/[deviceAlias]/processdata/value	POST	Writing of process output data on a specific IO-Link device
Parameter values		
.../devices/[deviceAlias]/parameters /{index}/value	GET	Read out of parameter values of a specific device by means of the index. Index value can be found in the sensor specific documentation.
.../devices/[deviceAlias]/parameters /{index}/subindices /{subindex}/value	GET	Read out of parameter values of a specific device by means of the index and subindex. Index and subindex values can be found in the sensor specific documentation.

Resource	HTTP Method	Description
.../devices/[deviceAlias]/parameters /{index}/value	POST	Writing of a parameter value using the index. Index value can be found in the sensor specific documentation.
.../devices/[deviceAlias]/parameters /{index}/subindices /{subindex}/value	POST	Writing of a parameter value using the index and subindex. Index and subindex values can be found in the sensor specific documentation.
Events		
.../devices/[deviceAlias]/events	GET	Read out of all events for a specific device

It is possible to read data from a device in a human readable IODD format. This requires an upload of the IODD file of the referring device during commissioning. Input in browser:
`GET` [SIG300 IPaddress]`/iolink/v1/devices/[deviceAlias]/processdata/value?format=iodd`

/iodds

With the following resources, it is possible to manage additional information about installed IODD (IO Device Description).

Table 30: IODD file handling
Input in browser:
[SIG300 IPaddress]`/iolink/v1/iodds`

Resource	HTPP Method	Description
.../iodds	GET	Get a list of all IODD (representations) that are available on the Gateway
.../iodds/file	POST	Store or update an IODD
.../iodds/file	GET	Get a specific IODD
.../iodds	DELETE	Delete a specific IODD representation

/mqtt

With the following resources, it is possible to manage information about MQTT services.

Table 31: MQTT client configuration
Input in browser:
[SIG300 IPaddress]`/iolink/v1/mqtt/configuration`

Resource	HTPP Method	Description
.../configuration	GET	Read the MQTT configuration of the Gateway
.../configuration	POST	Update the MQTT configuration of the gateway
.../topics	GET	Get the list of MQTT topics
.../topics	POST	Create a new MQTT topic
.../topics/{topicId}	DELETE	Delete a specific MQTT topic
.../topics/{topicId}	GET	Get one specific MQTT topic
../connectionstatus	GET	Read the connection status of the MQTTclient to the MQTT server

Logic editor REST commands

Special Resources linked to the “Logic Editor”

The **[BasePath]** is: “/ac/restblocks/v1”

The **[Resource]** are described on below table.

Input in the browser:
[SIG300 IPaddress]`/ac/restblocks/v1/flow/openapi`

Table 32: Logic editor handling
Input in the browser:
[SIG300 IPaddress]/ac/restblocks/v1/flow/openapi

Resource	HTTP Method	Description
.../ac/restblocks/v1/flow/openapi	GET	Get the REST interface description of the IO-Link API as OpenAPI YAML document.
.../ac/restblocks/v1/flow/ apiversion	GET	Get the REST interface version identification.
.../ac/restblocks/v1/flow/inputs	GET	Description of active Rest Input blocks. The results show the offset and data type for each active block.
.../ac/restblocks/v1/flow/outputs	GET	Number of output blocks
.../ac/restblocks/v1/flow/inputs/process-data/value	GET	Get all input data
.../ac/restblocks/v1/flow/inputs/{offset number}/process-data/value	GET	Read an input value for a specified offset
.../ac/restblocks/v1/flow/inputs/{offset number}/process-data/value	POST	Write Rest block data to the specified offset
.../ac/restblocks/v1/flow/outputs/process-data/value	GET	Get all output data
.../ac/restblocks/v1/flow/outputs/{offset value}/process-data/value	GET	Read an output for a specified offset

8.3.1.5 Status code and error messages

Errors may occur when processing HTTP requests. Several errors are defined.

The following rules apply to troubleshooting:

- If multiple errors occur while processing the request, only the first detected error is responded to.
- If no REST API commands are available, error 103 is returned.

Error messages are structured as follows:

```
{
“code”: 102,
“message”: “Internal communication error”
}
```

The following table provides an overview of the possible error codes occurring with a REST command.

Table 33: Error messages

Error code	HTTP code	Message	Note
General error			
101	500	Internal server error	This error can occur with any request

Error code	HTTP code	Message	Note
102	500	Internal communication error	This error can occur with any request
103	404	Operation not supported	This error is returned if the requested function does not exist.
104	400	Action locked by another client	Fieldbus controller or another participant blocks access
150	403	Permission denied	Access is not allowed. Check access rights in the configuration. This error can occur with any request.
JSON parsing error			
201	400	JSON parsing failed	Error while parsing the incoming JSON value
202	400	JSON data value invalid	Error while parsing a specific JSON value, such as an incorrect IP address
203	400	JSON data type invalid	E. g.: data type string instead of number
204	400	Enumeration value unknown	
205	400	JSON data value out of range	Exceeds the minimum or maximum value
206	400	JSON data value out of bounds	An array/string was accessed whose maximum length was exceeded.
207	400	deviceAlias is not unique	
208	400	POST request without content	
Error during resource access			
301	404	Resource not found	E. g. incorrect URL
302	404	masterNumber not found	
303	404	portNumber not found	
304	404	deviceAlias not found	
305	400	Query parameter name invalid	
306	400	Query parameter value invalid	
307	400	Port is not configured to IO-Link	E. g.: IOLINK_MANUAL or IOLINK_AUTOSTART mode
308	404	IO-Link device is not accessible	E. g. not connected or communication error
309	404	IO-Link parameter not found	
310	404	IO-Link parameter access not supported by the device	
311	400	IO-Link parameter access error	The additional “iolinkErrorCode” and “iolinkErrorMessage” fields contain the IO-Link error code and the event text from the ErrorTypes table.
312	404	IO-Link parameter name is not unique	Please use the format [Name]_[Index].
DataStorage error			
401	400	Data storage mismatch	No match between configured device and data from data memory. Check device ID.
Processing error in the process data			
501	400	I/Q is not configured as DIGITAL_OUTPUT	Writing process data on I/Q is not possible
502	400	C/Q is not configured as DIGITAL_OUTPUT	Writing process data on C/Q is not possible

Error code	HTTP code	Message	Note
503	400	IO-Link device has no output process data	
Error in the payload			
701	400	Data set incomplete	
702	400	Data set not applicable	The entire data set is denied
703	400	Data set combination incompatible	The entire data set is denied

8.3.2 MQTT client

The SIG300 provides an MQTT interface with JSON data format for accessing the data of the IO-Link Master and of the connected devices. These operating instructions provide an overview of the available messages and the access mechanisms.

INFO: MQTT services can be set via REST API or via the UI.

8.3.2.1 General description

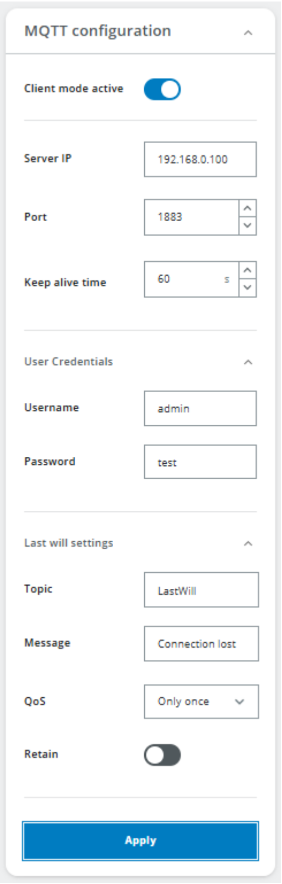
The MQTT (Message Queuing Telemetry Transport) protocol is an open network protocol for machine-to-machine communication that enables the transmission of telemetric data between devices. The built-in MQTT client allows the device to publish a specific set of information to an MQTT broker.

To access MQTT services via laptop it is suggested to use dedicated software (e. g.: “Mosquitto” or MQTT explorer”)

MQTT Quality of Service can be configured for each topic separately via dedicated UI interface.

8.3.2.2 MQTT client configuration

Select Configuration – MQTT settings to setup the MQTT communication.



MQTT configuration

Client mode active

Server IP 192.168.0.100

Port 1883

Keep alive time 60 s

User Credentials

Username admin

Password test

Last will settings

Topic LastWill

Message Connection lost

QoS Only once

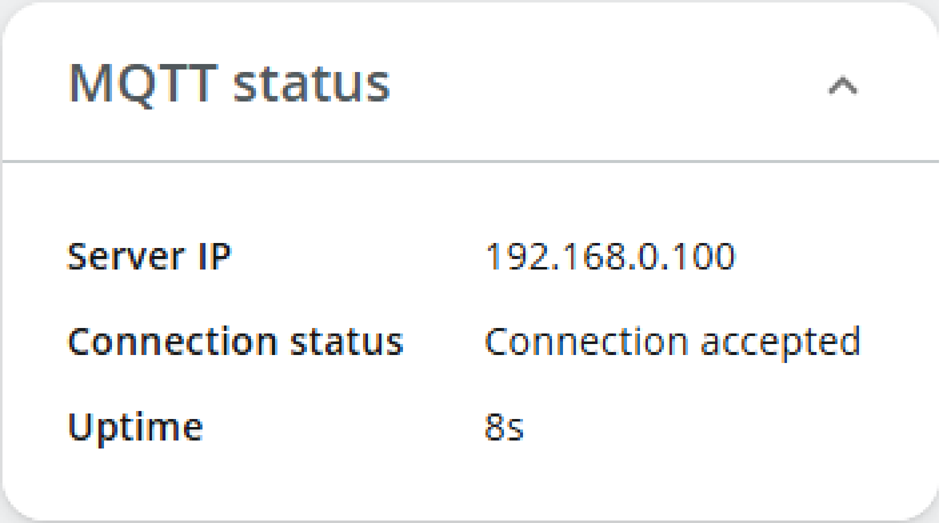
Retain

Client mode active – toggle to activate the MQTT client on SIG300
Server IP – IP address of the MQTT Broker
Port – port number used for MQTT communication
Username – username set in the broker's optional credentials
Password – password set in the broker's optional credentials
Keep alive time – how often a heartbeat message is sent to the broker to keep the connection alive
Last will message – an optional message that is sent to the specified topic to notify the other clients that this client (SIG300) has disconnected
Topic – topic where the last will message will be sent to
Message – message content
QoS – quality of service:

- **Only once (QoS 0)** – sending message only once without acknowledgement
- **At least once (QoS 1)** – sending with acknowledgement, duplicates might be possible
- **Exactly once (QoS 2)** – with acknowledgement and making sure there are no message duplicates

Retain – if set to true, the clients will receive this message even if they are connected to the broker after the message was sent

You will see "Connection accepted" status if the connection with the MQTT broker is established successfully.



MQTT status

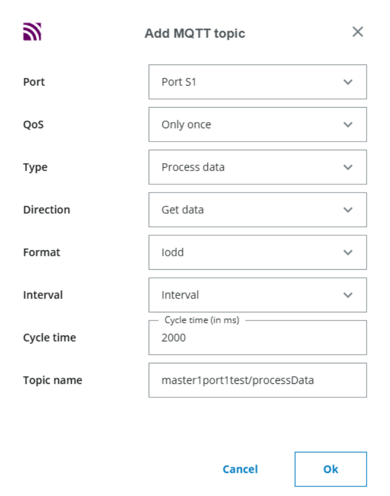
Server IP 192.168.0.100

Connection status Connection accepted

Uptime 8s

8.3.2.3 Messages – topics

To start sending the data via MQTT, click "Add topic" to configure a new message.



Add MQTT topic

Port	Port S1
QoS	Only once
Type	Process data
Direction	Get data
Format	Iodd
Interval	Interval
Cycle time	Cycle time (in ms) 2000
Topic name	master1port1test/processData

Port – sensor port you want to forward data from

QoS – quality of service:

- **Only once (QoS 0)** – sending message only once without acknowledgement
- **At least once (QoS 1)** – sending with acknowledgement, duplicates might be possible
- **Exactly once (QoS 2)** – with acknowledgement and making sure there are no message duplicates

Type – what kind of data shall be sent

- **Process data** – process data of the IOLink device
- **Event** – an array with the latest port events, such as disconnection or change of device

Direction (Type = Process data)

- **Get data** – read process data of the device
- **Set data** – read the latest data written to the device
- **Get/Set data** – both Get and Set objects will be included in the payload

Format (Type = Process data)

- **Byte** – raw byte array format of process data
- **IODD** – pre-processed process data according to IODD description

Interval – select when the data must be sent

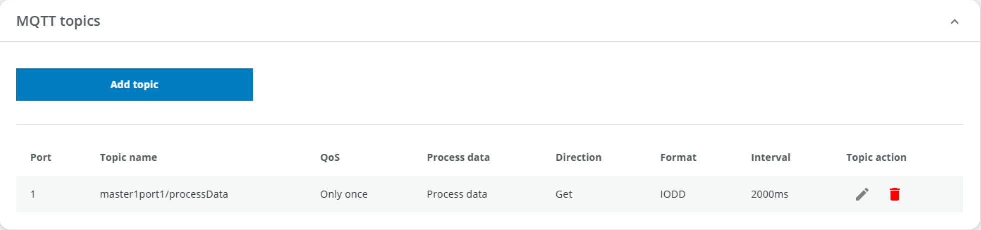
- **Interval** – send data periodically
- **OnChange** – send data when the value changes (useful when Type = Event)

Cycle time – period in milliseconds in case of interval sending is selected

Topic name - topic where the data will be sent to

After the topic is added successfully, the data transfer over MQTT starts immediately.

It is possible to edit and delete the topic if necessary (see the “Topic action” below).



MQTT topics

Port	Topic name	QoS	Process data	Direction	Format	Interval	Topic action
1	master1port1/processData	Only once	Process data	Get	IODD	2000ms	✎ 🗑

8.4 Web-Interface

The SIG300 can be accessed via the integrated web interface if proper rights are granted (see "Cybersecurity", page 66).

Access by Ethernet

SIG300 offers two Ethernet ports: P1, P2.

Additionally, the current IP address can be read out via the SOPAS engineering tool.

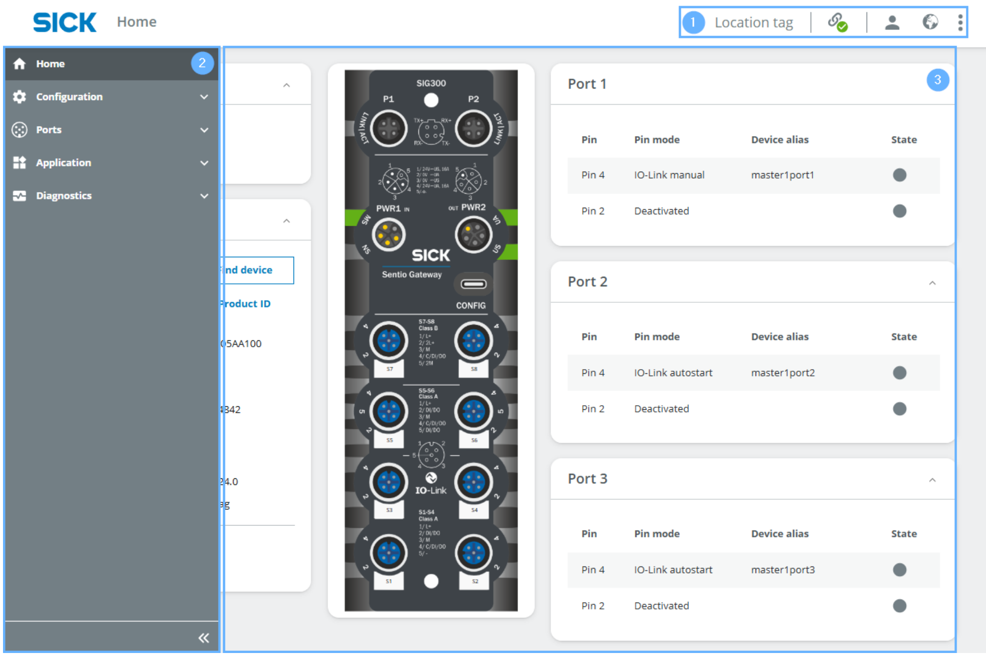
NOTE

- The SIG300 supports HTTP only (no HTTPs).
- To change settings, it is necessary to log in with a certain user level, see see table 34, page 58.

Access by USB

To access the web interface, connect the device to your computer via USB. This creates a virtual network in which SIG300 gets the IP adress 169.254.0.1. Enter the IP adress in the adress line of the web browser.

8.4.1 Web interface structure



Port 1

Pin	Pin mode	Device alias	State
Pin 4	IO-Link manual	master1port1	●
Pin 2	Deactivated		●

Port 2

Pin	Pin mode	Device alias	State
Pin 4	IO-Link autostart	master1port2	●
Pin 2	Deactivated		●

Port 3

Pin	Pin mode	Device alias	State
Pin 4	IO-Link autostart	master1port3	●
Pin 2	Deactivated		●

Figure 7: GUI parts of the integrated web interface

① Top menu

② Navigation menu

③ Working Area

When configurations are changed in the module/tile, these changes are implemented immediately. Manual saving of changes is not required.

However, there are some functions or commands that require to be actively confirmed and transmitted to the module. This case is represented by a blue "Apply" button directly below the corresponding input area.

Whether the process activated by “Apply” was successful, this is reported back directly via a message at the upper right edge of the web interface. The message will disappear automatically after a few seconds.

8.4.2 Top menu

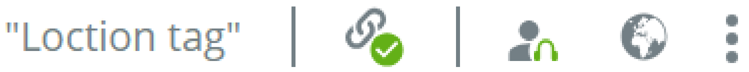
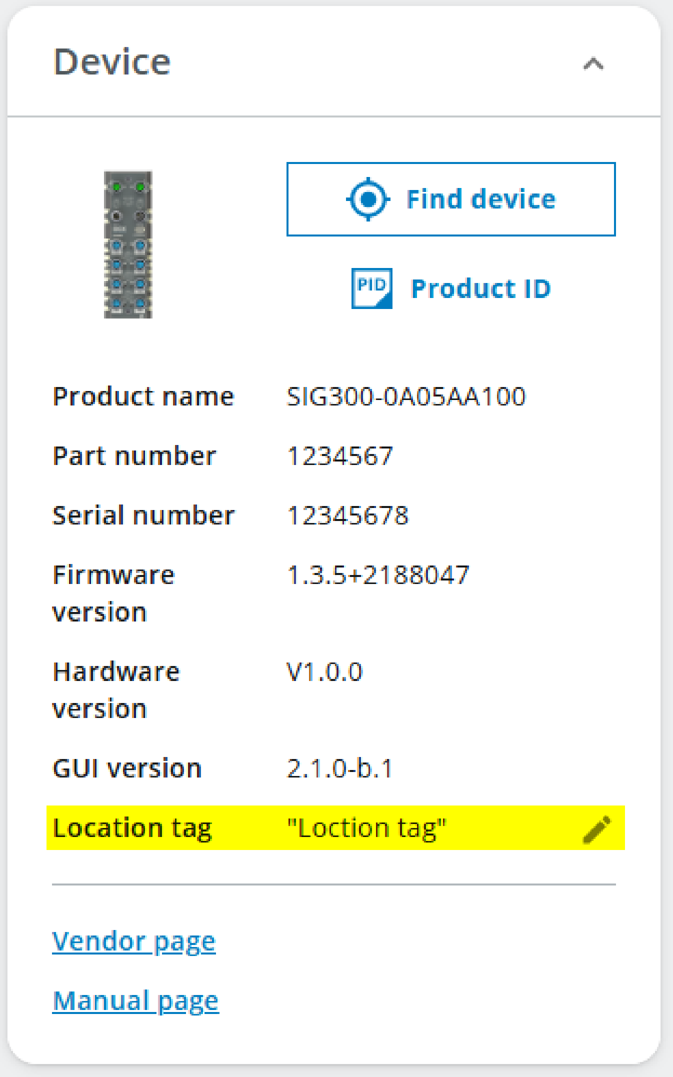


Figure 8: Top menu command bar

Location tag

 "Location tag"	It is showed the value of the “Location tag” parameter. This value can be set with appropriated rights in the “Device” tile.  Device Find device Product ID Product name SIG300-0A05AA100 Part number 1234567 Serial number 12345678 Firmware version 1.3.5+2188047 Hardware version V1.0.0 GUI version 2.1.0-b.1 Location tag "Loction tag" Vendor page Manual page

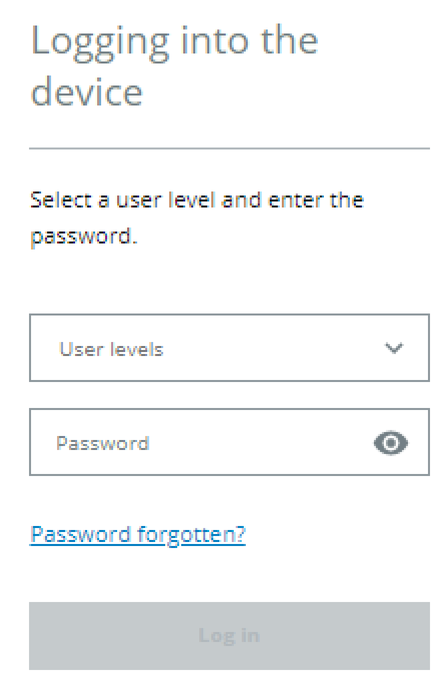
Device connection

🔗✓	Device ethernet communication active
🔗!	Device ethernet communication interrupted

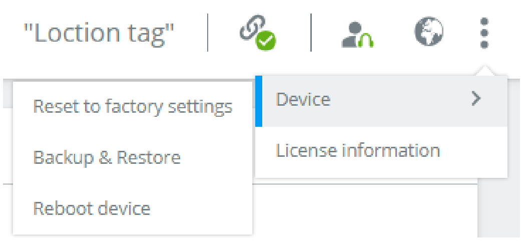
Language options

🌐	Select language for the software interface (German/English).

User management

	 Logging into the device Select a user level and enter the password. Clicking on “User” icon, the “Login interface” layer is shown. More information on “user levels”: see table 34, page 58 “Password forgotten” (Reset password) process can be started only using an USB-connection.

Device specific operations

 "Loction tag" Reset to factory settings Backup & Restore Reboot device Device License information	Device specific operations and messages. - Device > Device > “Reset to factory settings”: All parameters of the master are set to their default factory values. Only “Service” use role can perform this operation. - Device > Device > “Reboot”: Device reboot is performed, and the page is automatically refreshed after few seconds. - Device > "Backup & Restore" Here you can create a backup of the current device or transfer a configuration to a new device.

To change settings, you must log in at the Maintenance user level (read and write access). By default, you are logged in with the “Run” (read-only) user level, where you can only view data and parameterization.

User other than “Run” require a password.

Saving the login may depend on the browser’s cookie settings.

When logging in for the first time, you will be prompted to change the default password. In the interest of cybersecurity, it is strongly recommended that you create a new, different password.

If the password is lost or it is needed to reset all passwords, it is possible to require an unlocking code (see “Password forgotten” in “User management” top menu)

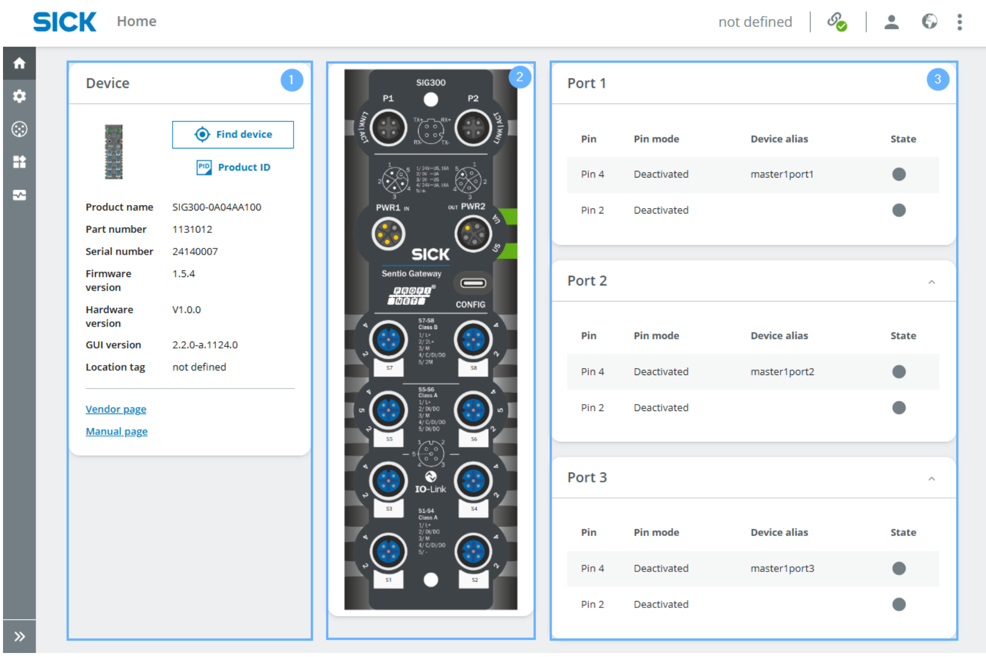
Table 34: User and default passwords

User	User level	Default password	Role description
Operator	Run	No password required	Reading parameterization
Maintenance	Maintenance personnel	"main"	Reading and writing parameterization
Service	Service	"servicelevel"	Performing advanced settings like firmware updates.

8.4.3 Home

⌂	**“Home”**

The “Home” page is the start page for the SIG300. It provides an overview of the current module status and device function.



Product name	SIG300-0A04AA100
Part number	1131012
Serial number	24140007
Firmware version	1.5.4
Hardware version	V1.0.0
GUI version	2.2.0-a.1124.0
Location tag	not defined

Vendor page

Manual page

Port 1

Pin	Pin mode	Device alias	State
Pin 4	Deactivated	master1port1	●
Pin 2	Deactivated		●

Port 2

Pin	Pin mode	Device alias	State
Pin 4	Deactivated	master1port2	●
Pin 2	Deactivated		●

Port 3

Pin	Pin mode	Device alias	State
Pin 4	Deactivated	master1port3	●
Pin 2	Deactivated		●

'The “Home” working area is divided into three areas.

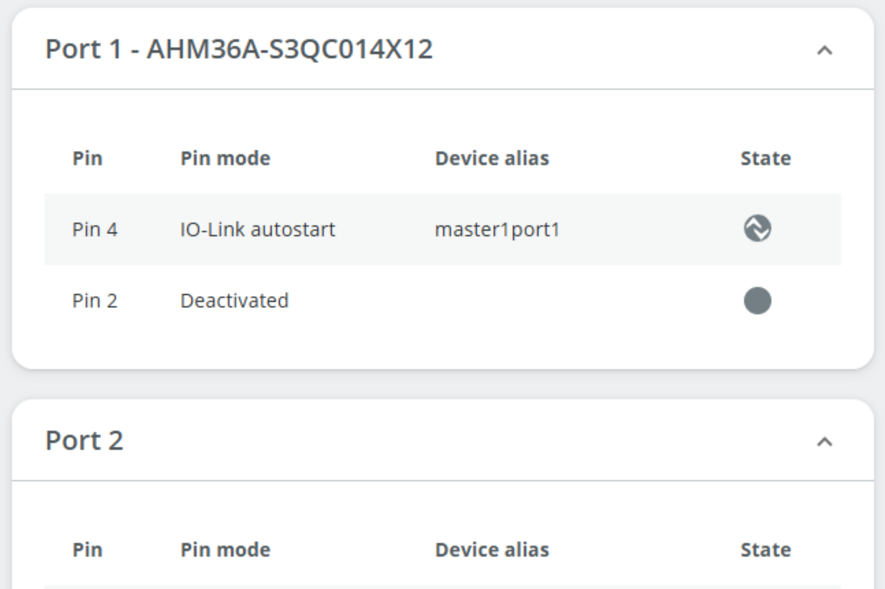
① This area shows identification data, information about the firmware and software versions as well as the vendor information of the module.

② This section shows a compact image of the SIG300, which shows the “deviceAlias” (port alias identifier) for each port and the live state of the device LEDs. (For more information about the LED blinking patterns see Chapter "Status indicators")

NOTE

The LED displays do not work in real time.

③ This section shows an overview of the parameterization of the individual ports. In addition to the settings of pin 2 and pin 4, the “deviceAlias” and the communication state of each port is displayed.
This overview corresponds to the configuration as it has been made on the “Ports” menu.



Port 1 - AHM36A-S3QC014X12

Pin	Pin mode	Device alias	State
Pin 4	IO-Link autostart	master1port1	⟲
Pin 2	Deactivated		●

Port 2

Pin	Pin mode	Device alias	State

8.4.4 Configuration

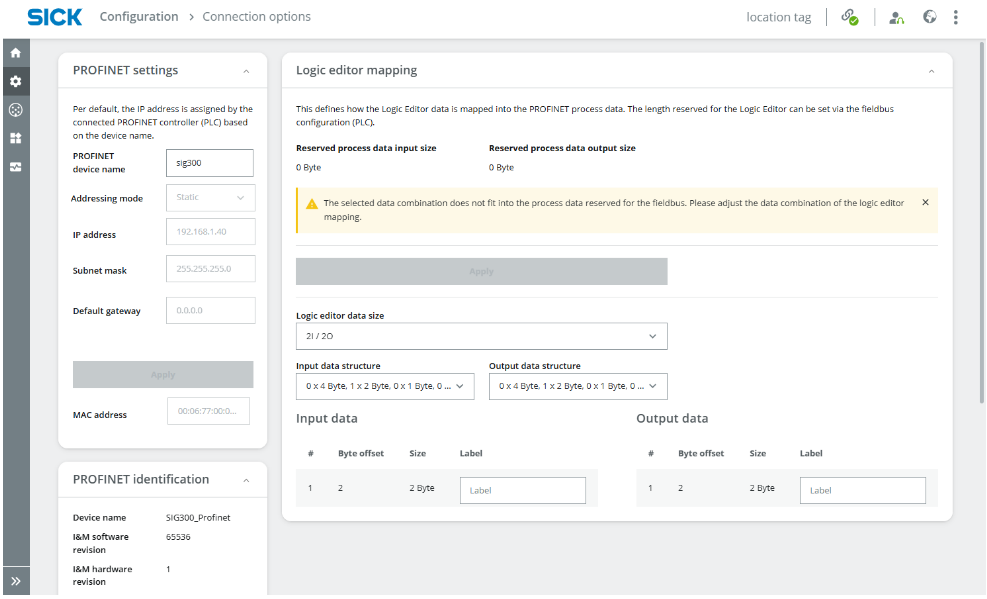
⚙	"Configuration"

8.4.4.1 Connection options

On the “Connection options” page, the Ethernet settings such as the IP address or the subnet mask can be changed.

Standard IP adress: see "IP address", page 24

The data that the Logic Editor is to exchange with the fieldbus can also be described here.



PROFINET settings

Per default, the IP address is assigned by the connected PROFINET controller (PLC) based on the device name.

PROFINET device name	sig300
Addressing mode	Static
IP address	192.168.1.40
Subnet mask	255.255.255.0
Default gateway	0.0.0.0
MAC address	00:06:77:00:0...

PROFINET identification

Device name	SIG300_Profinet
I&M software revision	65536
I&M hardware revision	1

Logic editor mapping

This defines how the Logic Editor data is mapped into the PROFINET process data. The length reserved for the Logic Editor can be set via the fieldbus configuration (PLC).

Reserved process data input size	Reserved process data output size
0 Byte	0 Byte

⚠ The selected data combination does not fit into the process data reserved for the fieldbus. Please adjust the data combination of the logic editor mapping. ×

Logic editor data size: 2I / 2O

Input data structure: 0 x 4 Byte, 1 x 2 Byte, 0 x 1 Byte, 0 ...

Output data structure: 0 x 4 Byte, 1 x 2 Byte, 0 x 1 Byte, 0 ...

Input data

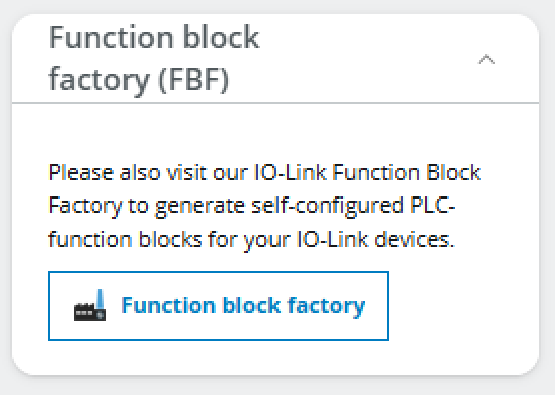
#	Byte offset	Size	Label
1	2	2 Byte	Label

Output data

#	Byte offset	Size	Label
1	2	2 Byte	Label

Figure 9: PROFIENT configuration page

8.4.4.1.1 Function Block Factory



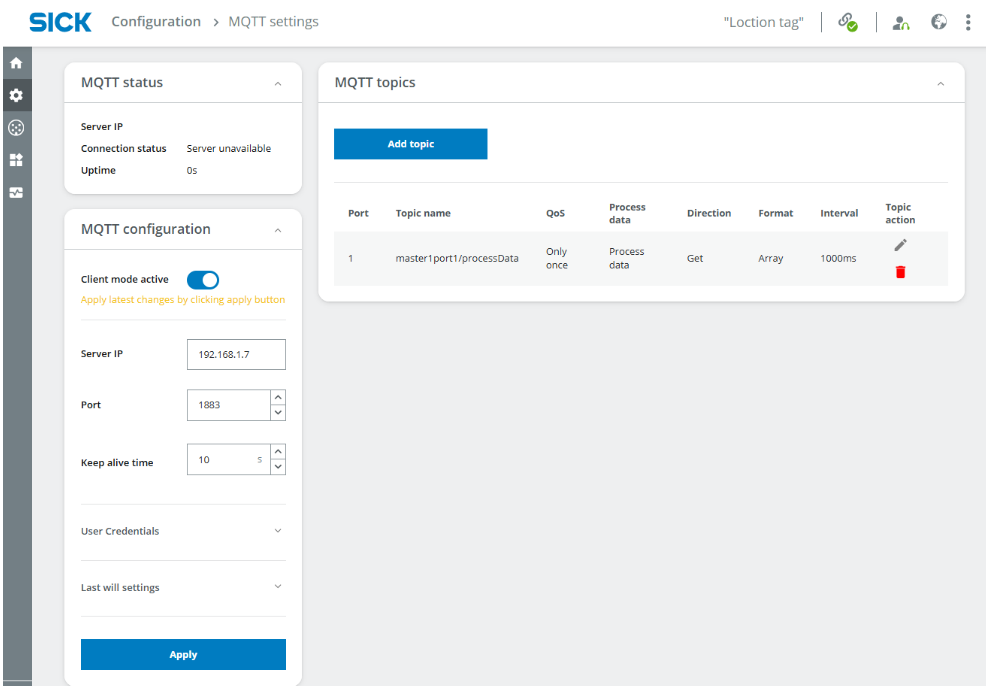
Function block factory (FBF)

Please also visit our IO-Link Function Block Factory to generate self-configured PLC-function blocks for your IO-Link devices.

The Function Block Factory (FBF), a tool developed by SICK, enables the generation of PLC function blocks based on an IODD, independent of the device type or manufacturer.

8.4.4.2 MQTT Settings

Using the MQTT protocol, you can read or write IO-Link process data from the connected devices.



MQTT status

Server IP
Connection status Server unavailable
Uptime 0s

MQTT configuration

Client mode active ●
Apply latest changes by clicking apply button

Server IP 192.168.1.7
Port 1883
Keep alive time 10 s

MQTT topics

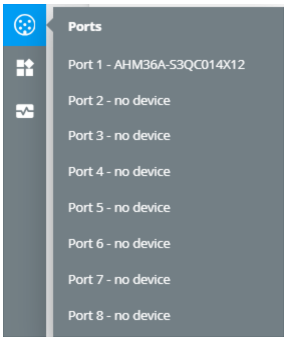
Port	Topic name	QoS	Process data	Direction	Format	Interval	Topic action
1	master1port1/processData	Only once	Process data	Get	Array	1000ms	

It is possible to enable or disable MQTT server via “Client mode active” toggle switch. Server IP and additional parameters must be entered according to the MQTT broker settings.

To accept the MQTT configuration, parameter changes must be accepted with "Apply".

The MQTT topics and its content can be defined according to the topic configuration.

8.4.5 Ports

 Ports Port 1 - AHM36A-S3QC014X12 Port 2 - no device Port 3 - no device Port 4 - no device Port 5 - no device Port 6 - no device Port 7 - no device Port 8 - no device	"Ports"

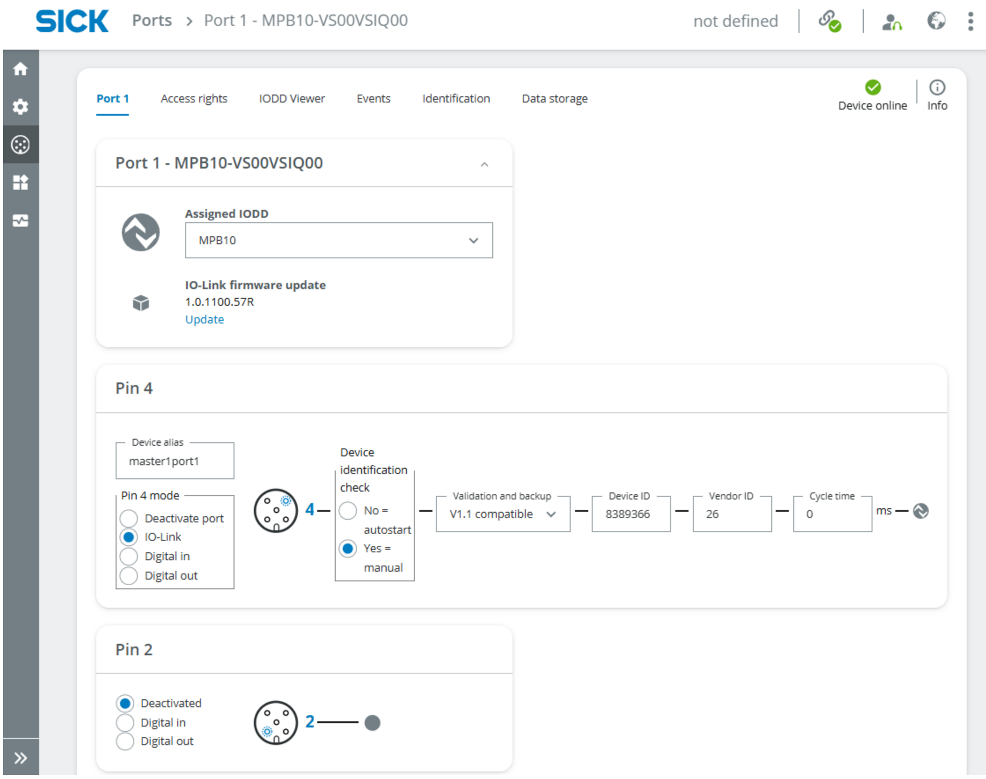
Settings for connection ports S1 to S8 can be changed in the “Ports” pages. There is a separate subpage for each port of the SIG300.

8.4.5.1 Ports 1 ... 8

All settings relating to the IO-Link port can be made here.

Editing the port settings requires to be logged in as “Maintenance” or “Service” and additionally setting the proper “Access rights” for that single port using the specific tile (see "Port Access Rights", page 25).

If the access rights are assigned to the fieldbus, the configuration is done via the configuration assembly.



SICK Ports > Port 1 - MPB10-VS00VSIQ00 not defined

Port 1 | Access rights | IODD Viewer | Events | Identification | Data storage

Device online | Info

Port 1 - MPB10-VS00VSIQ00

Assigned IODD: MPB10

IO-Link firmware update
1.0.1100.57R
Update

Pin 4

Device alias: master1port1

Pin 4 mode
○ Deactivate port
● IO-Link
○ Digital in
○ Digital out

Device identification check
○ No = autostart
● Yes = manual

Validation and backup: V1.1 compatible

Device ID: 8389366

Vendor ID: 26

Cycle time: 0 ms

Pin 2
● Deactivated
○ Digital in
○ Digital out

Port setup

The “Device alias” represent the alias for the specific port. The alias will not change if a different device is connected to this port.

Pin 4 setting includes the IO-Link communication. The “pin mode” can be:

- Deactivate port
- IO-Link (default)
- Digital in
- Digital out

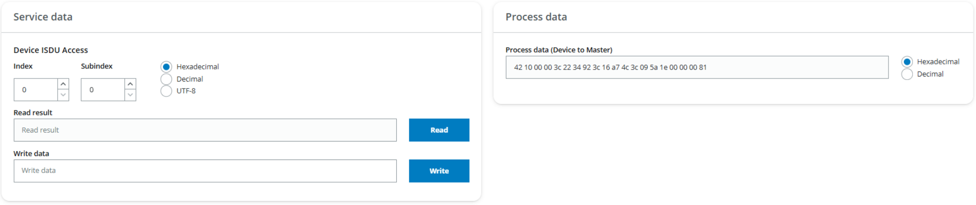
Selecting “IO-Link” mode, the option box “Device identification check” is shown.

Selecting “Yes” there will be no validation of the connected device. It means that there will be not a check of the “IO-Link ID device” and even the “backup and restore” related functions cannot be set.

Selecting “No” a validation check of the connected device is performed and the “backup and restore” related functionalities are available.

For a description of “Backup and Restore” and “Restore” functions: see "Data storage", page 64

Generic IO-Link data access



Service data

Device ISDU Access

Index: 0

Subindex: 0

● Hexadecimal
○ Decimal
○ UTF-8

Read result: Read result

Write data: Write data

Process data

Process data (Device to Master): 42 10 00 00 3c 22 34 92 3c 16 a7 4c 3c 09 5a 1e 00 00 00 81

● Hexadecimal
○ Decimal

Depending on the connected device's capabilities, the process data panels will be shown. If there is any "Output data", the panel "Process data (Master to Device)" is shown. If there is any "Input data", the panel "Process data (Device to Master)" is shown.

Furthermore, depending on the configuration on the port page for the individual pins further tiles can be represented:

- In case of any pins configured as "Digital out", they will be shown in the "Process data (Master to device)" panel.
- In case of any pins configured as "Digital in", they are shown in "Process data (Device to Master)" panel.

IO-Link data can be read from the connected device (Device to Master), but commands can also be issued to the connected device (Master to Device) in the IODD viewer tab.

Process data from or to the IO-Link device is transported cyclically between IO-Link master and connected device. For parameter data (service data), it is necessary that these are explicitly requested by the master. They are transmitted acyclic.

Using the ISDU (Index Service Data Unit), access is made via the corresponding index number and subindex number. This method is used if IODD file of the connected device is not available.

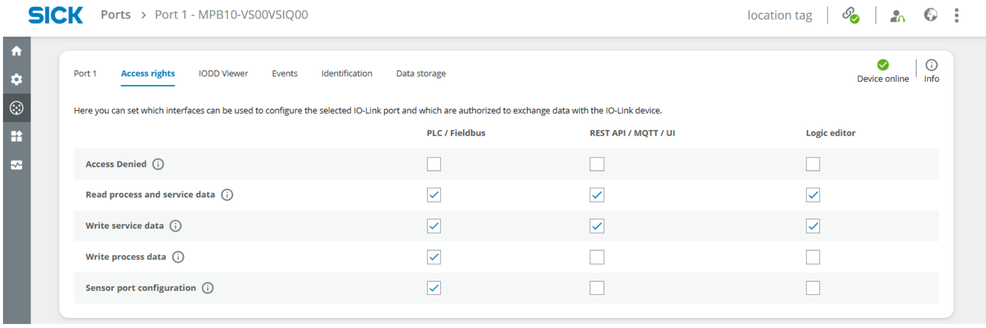
NOTE
The available process data and index number are provided by the manufacturer of the IO-Link connected device in the data sheet

To edit the value, a proper access right has to be set.

8.4.5.2 Access rights

Here you can set which interfaces should be used to configure the selected IO-Link port and which are authorized to exchange data with the IO-Link device.

The writing of process data and the port configuration can only be handled by one client.



Here you can set which interfaces can be used to configure the selected IO-Link port and which are authorized to exchange data with the IO-Link device.

	PLC / Fieldbus	REST API / MQTT / UI	Logic editor
Access Denied ⓘ	□	□	□
Read process and service data ⓘ	☑	☑	☑
Write service data ⓘ	☑	☑	☑
Write process data ⓘ	☑	□	□
Sensor port configuration ⓘ	☑	□	□

Figure 10: Access rights for the specific port

NOTE
Each port has its own “Access rights” setting.

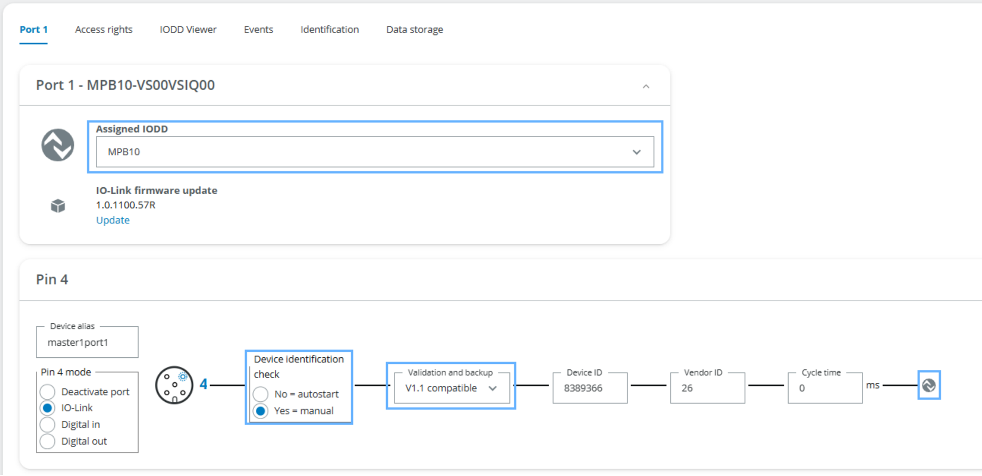
The Logic Editor only supports read and write process data.

8.4.5.3 IODD Viewer

The IODD Viewer is a graphic view of the IODD file. The process data and service data of the IO-Link device can be read or written here.

The IODD viewer can be used once the IODD has been uploaded to the SIG300 and assigned to the corresponding port.

To upload IODD files navigate to "Application ➜ IODD File Management" in the web interface. Then go to “Ports ➜ Port X”, set “Device identification check” to “Yes = manual” and configure “Port assignment from IODD”



Port 1 - MPB10-VS00VSIQ00

Assigned IODD: MPB10

IO-Link firmware update
1.0.1100.57R
Update

Pin 4

Device alias: master1port1

Pin 4 mode
○ Deactivate port
● IO-Link
○ Digital in
○ Digital out

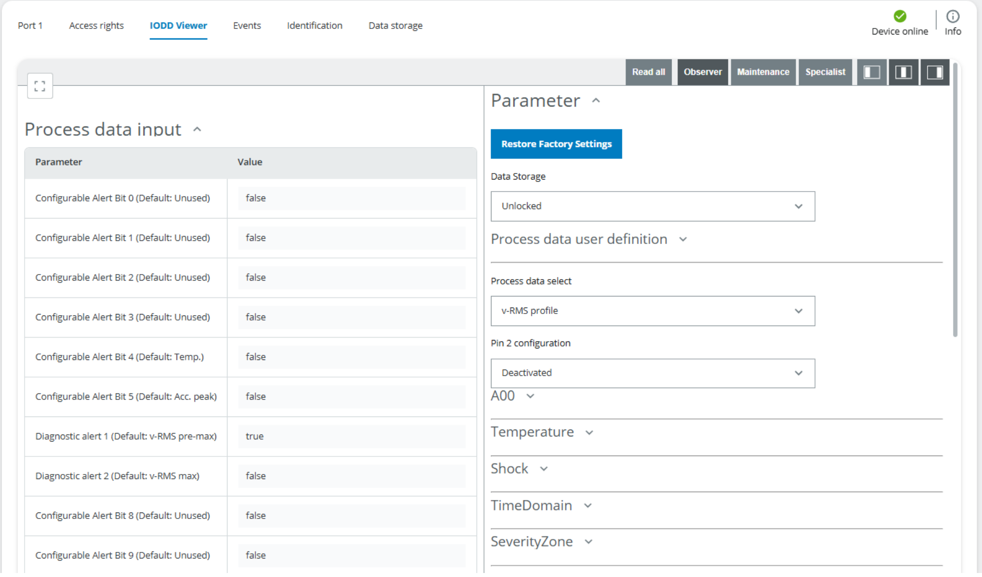
Device identification check
○ No = autostart
● Yes = manual

Validation and backup: V1.1 compatible

Device ID: 8389366

Vendor ID: 26

Cycle time: 0 ms



Process data input

Parameter	Value
Configurable Alert Bit 0 (Default: Unused)	false
Configurable Alert Bit 1 (Default: Unused)	false
Configurable Alert Bit 2 (Default: Unused)	false
Configurable Alert Bit 3 (Default: Unused)	false
Configurable Alert Bit 4 (Default: Temp.)	false
Configurable Alert Bit 5 (Default: Acc. peak)	false
Diagnostic alert 1 (Default: v-RMS pre-max)	true
Diagnostic alert 2 (Default: v-RMS max)	false
Configurable Alert Bit 8 (Default: Unused)	false
Configurable Alert Bit 9 (Default: Unused)	false

Parameter

Restore Factory Settings

Data Storage: Unlocked

Process data user definition

Process data select: v-RMS profile

Pin 2 configuration: Deactivated

A00

Temperature

Shock

TimeDomain

SeverityZone

8.4.5.4 Events

All IO-Link events associated to the port are displayed here.

8.4.5.5 Identification

Here are the device identification and IO-Link capability details.

8.4.5.6 Data storage

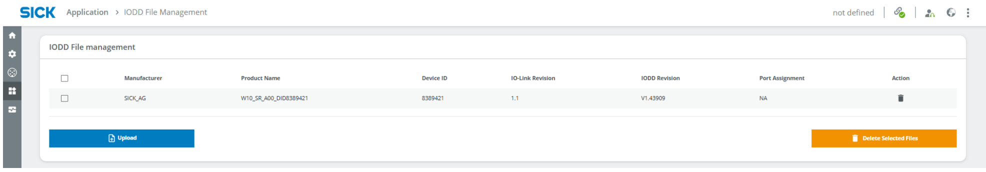
Here you have the option to create a backup file of the connected IO-Link device (V1.1) or transferring an existing backup.

This feature is only available if Validation and Backup is set for this port to "V1.1 Backup and Restore" or "V1.1 Restore" (see "Use case “Backup and Restore” ", page 68, see "Use case “Restore” ", page 69).

8.4.6 Application

8.4.6.1 IODD File Management

The IODD (IO-Link Device Description) File Management page is a centralized interface designed to manage IODD files efficiently. This page allows users to upload, delete, and view detailed information about IODD files, ensuring seamless integration and management of device descriptions.



□	Manufacturer	Product Name	Device ID	IO-Link Revision	IODD Revision	Port Assignment	Action
□	SICK_AG	W10_SR_A00_DID8389421	8389421	1.1	V1.43909	NA	

Figure 11: IODD file management layout

8.4.6.2 Logic Editor

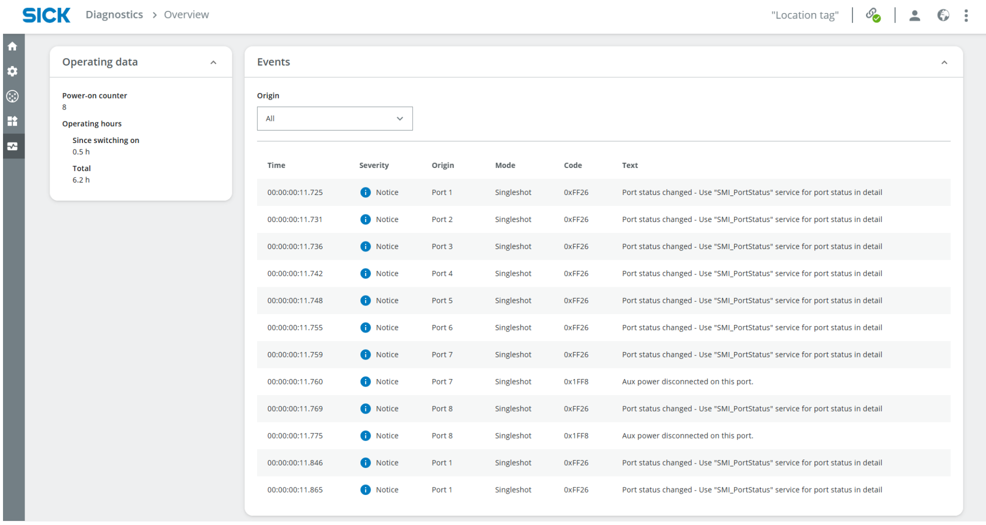
The logic editor of SIG300 is a key function allowing you to realize dedicated applications within the device by utilizing connected sensors or actuators.

For Detailed information and user manuals: see "Logic Editor ", page 70

8.4.7 Diagnostics

8.4.7.1 Overview

The Diagnostic Overview page is a comprehensive interface designed to monitor and analyze the operating data of your system. This page provides detailed insights into the performance and events of gateway master ports and connected devices, ensuring optimal operation and quick troubleshooting.



Power-on counter
8
Operating hours
Since switching on
0.5 h
Total
6.2 h

Origin: All

Time	Severity	Origin	Mode	Code	Text
00:00:00:11.725	Notice	Port 1	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.731	Notice	Port 2	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.736	Notice	Port 3	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.742	Notice	Port 4	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.748	Notice	Port 5	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.755	Notice	Port 6	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.759	Notice	Port 7	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.760	Notice	Port 7	Singleshot	0x1FF8	Aux power disconnected on this port.
00:00:00:11.769	Notice	Port 8	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.775	Notice	Port 8	Singleshot	0x1FF8	Aux power disconnected on this port.
00:00:00:11.846	Notice	Port 1	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail
00:00:00:11.865	Notice	Port 1	Singleshot	0xFF26	Port status changed - Use "SMI_PortStatus" service for port status in detail

Figure 12: Diagnostic Overview Page

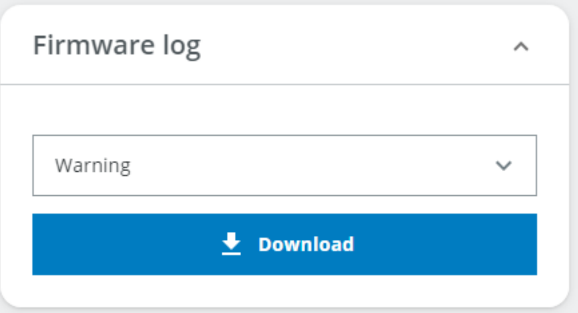
An event is structured as follows:

Time	**shows the time from last reboot of the device.**
Severity	IO-Link specification defines the security (Error, Warning, Notification) of the event.

Time	shows the time from last reboot of the device.
Mode	shows the types of the events. Possible cases: • Singleshot • Appears • Disappears
Code	is a numerical code defined by the IO-Link specification or vendor specific

8.4.7.1.1 Logging

SIG300 automatically saves specific events (“Warnings” and “Fatal Errors”) in a “log” file. The Firmware Log tile is available, and it is shown when user is logged as “Service” level. The user can set different logging filter levels to register more event types and download the “log” file from the device.
Firmware logging may affect device performance and should be used only at the direction of Sick support.



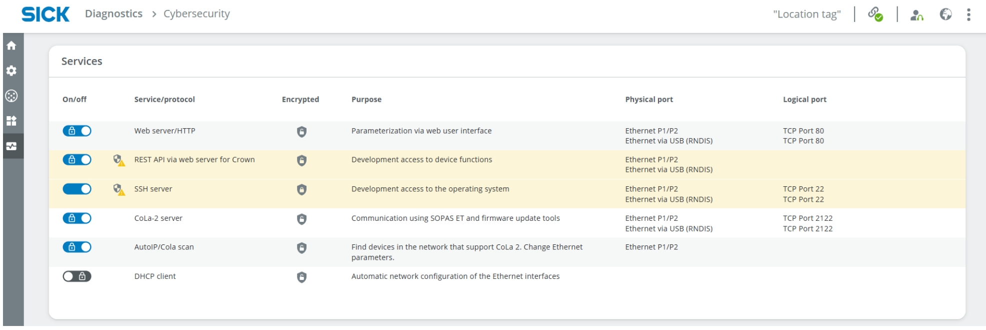
Firmware log

Warning

Download

8.4.7.2 Cybersecurity

The page lists various network services that are running, along with details about their encryption status, purpose, physical and logical ports.

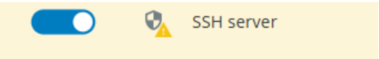
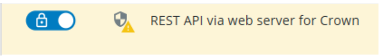


SICK Diagnostics > Cybersecurity "Location tag"

Services

On/off	Service/protocol	Encrypted	Purpose	Physical port	Logical port
	Web server/HTTP		Parameterization via web user interface	Ethernet P1/P2 Ethernet via USB (RNDIS)	TCP Port 80 TCP Port 80
	REST API via web server for Crown		Development access to device functions	Ethernet P1/P2 Ethernet via USB (RNDIS)	
	SSH server		Development access to the operating system	Ethernet P1/P2 Ethernet via USB (RNDIS)	TCP Port 22 TCP Port 22
	CoLa-2 server		Communication using SOPAS ET and firmware update tools	Ethernet P1/P2 Ethernet via USB (RNDIS)	TCP Port 2122 TCP Port 2122
	AutoIP/Cola scan		Find devices in the network that support CoLa 2. Change Ethernet parameters.	Ethernet P1/P2	
	DHCP client		Automatic network configuration of the Ethernet interfaces		

Figure 13: Cybersecurity

 SSH server	Services Highlighted with yellow indicates security concerns
 REST API via web server for Crown	The lock Symbol indicates that this setting cannot be changed

Please Note that changing the Cybersecurity Settings requires a service level user.

8.5 USB connection

It is possible to get a connection between a laptop and SIG300 via USB-C port and filling the address “http://169.254.0.1” in a standard web browser.

NOTE
SIG300 cannot be powered via the USB-C port and has to be separately powered via the power connector.

9 Device functions

9.1 Overview of the device functions

This section specifies the available functions at each parameterization interface (i. e. SOPAS ET, REST API and/or web server).

NOTE
see "Available variables and methods", page 45

Function	Web server/ SOPAS	REST API/MQTT
Display process data	Available see "Web-Interface", page 54	Available REST: see "Available variables and methods", page 45 MQTT: see "General description", page 52
Application version		[SIG300IP address]/api/ApplicationVersion
AppEngine version		[SIG300IP address]/api/AppEngineVersion
Switch on counting	Available	[SIG300IP]/api/PowerOnCnt
Operating instructions	Available	[SIG300IP address]/api/OpHours
Hours since last start	Available	[SIG300IP address]/api/DailyOpHours

9.2 IO-Link Data storage

The whole parameter set of the device, e. g. switching point, additional logic or teach-in settings of IO-Link sensors or actuators can be stored centralized in the SIG300.

The Data Storage feature brings major advantages when it comes to easy replacement of IO-Link sensor or actuator due to defects.

In case is established a connection with a replaced compatible IO-Link sensor or actuator, the stored parameter set is transferred from SIG300 to the device. The device behaves like the device to be replaced.

There are two different use cases how to utilize this mechanism:

- Backup and Restore
- Restore

9.2.1 Use case “Backup and Restore”

Parameters are read and written in both directions, from the IO-Link master to the device and vice versa.

The configurations defined in IO-Link connected device and saved in SIG300 IO-Link master are always synchronized independently if changes in configuration are done at sensor side (e. g.: via display) or via IO-Link master interface (e. g: SOPAS ET, web server or Asset management software).

“Backup and Restore” is mostly used for commissioning. For instance, if it is performed/triggered a “teach-in” operation on the connected IO-link device, the modification in its configuration is automatically uploaded and stored in the data storage object within the SIG300.

The use case supports also device replacement, e.g. the configuration will be automatically copied to the new device, if one needs to be replaced. In this case the IO-Link device configuration stored on SIG300 has the priority and overwrite the configuration in the new connected device.

Example

The “Data Storage” function of the SIG300 IO-Link Master enables to create a backup of the configuration of single IO-Link connected devices.

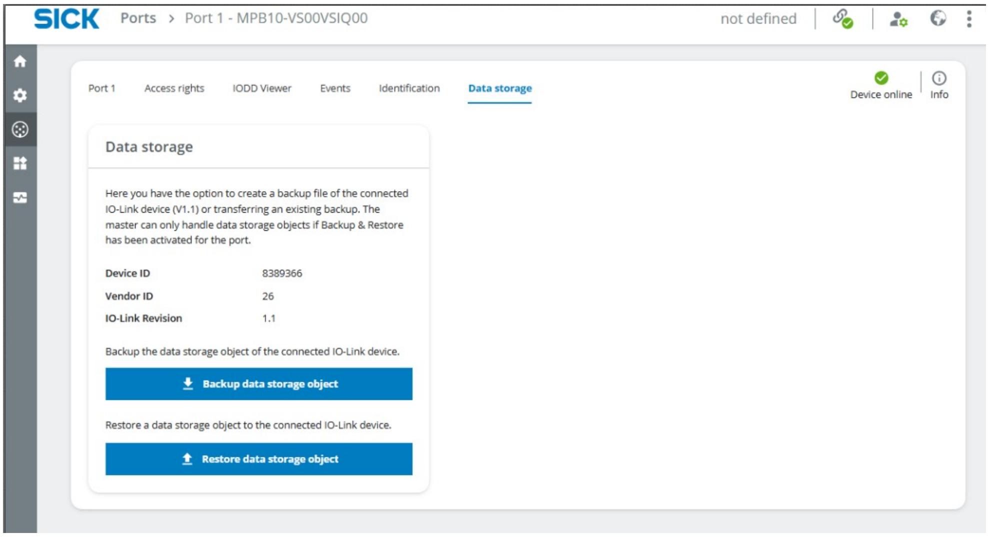
The following example shows step-by-step how to perform a “Backup” so that in case of replacement device, it can be automatically parameterized according to the original device.

1. Select “Ports” in navigation menu.
2. Select “IO-Link” in “Pin 4 mode”.
3. Select “Yes=manual” in “Device identification check”.
4. Select the option “Backup and Restore" in the “Validation and backup” drop down menu.

✓ The function automatically saves the sensor parameter data of the connected devices in the memory of IO-Link master.

NOTE
This setting is only available for pin 4, as the Data Storage function is restricted to IO-Link communication.

It is also possible to export the configuration of the connected IO-Link device using the command “Backup data storage object” (export from SIG300 to PC) in the “Data storage” tab.



SICK Ports > Port 1 - MPB10-VS00VSIQ00 not defined

Port 1 | Access rights | IODD Viewer | Events | Identification | **Data storage** | Device online | Info

Data storage

Here you have the option to create a backup file of the connected IO-Link device (V1.1) or transferring an existing backup. The master can only handle data storage objects if Backup & Restore has been activated for the port.

Device ID	8389366
Vendor ID	26
IO-Link Revision	1.1

Backup the data storage object of the connected IO-Link device.

Backup data storage object

Restore a data storage object to the connected IO-Link device.

Restore data storage object

The configuration export data is in JSON format.

9.2.2 Use case “Restore”

In this use case a configuration change to the IO-Link device directly will be ignored. To use the “Restore” option a former configuration must be available. Replacement of broken devices is also possible.

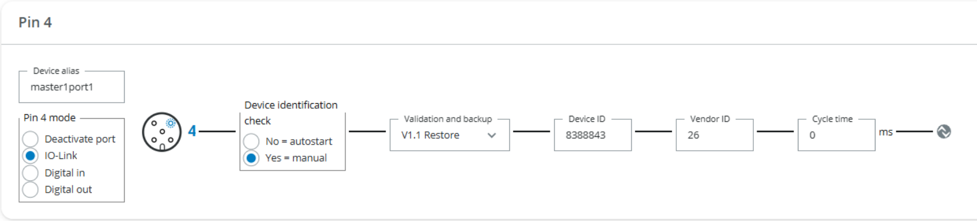
However, this function only works if the devices are compatible with each other. For this reason, the expected "Device ID" and expected "Vendor ID" must also be specified.

Example

The “Data Storage” function of the SIG300 IO-Link Master enables easy replacement of defective IO-Link sensors. The following example shows step-by-step how the SIG300 can be used to commission a new IO-Link device so that a replacement device is automatically parameterized according to the original device.

To replace a sensor/actuator with “Restore” function a former “Backup” of the sensor/actuator configuration must be available on SIG300 or as JSON format file.

1. Select “Ports” in navigation menu.
2. Select “IO-Link” in “Pin 4 mode” .
3. Select “Yes=manual” in “Device identification check”.
4. Select the option “Restore" in the “Validation and backup” drop down menu.



Pin 4

Device alias: master1port1

Pin 4 mode
○ Deactivate port
● IO-Link
○ Digital in
○ Digital out

Device identification check
○ No = autostart
● Yes = manual

Validation and backup: V1.1 Restore

Device ID: 8388843

Vendor ID: 26

Cycle time: 0 ms

- “Device ID” and “Vendor ID” of the new sensor/actuator must correspond to the data of the replaced device.
  This data is validated by the system and a proper warning message is shown.
- In case the “backup” file is saved on a PC, it is also possible to download the configuration (from “PC” to SIG300) using the proper command in in “Data storage” layer.
  “Restore” or “Backup and Restore” function must be activated to load the new configuration to the connected device.

The configuration file data format is JSON.

9.3 Logic Editor

The Logic Editor of SIG300 is a key function allowing you to realize dedicated applications within the device by utilizing connected sensors or actuators.

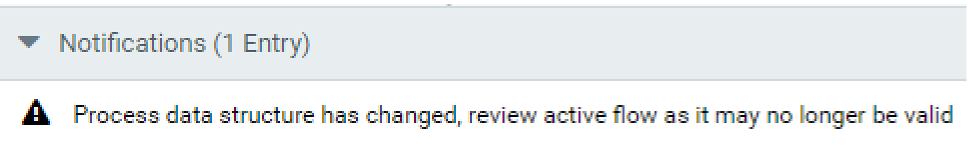
NOTE
The drag & drop Logic Editor configuration is not accessible via the fieldbus or the REST API.

In Logic Editor only process data can be used as input or output values.

The Logic Editor can use all available signal **inputs** as sources for the logic application:

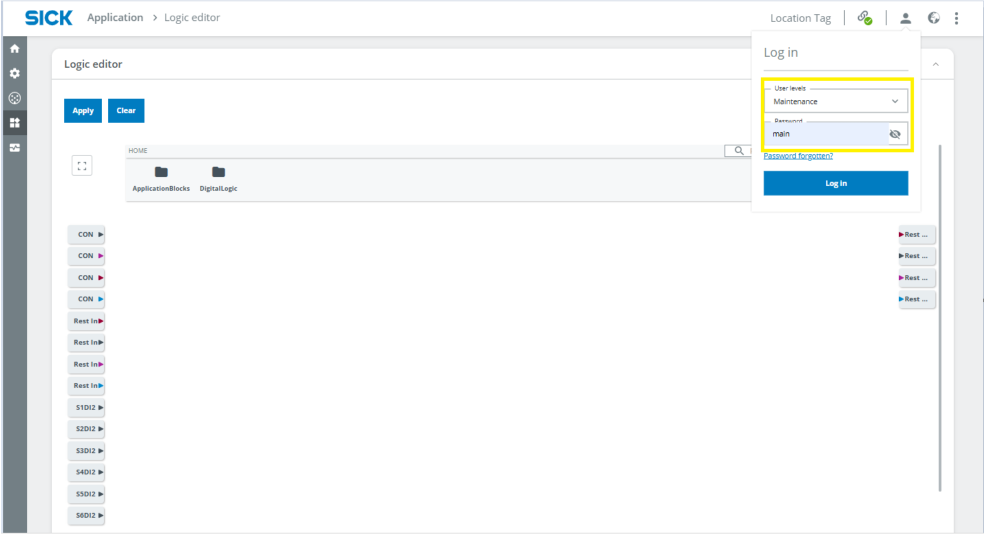
- All IO-Link port pins configured as “Digital Input”
- IO-Link Process Data Input from all “Sx” port pins 4 configured to IO-Link mode (Port S1 ... S8)
- REST API Input values

NOTE
It is necessary to upload and assign the IODDs of the devices in order to be used in the Logic Editor.



Notifications (1 Entry)

⚠ Process data structure has changed, review active flow as it may no longer be valid



User levels: Maintenance

Password: main

Figure 14: Editing Mode

1. To start your configuration change the “user level” from Run (read-only) to Maintenance: select “user levels” icon on the top right and click on the “Change user level” button, after that select “Maintenance” as “user levels”.
2. The default login password for the maintenance mode is: main.
3. Drag and drop the desired function block connecting inputs and outputs. A colored arrow allows to recognize the valid data format for the selected function block.

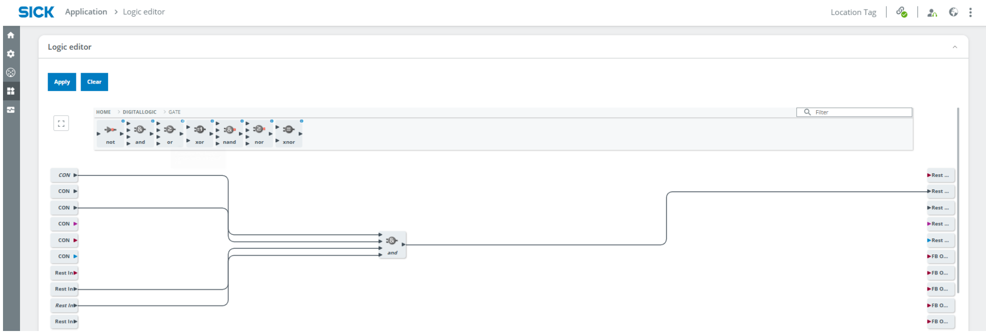


Figure 15: Editing Mode

Overview

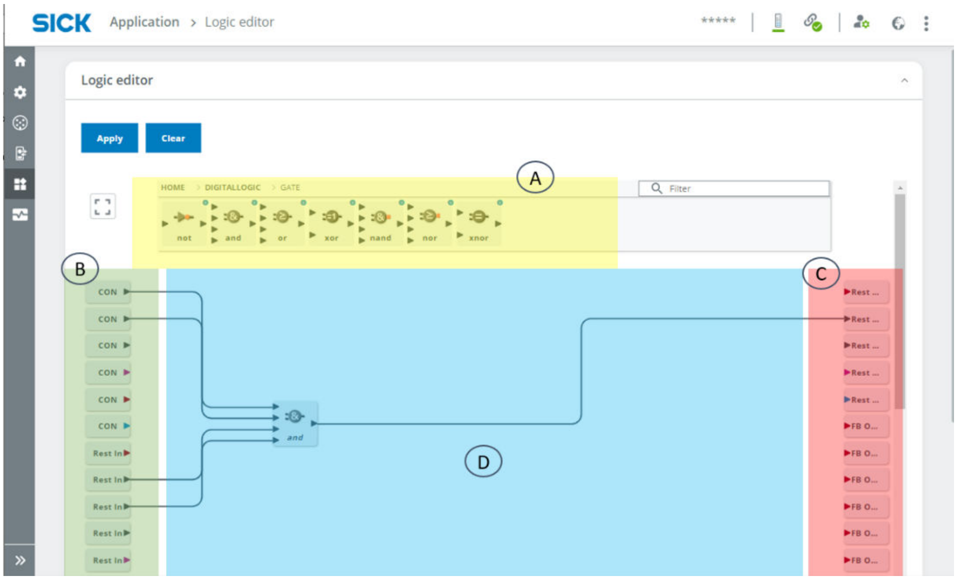
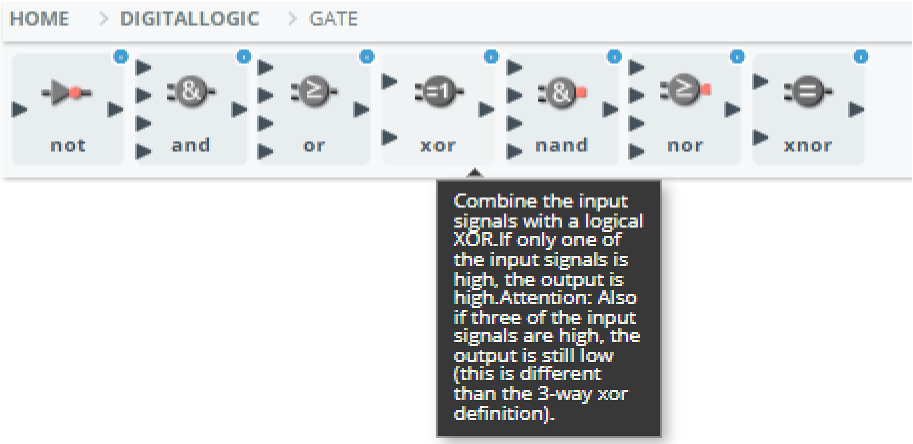


Figure 16: Logic editor screen

A Logic blocks organized in different folders
B Inputs
C Outputs
D Workspace

Placing the mouse over the logic blocks, it shows a tooltip with a short description of the function.

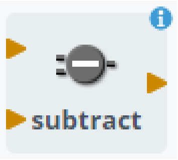


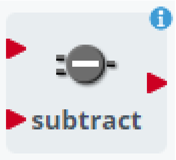
Combine the input signals with a logical XOR.If only one of the input signals is high, the output is high.Attention: Also if three of the input signals are high, the output is still low (this is different than the 3-way xor definition).

Figure 17: Function block detailed information

Within the logic function in the top bar there are some functions mentioned twice. One time with red triangles (integer) and one time with orange triangles (float). So, the logic function is the same, but the data types which can be used are different.

Example:





- Use drag & drop to select the desired logic block and put it into the workspace.
- To delete logic blocks put them back in the upper area via drag & drop.
- The maximum amount of logic blocks which can be used in the logic editor in parallel is 500 blocks.

NOTE

The input and output blocks can be moved to the workspace to achieve a better routing and overview.

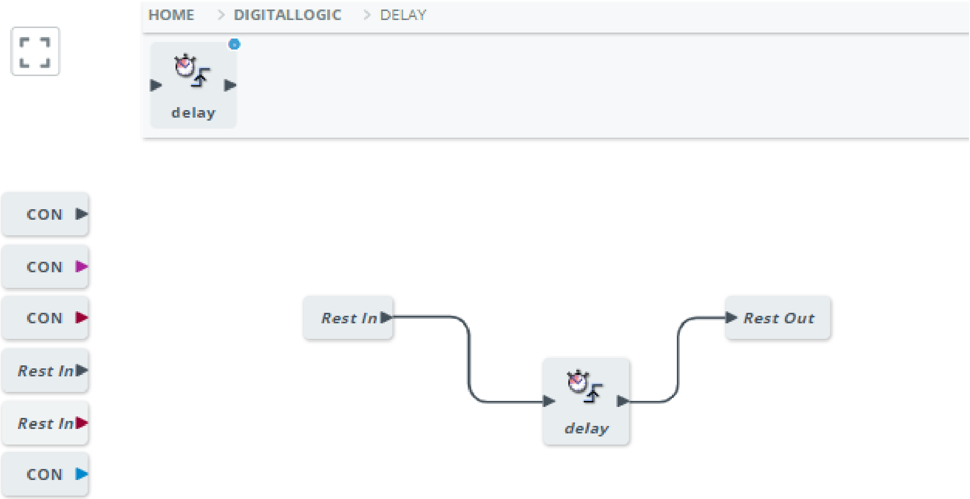


Figure 18: Input and Output blocks can be moved inside the workspace

Connect your logic blocks with drag & drop with the inputs and outputs. First click on the triangle on the input, hold the line and connect it to a triangle of the logic block.

Please note to always use the upper inputs first, starting at A, then B, then C. In case you use only two inputs please use always the top two inputs A+B and not e. g. B+D.

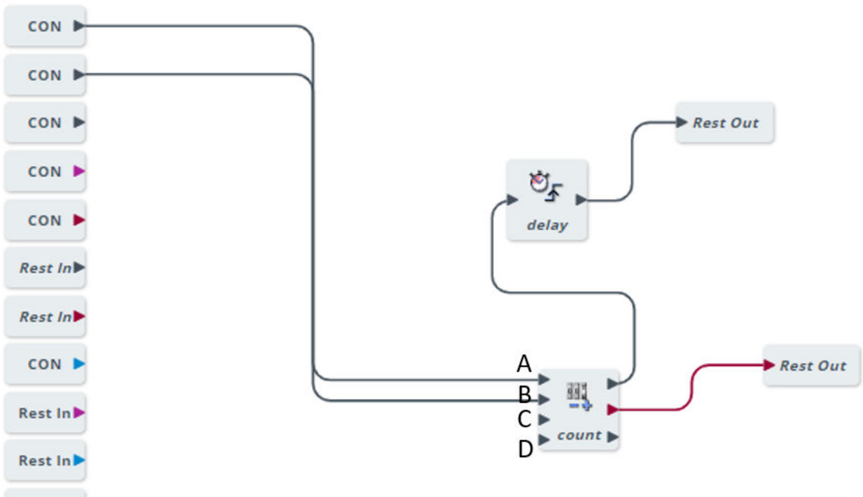


Figure 19: Possible connections

Please note whether the values are Integer or Boolean: it is only possible to connect Integer with Integer and Boolean with Boolean.

Table 35: Data type and size

Arrow color	Data type
CON ▶	Boolean (1-bit)
CON ▶	Integer (64-bit)
CON ▶	String (dynamically allocated)
CON ▶	Float (64-bit)

By clicking on logic block you get information about the possible connections to this individual block.

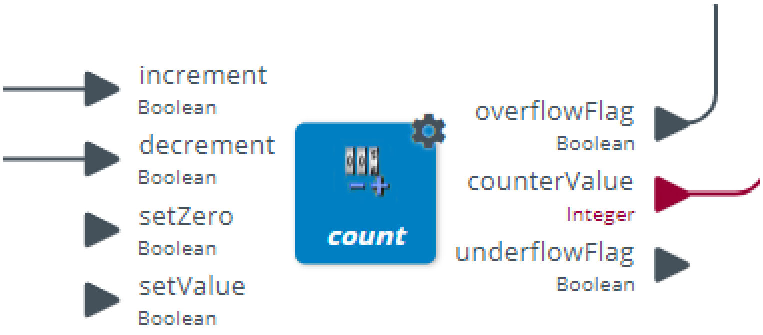


Figure 20: Clicking on a function block in workspace shows the valid data types

It is possible to connect several inputs and outputs using logic blocks to create a customized application.

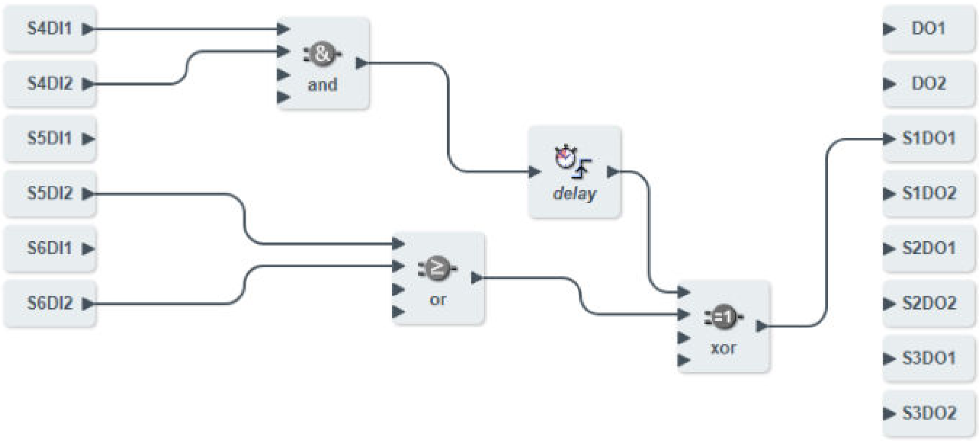
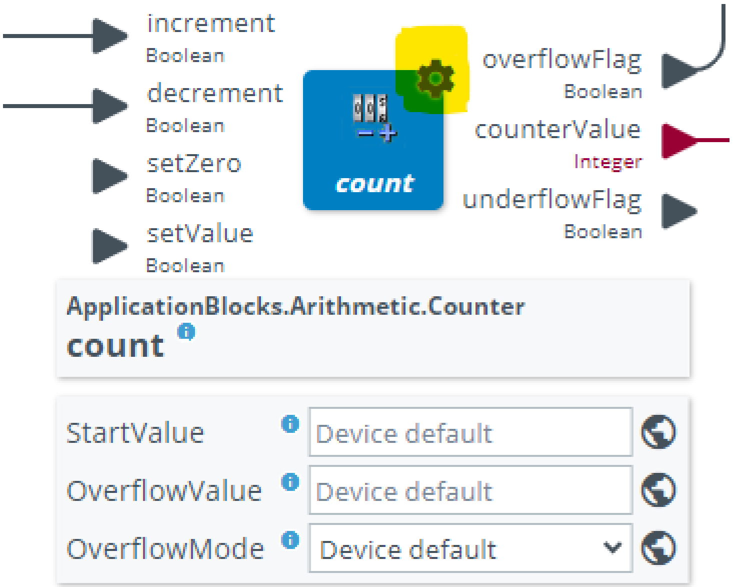


Figure 21: Example of a logic block combination using only Boolean values

Clicking on the “Gear” icon on function block (Input and Output block as well), it is possible to configure specific parameters and values.

Depending by function block data type, the related value range is limited.



ApplicationBlocks.Arithmetic.Counter
count

StartValue	Device default
OverflowValue	Device default
OverflowMode	Device default

Figure 22: Function block parametrization

NOTE
Not all logic blocks are adjustable.

To remove a connection or a function block, click on your desired connection and put it in into the garbage bin on the upper area via drag & drop.

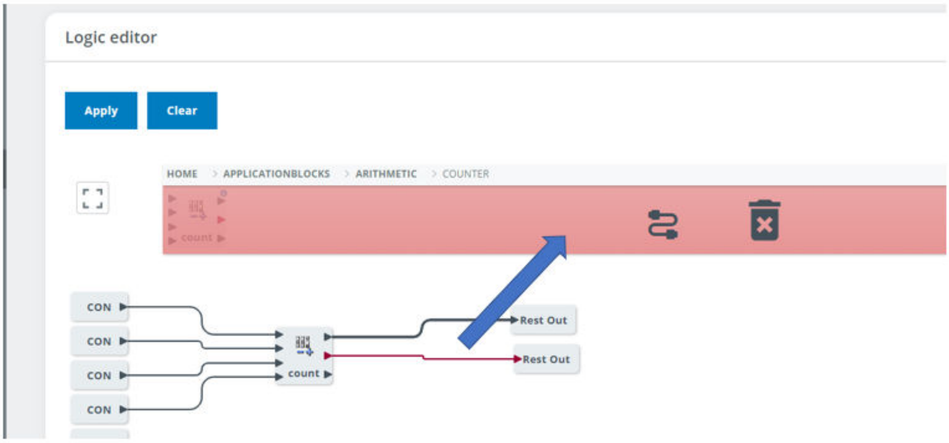


Figure 23: Delete a connection.

Apply new logic to the SIG300

► Press “Apply” to transfer/download and to execute the logic.

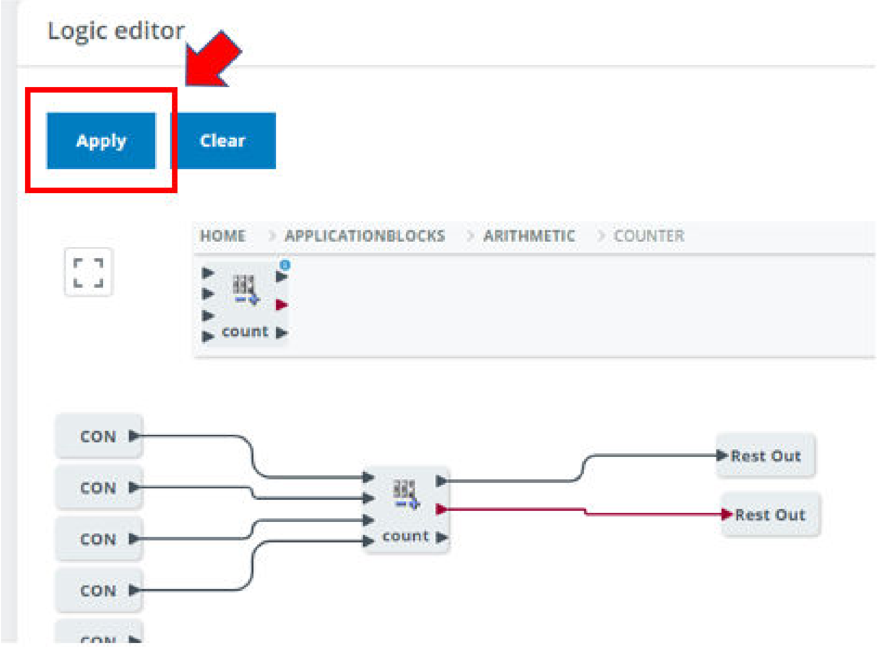
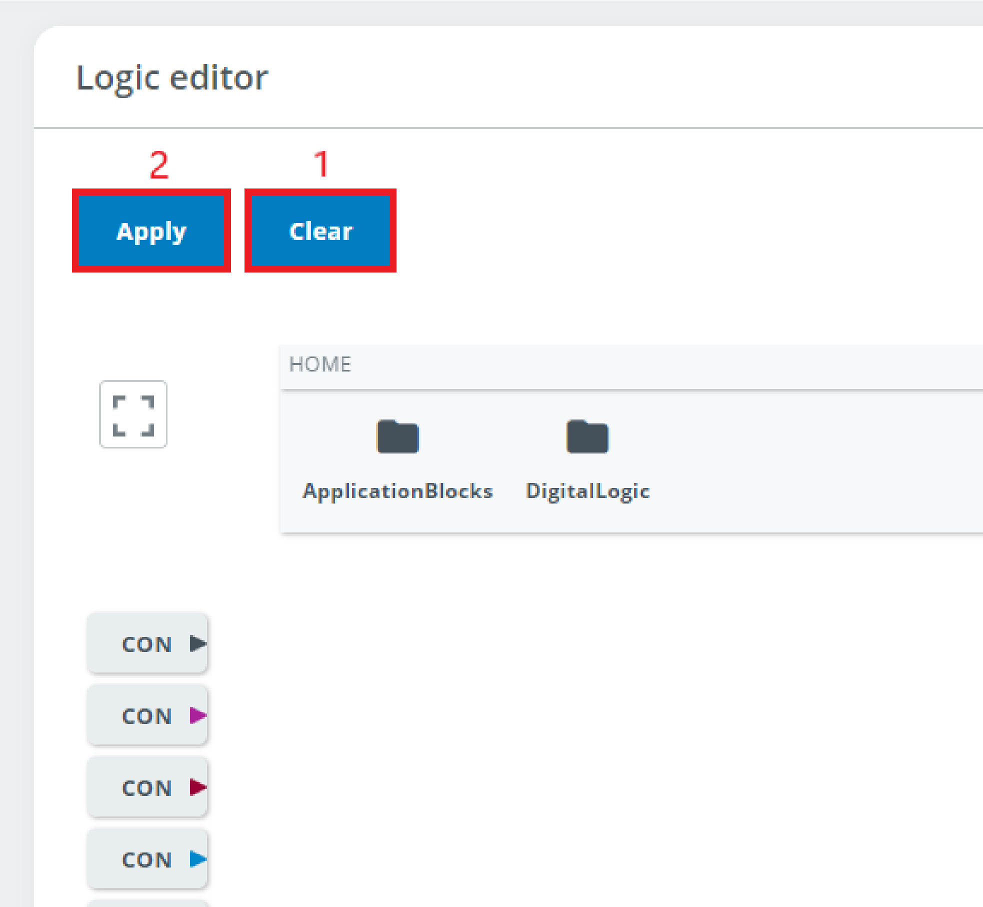


Figure 24: Transfer the logic to SIG300

Delete logic

1. Click on the “Clear” button, to clear the logic.



9.3.1 Logic editor inputs, outputs and logic blocks

The logic editor visualizes the process data according to the IODD of the IO-Link device. Inputs are displayed on the left side, outputs are visualized on the right side of the logic editor workspace.

Example: If you connect an inductive proximity sensor IMC on port S1 of SIG300, the input side looks like this:

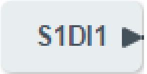


Figure 25: Example of an input logic block: [port number: S1 ... S8] [Digital Input DI or Digital Output DO] [pin number]

To display IODD variable names in the logic editor:

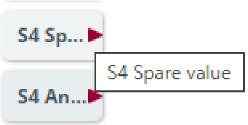
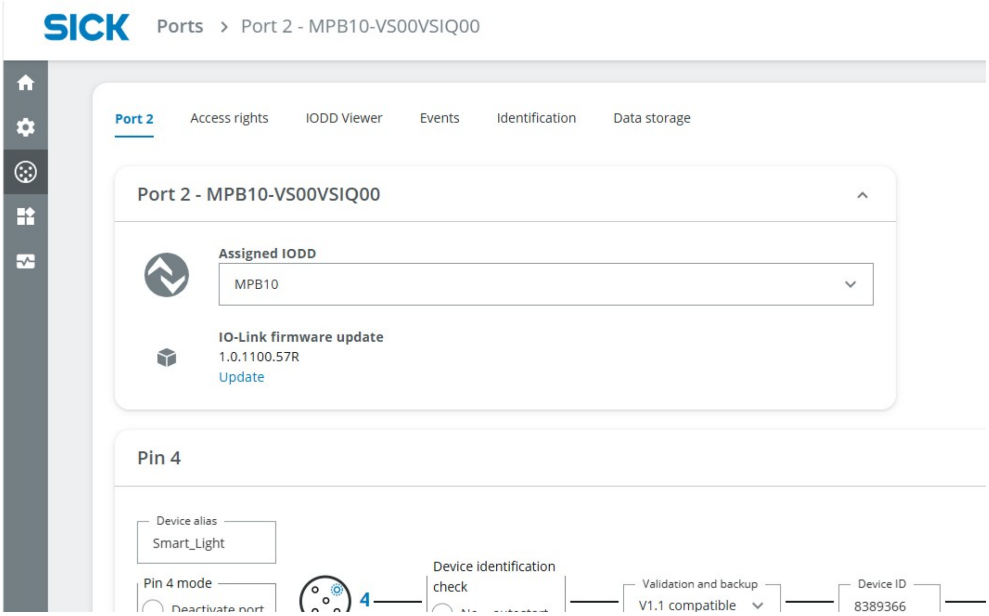


Figure 26: IODD variable names in logic editor

1. Load device IODD file in the “Application/IODD File Management” page.
2. Open the port settings related to the connected device (e. g: Port4) and select:
   - “PIN4 mode” as “IO-Link”
   - “Device identification check” to “Yes = manual”
   - “Validation and Backup”: to “Device V1.0” OR “Device V,1.1” OR “Backup and Restore” OR “Restore”

3. Assign the correct IODD file name to the port.


SICK Ports > Port 2 - MPB10-VS00VSIQ00

Port 2 | Access rights | IODD Viewer | Events | Identification | Data storage

Port 2 - MPB10-VS00VSIQ00

Assigned IODD: MPB10

IO-Link firmware update
1.0.1100.57R
Update

Pin 4

Device alias: Smart_Light

Pin 4 mode: ○ Deactivate port

NOTE

- Last valid process data value is provided in case of an IO-Link connection loss to the connected device.
- The processing of process data in the logic editor is not consistently clocked. This is due to potential delays in output process data resulting from factors such as increased network load.
- If the port setting is changed, the logic configuration might not be valid any more.

Logic editor Digital Inputs

Digital input blocks are located on the left of the working area. They represent the physical digital inputs delivered by connected device or incoming fieldbus parameter values.

Their label is built as follow:

[port number: S1…S8][Digital Input DI or Digital Output DO][pin number]

e. g.: “S1DI2” is the digital input (“DI”) of “pin 2” related to the device connected to port “S1”

Depending of port type and its configuration, pin 2, pin 4, pin 5 can be assigned as digital input (see "Pin assignment", page 21)

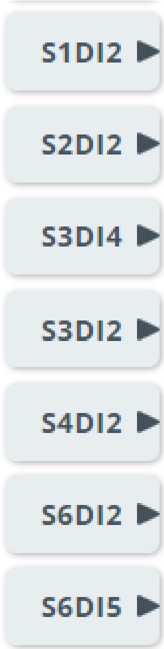
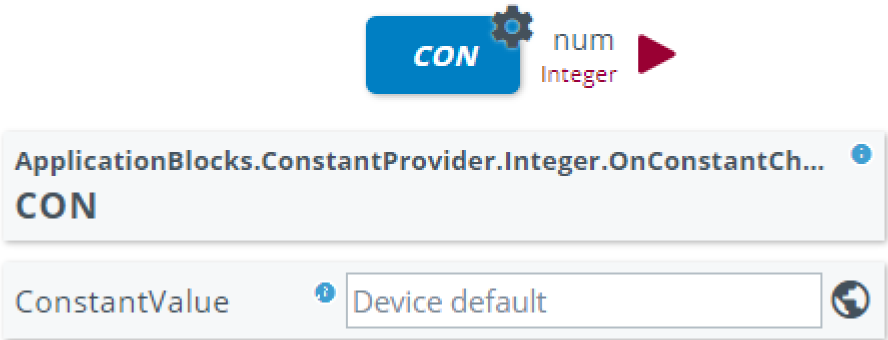


Figure 27: Digital input blocks – e. g.: port 3 provides two digital inputs: one on pin 2 and another one on pin 4

Logic editor Inputs / Constant:

The constant blocks can be set to a fixed value to be used for further processing. There are four types of constant block: Boolean, Float, Integer and String.



ApplicationBlocks.ConstantProvider.Integer.OnConstantCh...
CON

ConstantValue: Device default

Figure 28: Setting of an “integer” type constant block

Logic editor Inputs / via Rest:

It is possible to set an input value via REST to be processed by the logic configuration of the SIG300. This input will be visualized with "Rest In" on the logic editor page.

The value for a specified offset can be set by sending a POST request with the value formatted in JSON to:

```
[SIG300 IPaddress]/ac/restblocks/v1/flow/inputs/{offset number}/
processdata/value
```

The JSON containing the value should be structured like this:

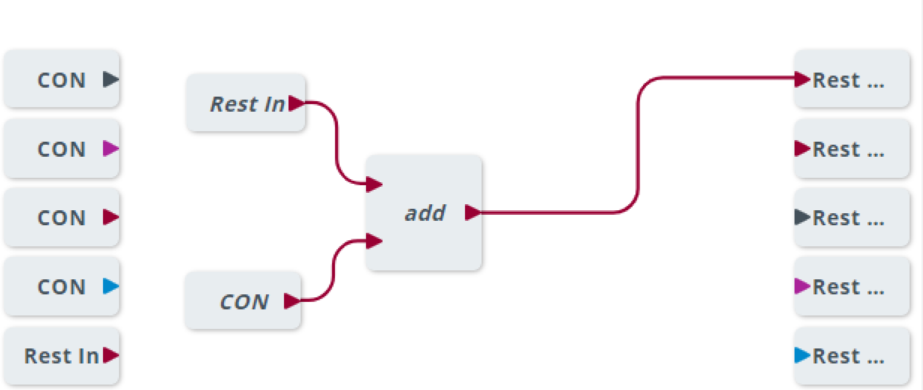
```
{
"value": [value to be set]
}
```

The offset number is a unique number associated at the REST block. It starts from “1” and it is incremented by one for each new REST block.
Be sure to have inserted a REST block in your flow and saved it in the device to use the “offset”.

For more Flow editor REST commands see: see table 32, page 50

Example to test REST block assignment using Insomnia software.

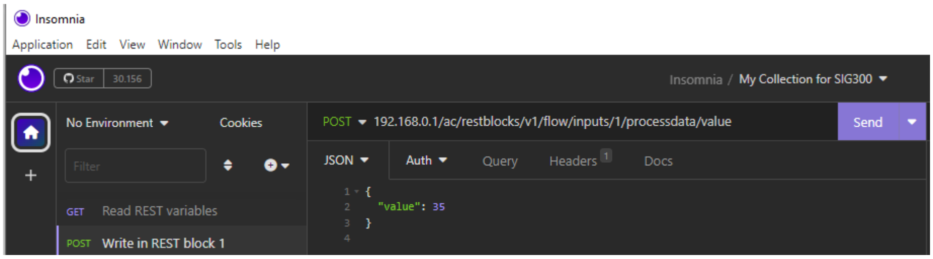
1. Create a simple flow like picture below.



2. Click “Apply” to save it in the SIG300
   Use “Insomnia” to deliver a value to the REST block.

.../ac/restblocks/v1/flow/inputs/{offset number}/processdata/value	GET	Read an input value for a specified offset
.../ac/restblocks/v1/flow/inputs/{offset number}/processdata/value	POST	Write Rest block data to the specified offset

POST: “192.168.0.1/ac/restblocks/v1/flow/inputs/1/processdata/value”
(Write in the “JSON” part, the text structure as shown in the picture below)
The offset is a chronological number generated when you added an input REST block in the flow (this number is not shown on the UI).
There are dedicated offset for REST input blocks and offset for REST output blocks.



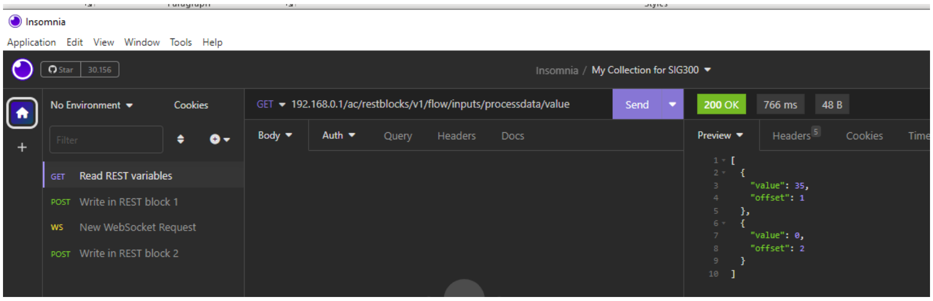
POST 192.168.0.1/ac/restblocks/v1/flow/inputs/1/processdata/value

```
{
  "value": 35
}
```

GET Read REST variables

POST Write in REST block 1

3. Use any software that can make REST requests (e. g. Insomnia) to read all input REST variable values stored in SIG300
   GET: “192.168.0.1/ac/restblocks/v1/flow/inputs/processdata/value” .

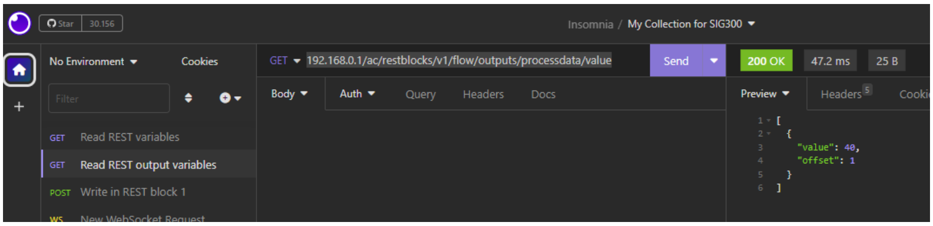


GET 192.168.0.1/ac/restblocks/v1/flow/inputs/processdata/value

200 OK 766 ms 48 B

```
[
  {
    "value": 35,
    "offset": 1
  },
  {
    "value": 0,
    "offset": 2
  }
]
```

To read all output REST variables values:
GET: “192.168.0.1/ac/restblocks/v1/flow/outputs/processdata/value”



GET 192.168.0.1/ac/restblocks/v1/flow/outputs/processdata/value

200 OK 47.2 ms 25 B

```
[
  {
    "value": 40,
    "offset": 1
  }
]
```

Logic editor Outputs / Digital:

Digital output blocks are located on the right of the working area. They represent the physical digital output and/or outcoming fieldbus parameter values.

Their label is built as follow:

[port number: S1...S8][Digital Input DI or Digital Output DO][pin number]

e. g.: “S1DO2” is the digital output (“DO”) of pin 2 related to the port S1

Depending on port type and its configuration, pin 2, pin 4, pin 5 can be assigned as digital input (see "Pin assignment", page 21)

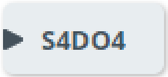
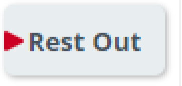


Figure 29: Digital output block for “pin 4” of port S4

NOTE
It is not possible to connect a digital output on pin 2 of port 7 and 8.

Logic editor Output / Rest:

Through the “Rest Out” block, data from the logic can be sent via REST interface to an upper system (e. g. HTTP Client).



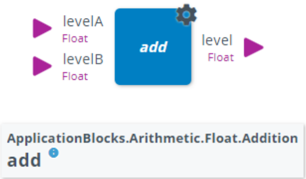
The value of a specified offset can be read by sending a GET request to:

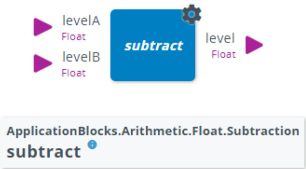
`[SIG300 IPaddress]/ac/restblocks/v1/flow/inputs/{offset value}/processdata/value`

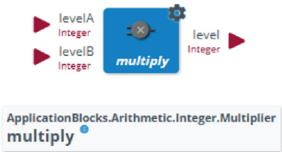
See also example for section **Logic editor Inputs / via Rest.**

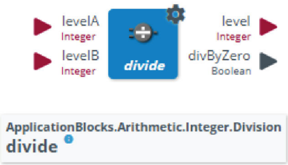
For more Flow editor REST commands: see "Available variables and methods", page 45

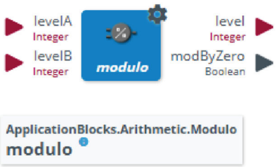
9.3.1.1 Application Blocks/Arithmetic

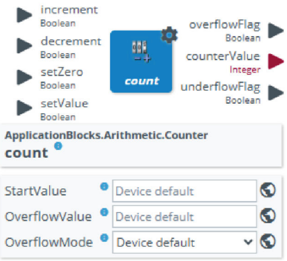
Visualization	Description	Addition of the two input values.
 levelA Float levelB Float add level Float ApplicationBlocks.Arithmetic.Float.Addition add	Number of inputs	2
	Input data type	Float, Integer
	Input description	num1: first input value num2: second input value
	Number of outputs	1
	Output data type	Identical to input data type
	Output description	result: result after addition of the two input values
	Settings	no settings available

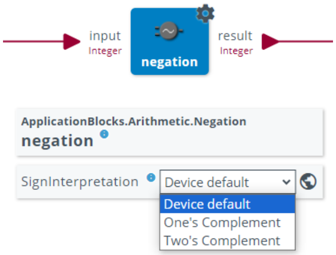
Visualization	Description	Subtraction of the two input values.
 levelA Float levelB Float subtract level Float ApplicationBlocks.Arithmetic.Float.Subtraction subtract	Number of inputs	2
	Input data type	Float, Integer
	Input description	num1: first input value num2: second input value
	Number of outputs	1
	Output data type	Identical to input data type
	Output description	result: result after subtraction of the two input values
	Settings	no settings available

Visualization	Description	Multiplication between the two input values.
 levelA Integer levelB Integer multiply level Integer ApplicationBlocks.Arithmetic.Integer.Multiplier multiply	Number of inputs	2
	Input data type	Float, Integer
	Input description	num1: first input value num2: second input value
	Number of outputs	1
	Output data type	Identical to input data type
	Output description	result: result after multiplying the two input values
	Settings	no settings available

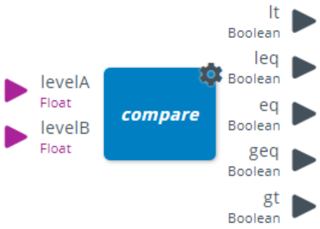
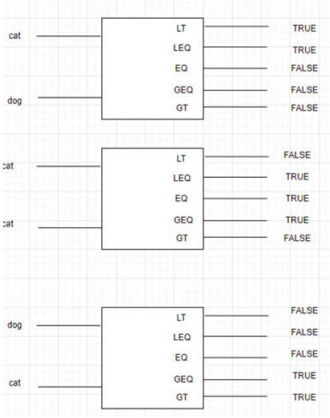
Visualization	Description	Division between the two input values.
 levelA Integer levelB Integer divide level Integer divByZero Boolean ApplicationBlocks.Arithmetic.Integer.Division divide	Number of inputs	2
	Input data type	Float, Integer
	Input description	num1: first input value num2: second input value
	Number of outputs	2
	Output data type	Output 1: Identical to input data type Output 2: 1-bit
	Output description	Output1: Result after dividing the two input values Output2: divByZero: In case num2 is null, this output is set to “1”
	Settings	no settings available

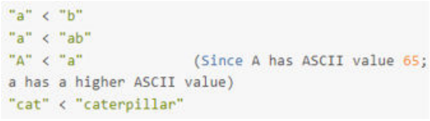
Visualization	Description	Modulo operation between the two input values.
 levelA Integer levelB Integer modulo level Integer modByZero Boolean ApplicationBlocks.Arithmetic.Modulo modulo	Number of inputs	2
	Input data type	Integer
	Input description	num1: first input value num2: second input value
	Number of outputs	2
	Output data type	Output 1: Identical to input data type Output 2: 1-bit
	Output description	Output1: Remainder after dividing the two input values Output2: divByZero: In case num2 is null, this output is set to “1”
	Settings	no settings available

Visualization	Description	**Event counter for digital signals.** **Maximum switching frequency (e. g. for a NOT gate): 35 Hz** **Maximum switching frequency for the Counter: 90 Hz.**
 increment Boolean decrement Boolean setZero Boolean setValue Boolean count overflowFlag Boolean counterValue Integer underflowFlag Boolean ApplicationBlocks.Arithmetic.Counter count StartValue: Device default OverflowValue: Device default OverflowMode: Device default	Number of inputs	4
	Input data type	Input 1 ("Up"): 1-bit Input 2 ("Down"): 1-bit Input 3 ("Reset to 0"): 1-bit Input 4 ("Set to start value"): 1-bit
	Input description	increment: value will be counted up decrement: value will be counted down setZero: set counter to zero setValue: set counter to StartValue
	Number of outputs	3
	Output data type	Output 1 ("Overflow"): 1-bit Output 2 ("Counter value"): 32-bit Output 3 ("Underflow"): 1-bit
	Output description	Output 1 overflowFlag: Bit is set if the counter value exceeds the over-flow value Output 2 counterValue: Current counter value. Counter values are NOT stored by a power cycle. Output 3 underflowFlag: Bit is set if the value is below the overflow value. The default overflow value is 4,294,967,295.
	Settings	**StartValue**: Counter value that is set when “setValue” is trig- gered (default: 0) **OverflowValue**: Maximum value of the counter output (default: 4,294,967,295) **OverflowMode**: Behavior of the counter value in the event of an underflow or overflow AUTO: After reaching the overflow value, the counter is auto-matically reset to the defined start value. MANU: After reaching the overflow value, the counter value can only be reset manually by the "setZero" or "setValue" signal. Additional information: When the maximum counter value (overflow value) is reached, the overflow output is set to "High". However, there is a difference between the automatic and manual modes. The automatic mode the value will be set to 0 on next rising edge of the increment input and of course the counter value can be changed by the setZero or setValue input. In the manual mode, the countervalue will stay on the over-flowvalue until a rising edge on the decrement, setZero or setValue input is detected. The default value for the counter start is 0, but it can be set to any value within the range (32 bits).

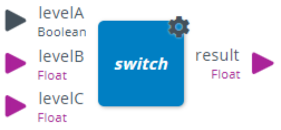
Visualization	Description	Negation of the input value either one´s or two´s complement depending on the configuration.
 ApplicationBlocks.Arithmetic.Negation negation SignInterpretation: Device default Device default One's Complement Two's Complement	Number of inputs	1
	Input data type	Signed Integer
	Input description	Input value
	Number of outputs	1
	Output data type	Output 1: Identical to input data type
	Output description	result: The “one's” or “two's complement” of the input value. (So the output value is the opposite of the input value).
	Settings	Selection of the one's or two's complement (Default Two's complement)

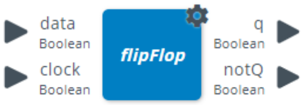
9.3.1.2 Application Blocks/Conditional

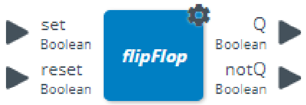
Visualization	Description	Compares
 levelA Float levelB Float compare lt Boolean leq Boolean eq Boolean geq Boolean gt Boolean Output examples table for “String comparison”.  cat dog LT LEQ EQ GEQ GT TRUE TRUE FALSE FALSE FALSE cat cat LT LEQ EQ GEQ GT FALSE TRUE TRUE TRUE FALSE dog cat LT LEQ EQ GEQ GT FALSE FALSE FALSE TRUE TRUE	Number of inputs	2
	Input data type	Float, Integer, String (String? -> converted in number?)
	Input description	num1: first input value num2: second input value
	Number of outputs	1...5
	Output data type	Output 1 ("<"): 1-bit Output 2 ("≤"): 1-bit Output 3 ("="): 1-bit Output 4 ("≥"): 1-bit Output 5 (">"): 1-bit
	Output description (Float, Integer)	lt: < input 1 is less than input 2 leq: ≤ input 1 is less or equal to input 2 eq: = input 1 is equal to input 2 geq: ≥ input 1 is greater or equal to input 2 gt: > input 1 is greater than input 2

	Output description (String)	The less-than operator on strings does a lexicographical comparison on the strings. This compares strings in the same way that they would be listed in dictionary order, generalized to work for strings with non-letter characters.
		For “output” behavior, see the “behaviour” table.
		Example:
		
		`"a" < "b"`
		`"a" < "ab"`
		`"A" < "a"             (Since A has ASCII value 65;`
		`a has a higher ASCII value)`
		`"cat" < "caterpillar"`

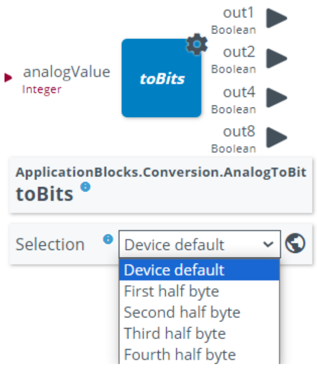
	Settings	No settings available

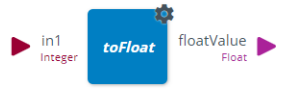
Visualization	Description	Selection between two input values depending on the boolean input.
 levelA Boolean levelB Float levelC Float switch result Float	Number of inputs	3
	Input data type	Float, Integer, String & Boolean Input 1 ("If"): 1-bit Input 2 ("Then"): Float, Integer, String Input 3 ("Else"): Float, Integer, String
	Input description	Input 1: “IF” trigger condition (True) Input 2: Value for the output if the “Input 1” is “true” Input 3: Value for the output if the “Input 1” is “false”
	Number of outputs	1
	Output data type	Integer
	Output description	results: see Input description
	Settings	No settings available

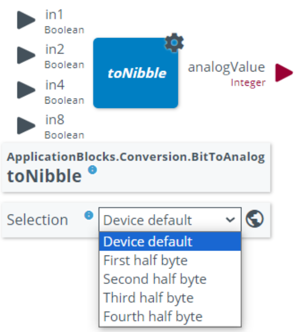
Visualization	Description	Clocked (rising edge) D-Flip Flop.
 data Boolean clock Boolean flipFlop q Boolean notQ Boolean	Number of inputs	2
	Input data type	Input 1 ("data"): 1-bit Input 2 ("clock"): 1-bit
	Input description	data: State of this input to be transferred to output on rising edge. clock: Rising edge of this input triggers the capture of the data input.
	Number of outputs	2
	Output data type	Output 1 ("Q"): 1-bit Output 2 ("notQ"): 1-bit
	Output description	q: Set when data input is high and a rising egde occurs on the clock input. Reset when data input is low and a rising edge occurs on the clock input. notQ: Inverted signal of output Q.
	Settings	No settings available

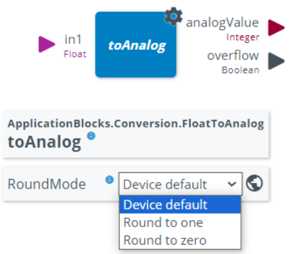
Visualization	Description	Basic RS-Flip Flop functionality.
 set Boolean reset Boolean flipFlop Q Boolean notQ Boolean	Number of inputs	2
	Input data type	Input 1 ("Set"): 1-bit Input 2 ("Reset"): 1-bit
	Input description	set: See above truth table description reset: See above truth table description
	Number of outputs	2
	Output data type	Output 1 ("Q"): 1-bit Output 2 ("notQ"): 1-bit
	Output description	Q: if (set == false and reset == false) then Q = Keeps it's last value elseif (set == false and reset == true) then Q = false elseif (set == true and reset == false) then Q = true elseif (set == true and reset == true) then Q = false notQ: Always equals Q inverted
	Settings	No settings available

9.3.1.3 Application Blocks/Conversion

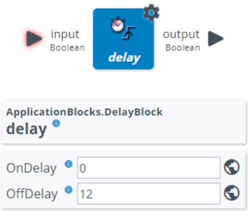
Visualization	Description	Conversion of an integer analog input to four digital Boolean outputs
 out1 Boolean out2 Boolean out4 Boolean out8 Boolean analogValue Integer toBits ApplicationBlocks.Conversion.AnalogToBit toBits Selection Device default Device default First half byte Second half byte Third half byte Fourth half byte	Number of inputs	1
	Input data type	Integer,
	Input description	analogValue: Analog input value
	Number of outputs	4
	Output data type	Output 1 ... 64: 1-bit
	Output description	out1: first digital output out2: second digital output out4: third digital output out8: fourth digital output
	Settings	To select which half byte of the integer bit representation should be connected to the output (Default First half byte) Integer bit representation: : xxxx|xxxx| xxxx|xxxx| xxxx|xxxx|xxxx|xxxx xxxx|xxxx| xxxx|xxxx xxxx|xxxx|xxxx|xxxx (64 bits) Bits are marked with “x” If “First half byte” is selected send lowest 4 bits ...|----|----|----|xxxx If “Second half byte” is selected send bits marked with x ...|----|----|xxxx|---- If “Third half byte” is selected send bits marked with x ...|----|xxxx|----|---- If “Fourth half byte” is selected send bits marked with x ...|xxxx|----|----|---- It follows the same way for other available selection settings.

Visualization	Description	Conversion of an integer to a float value
	Number of inputs	1
	Input data type	Integer,
	Input description	analogValue: Analog value to be converted
	Number of outputs	1
	Output data type	Float
	Output description	Converted float value
	Settings	No settings available

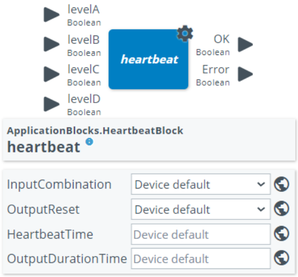
Visualization	Description	Conversion of four digital inputs to an analog half byte value.
 in1 Boolean in2 Boolean in4 Boolean in8 Boolean toNibble analogValue Integer ApplicationBlocks.Conversion.BitToAnalog toNibble Selection Device default Device default First half byte Second half byte Third half byte Fourth half byte	Number of inputs	4
	Input data type	Input 1 ... 64: 1-bit
	Input description	in1: first digital input in2: second digital input in4: third digital input in8: fourth digital input Unconnected input bits are assumed to be 0.
	Number of outputs	1
	Output data type	Output 1: Integer or UInteger, 8 or 16 bits
	Output description	analogValue: analog half byte output value
	Settings	To select which half byte of the integer bishould be connected to the output (Default First half byte). Integer bit representation: xxxx|xxxx|xxxx|xxxx| xxxx|xxxx|xxxx|xxxx xxxx|xxxx|xxxx|xxxx xxxx|xxxx|xxxx|xxxx (64 bits) Bits are marked with “x” If “First half byte” is selected send lowest 4 bits ...|----|----|----|xxxx If “Second half byte” is selected send bits marked with x ...|----|----|xxxx|---- If “Third half byte” is selected send bits marked with x ...|----|xxxx|----|---- If “Fourth half byte” is selected send bits marked with x ...|xxxx|----|----|---- It follows at the same way for other available selection settings.

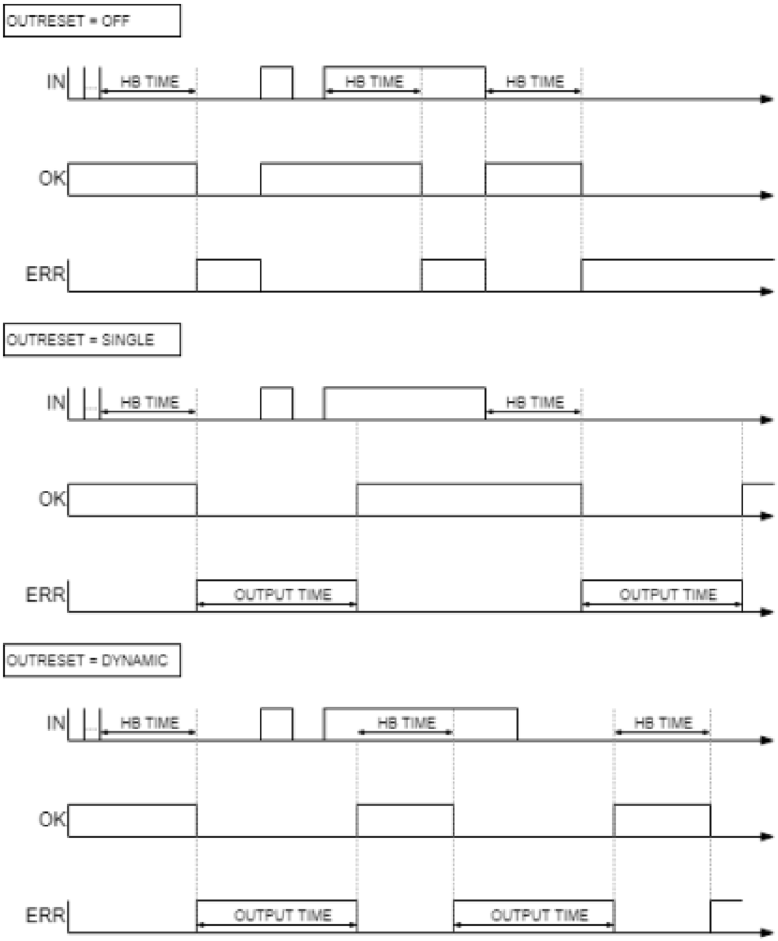
Visualization	Description	Conversion of a float input to an analog output
 in1 Float toAnalog analogValue Integer overflow Boolean ApplicationBlocks.Conversion.FloatToAnalog toAnalog RoundMode Device default Device default Round to one Round to zero	Number of inputs	1
	Input data type	Float
	Input description	in1: Float value to be converted
	Number of outputs	2
	Output data type	analogValue: Integer overflow: 1-bit
	Output description	analogValue: Converted integer value overflow: This output is set in case the floating input value exceeds the limitation of integer.
	Settings	RoundModes: To select if a number should be rounded to zero or to one.

9.3.1.4 Application Blocks/Delay blocks

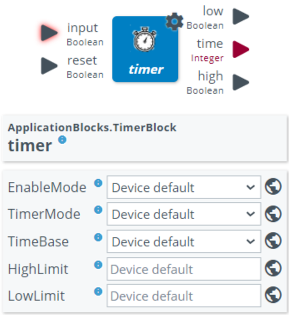
Visualization	Description	The input signal is delayed by the configured time
 ApplicationBlocks.DelayBlock delay OnDelay 0 OffDelay 12	Number of inputs	1
	Input data type	1-bit
	Input description	input: input value
	Number of outputs	1
	Output data type	1-bit
	Output description	Delayed output signal
	Settings	OnDelay: Time in ms to delay 0 -> 1 transition OffDelay: Time in ms to delay 1 -> 0 transition

9.3.1.5 Application Blocks/Heartbeat block

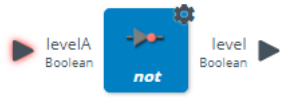
Visualization	Description	Monitors the state of the inputs and detects if they are not changing as expected within the heartbeat time.
 levelA Boolean levelB Boolean levelC Boolean levelD Boolean heartbeat OK Boolean Error Boolean ApplicationBlocks.HeartbeatBlock heartbeat InputCombination: Device default OutputReset: Device default HeartbeatTime: Device default OutputDurationTime: Device default	Number of inputs	4 2
	Input data type	Input 1 ... 16: 1-bit
	Input description	levelA: first input to be monitored levelB: second input to be monitored levelC: third input to be monitored levelD: fourth input to be monitored
	Number of outputs	2
	Output data type	Output 1: 1 bit Output 2: 1 bit
	Output description	ok: As long as the input signals are changing, this output wil be high. error: This output will be high in case the input signals are not changing within the defined heartbeat time.
	Settings	**InputCombination**: (Any / All) If “Any” is selected, the “ok” output will stay high as long as at least one input signal switches in the “Heartbeat time”. If “All” is selected, the “ok” output will only stay high as long as all input signals switch within the “Heartbeat time”. **OutputReset**: (Off / Single / Dynamic) If "Off” is selected, “Err” = high (and “OK” = low) output will stay this way until one of the inputs switches again. If "Single” is selected, Err = high (and OK = low) will revert automatically after the "Output duration Time" has elapsed and keep this state until a change in the inputs retrigger the heartbeat timer. If "Dynamic” is selected, “Err” = high (and “OK” = low) will revert automatically after the "Output duration time" has elapsed. In this case “Err” and “OK” will not revert due to any input switching. However, any input switching during this period will retrigger the heartbeat time. **HeartbeatTime**: 0...65535 ms Setting of the heartbeat time within the input(s) must change. **OutputDurationTime**: 0...65535 ms Setting of the time the output signal stays high after a "no input change" condition has been detected.

Visualization	Description	Monitors the state of the inputs and detects if they are not changing as expected within the heartbeat time.
		

9.3.1.6 Application Blocks/Timer block

Visualization	**Description**	**Timer: measures the pulse time of the digital input signal triggered by the rising or falling edge depending on the configuration. The time output will continually update as long as the input is active until it reaches the High Limit. The time output will stay at the High Limit until a reset**
 input Boolean reset Boolean timer low Boolean time Integer high Boolean ApplicationBlocks.TimerBlock timer EnableMode: Device default TimerMode: Device default TimeBase: Device default HighLimit: Device default LowLimit: Device default	Number of inputs	2
	Input data type	Input 1 ("Activate"): 1-bit Input 2 ("Reset"): 1-bit
	Input description	input: input signal Reset: Sets the timer to 0 at rising edge
	Number of outputs	3
	Output data type	Output 1 ("High"): 1-bit Output 2 ("Time"): UInteger 32 Output 3 ("Low"): 1-bit
	Output description	**low**: This output is active when the “time” output is lower than “LowLimit” (Information: The 1 ms option is not available). **time**: This value increments once per “TimeBase” whenever input is active. **high**: This output is active when the “time” output is higher than the “HighLimit”.
	Settings	**EnableMode**: To activate the mode to specify which time is to be measured. Selection between rising and falling edge of the input signal or between falling and rising edge (default: rising edge). **TimeBase**: To select the time base for the time measurement (default: 100 ms) **TimerMode** : If “StopWatch” is selected, the input is deactivated, the timer pauses at the current value. The timer can be restarted by activating the input. If “timer” is selected, the timer value is reset to 0 when the input signal becomes active. **HighLimit**: Defines an upper value for the Boolean “high” output signal that is set when the “time” value exceeds the defined upper limit (default: 0). **LowLimit**: Defines a lower value for the Boolean “low” output signal that is set when the “time” value falls below the defined lower limit (default: 0).

9.3.1.7 Digital logic/Gate

Visualization	Description	Invert the input signal with a logical NOT.
NOT  	Number of inputs	1
	Input data type	1-bit
	Input description	levelA: first input value
	Number of outputs	1
	Output data type	1 bit
	Output description	**level**: the input signal will be inverted with a logical not. Example: a high signal gets converted into a low signal.
	Settings	No settings available

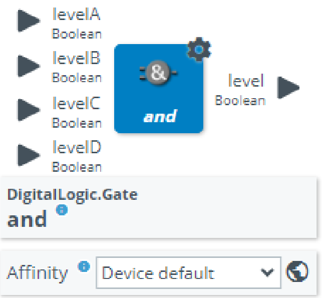
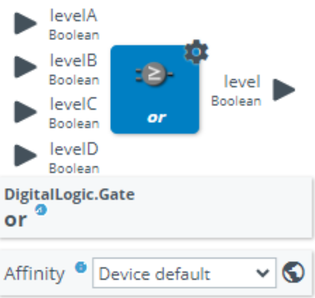
Visualization	Description	Combine the input signals with a logical AND.
AND  levelA Boolean levelB Boolean levelC Boolean levelD Boolean and level Boolean DigitalLogic.Gate and Affinity Device default	Number of inputs	4
	Input data type	1-bit
	Input description	**levelA**: first input **levelB**: second input **levelC**: third input **levelD**: fourth input Maximum 4 inputs can be linked together. If you want to link more signals, you can work with several “AND” blocks
	Number of outputs	1
	Output data type	1-bit
	Output description	**level**: the output depends on the various inputs. For more information see “Truth” table
	Settings	No settings available

Table 36: “AND” Truth table

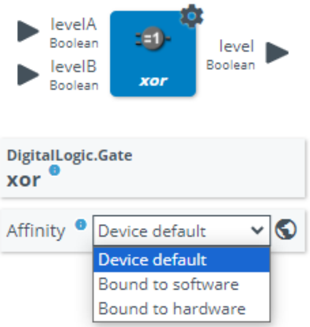
Input A	Input B	Output
1	1	1
1	0	0
0	1	0
0	0	0

Visualization	Description	Combine the input signals with a logical OR.
OR 	Number of inputs	4
	Input data type	1-bit
	Input description	**levelA**: first input **levelB**: second input **levelC**: third input **levelD**: fourth input Maximum 4 inputs can be linked together. If you want to link more signals, you can work with several “OR” blocks.
	Number of outputs	1
	Output data type	1-bit
	Output description	**level**: the output depends on the various inputs. For more information see truth table
	Settings	No settings available

Affinity Device default

Table 37: "OR” Truth table

Input A	Input B	Output
1	1	1
1	0	1
0	1	1
0	0	0

Visualization	Description	Combine the input signals with a logical XOR.
XOR 	Number of inputs	2
	Input data type	1-bit
	Input description	**levelA**: first input **levelB**: second input Maximum 2 inputs can be linked together. If you want more signals, you can work with several XOR blocks.
	Number of outputs	1
	Output data type	1-bit
	Output description	level: the output depends on the various inputs. For more information see truth table
	Settings	No settings available

Affinity Device default

Table 38: "XOR” Truth table

Input A	Input B	Output
1	1	0
1	0	1
0	1	1
0	0	0

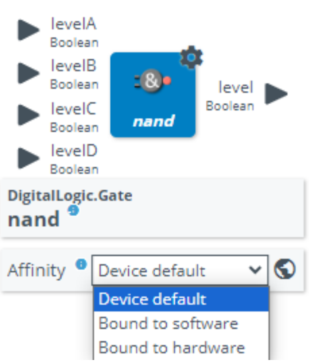
Visualization	Description	Combine the input signals with a logical NAND.
NAND 	Number of inputs	4
	Input data type	1-bit
	Input description	**levelA**: first input **levelB**: second input **levelC**: third input **levelD**: fourth input Maximum 4 inputs can be linked together. If you want to link more signals, you can work with several “NAND” blocks
	Number of outputs	1
	Output data type	1-bit
	Output description	**level**: the output depends on the various inputs. For more information see “NAND” truth table
	Settings	No settings available

Table 39: XOR” Truth table

Input A	Input B	Output
1	1	0
1	0	1
0	1	1
0	0	1

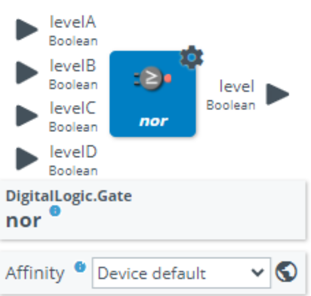
Visualization	Description	Combine the input signals with a logical NOR
NOR 	Number of inputs	4
	Input data type	1-bit
	Input description	**levelA**: first input **levelB**: second input **levelC**: third input **levelD**: fourth input Maximum 4 inputs can be linked together. If you want to link more signals, you can work with several “NOR” blocks
	Number of outputs	1
	Output data type	1-bit
	Output description	**level**: the output depends on the various inputs. For more information see “NOR” truth table
	Settings	No settings available

Table 40: "NOR” Truth table

Input A	Input B	Output
1	1	0
1	0	0
0	1	0
0	0	1

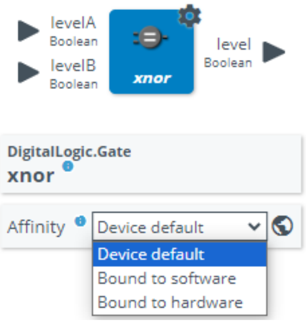
Visualization	Description	Combine the input signals with a logical XNOR
XNOR 	Number of inputs	2
	Input data type	1-bit
	Input description	**levelA**: first input **levelB**: second input **levelC**: third input **levelD**: fourth input Maximum 4 inputs can be linked together. If you want to link more signals, you can work with several “NOR” blocks
	Number of outputs	1
	Output data type	1-bit
	Output description	**level**: the output depends on the various inputs. For more information see “NOR” truth table
	Settings	No settings available

Table 41: “XNOR” Truth table

Input A	Input B	Output
1	1	1
1	0	0
0	1	0
0	0	1

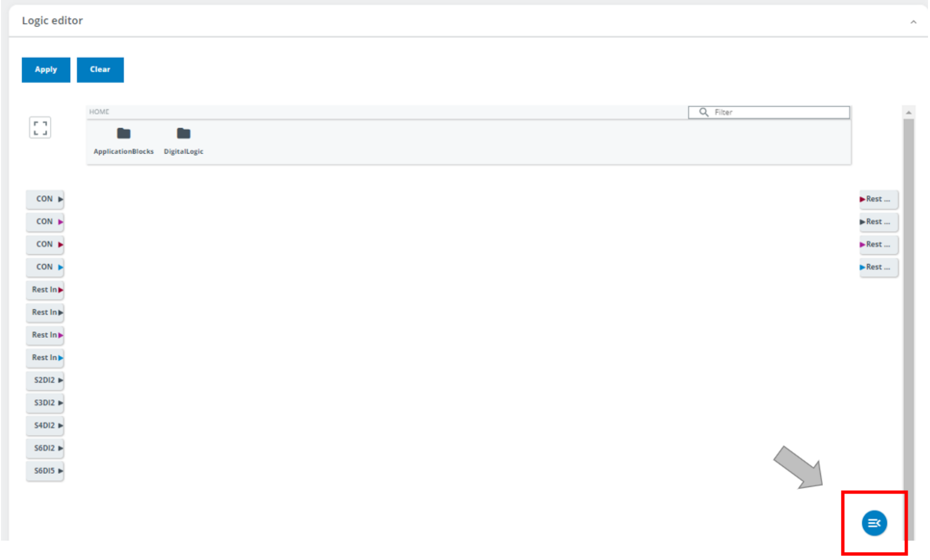
NOTE

Please be aware that the Integer values have a value range from 0 ... 65.535. There is no overflow or underflow indication.

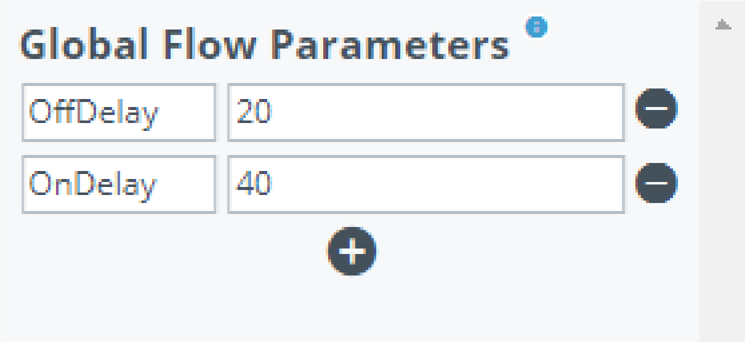
9.3.2 Global Flow Parameters

It is possible to define “global flow parameters” names and values and then apply them in the parameter editor of logic blocks. In this way, you can adjust the parameters for all blocks using the same global parameter values.

1. To open the global flow parameter menu, press the open menu button in the logic editor.



2. Press the “plus” button to add new flow parameter name and value and the “minus” button to remove an existing one.



Global Flow Parameters

OffDelay	20
OnDelay	40

✓ In the parameter settings menu for the logic blocks, a globe icon shows the possible global parameters that can be set.

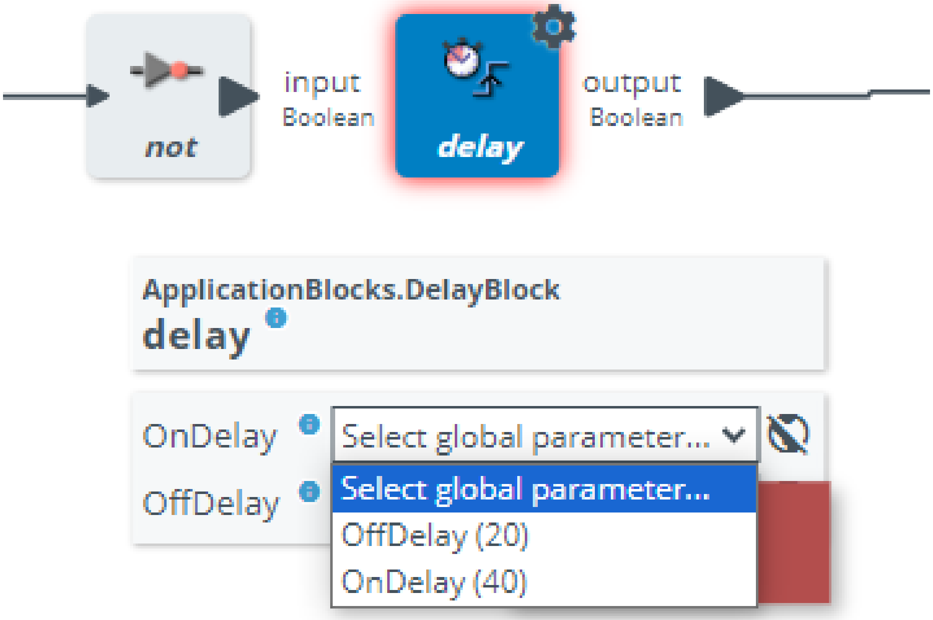
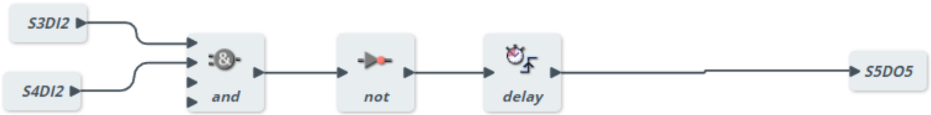
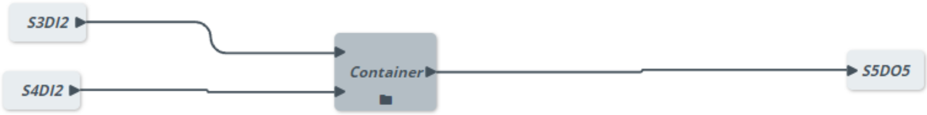

not
input
Boolean
delay
output
Boolean
ApplicationBlocks.DelayBlock
delay
OnDelay
OffDelay
Select global parameter...
Select global parameter...
OffDelay (20)
OnDelay (40)

Figure 30: Global Flow parameters within a function block setting

9.3.3 Logic Block Containers

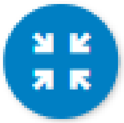
To simplify logic diagrams, logic blocks can be grouped to hide the underlying logic. For example, the first diagram in the below picture can be grouped in a single “Container” block.





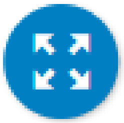
Create a container

1. “CTL-click” on each logic block you want grouped together.
2. Select the “group into container” button at the bottom of the working area.



Ungroup a set of logic blocks

1. Highlight the container by clicking on it.
2. Select the “Ungroup container” button.



10 Troubleshooting

If you have problems, SICK Technical Support is available to help. Contact your SICK Service partner in this case.

However, a large number of problems can be identified and fixed independently with the help of the following tips:

- ▸ Make sure the LEDs are not reporting problems.
- ▸ Check that the network IP address, subnet mask and gateway are configured correctly.
- ▸ Make sure that the IP address programmed in the IO-Link Master matches the unique, reserved, configured IP address assigned by the system administrator.
- ▸ Make sure that you use the correct cable types on the correct connections and that all cables are firmly connected.
- ▸ Disconnect the IO-Link device from the master and reconnect it. Possibly the master had not correctly identified the connected device.
- ▸ Restart the IO-Link Master.
- ▸ Check whether the Data Storage function has been activated correctly.
- ▸ Reset the module to factory settings (see "Device restart", page 101).
- ▸ Update the firmware of the module (see "Firmware update", page 101).

10.1 Factory reset

In some cases, it is helpful to reset the module to the factory settings.

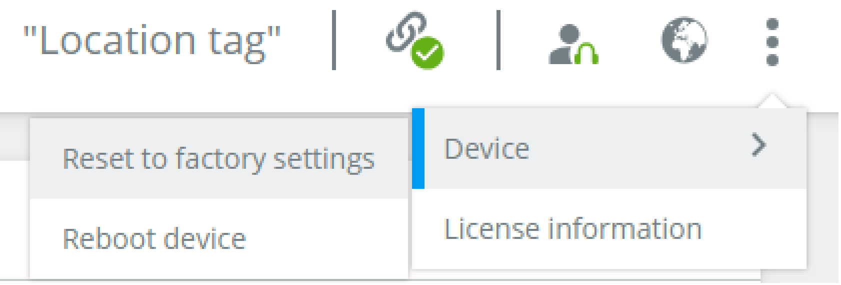
Changes to settings in the webserver/Sopas interface require “service” rights. If these are not available, the Settings tab is grayed out and cannot be edited.

NOTE
“Reset to factory settings” do not reset Maintenance and Service password.

NOTE
“GatewayAccessRights” need to be “SuperUser” (default delivery status) to perform FactoryReset

NOTICE
By resetting to the factory settings, all parameterizations made are lost and must be carried out again.

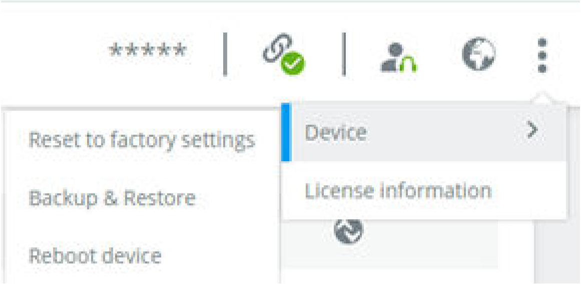
We recommend creating a backup before resetting to the factory settings. As soon as the module has been reset to factory settings, all parameterizations made are deleted and cannot be restored. A backup can save considerable effort in the event of an error.



10.2 Device restart

In some cases, it is necessary to restart the module. To do this, either use the web interface or SOPAS and logged as “maintenance”.

Switching the voltage supply off and on also results in a device restart.



10.3 Firmware update

Device`s firmware can be updated using "Sentio Creator" software using "Deployment Manager" extension.

Sentio Creator – Download

For detailed information please refer to the Operation Instruction of the Sentio Creator.

11 Disassembly and disposal

The SIG300 must be disposed of in line with applicable country-specific regulations. When disposing of them, you should try to recycle them (especially the precious metals).

NOTE
Disposal of batteries, electrical and electronic devices

- In accordance with international regulations, batteries, rechargeable batteries and electrical and electronic devices must not be disposed of with household waste.
- In accordance with international regulations, batteries, rechargeable batteries and electrical and electronic devices must not be disposed of with household waste.
- This symbol on the product, its packaging or in the document indicates that a product is subject to the specified regulations.

12 Maintenance

Sensor Integration Gateways from SICK are maintenance-free.

We recommend performing the following on a regular basis:

- Clean device.
- Check screw connections and plug connections.

No modifications may be made to devices.

13 Technical data

13.1 General technical data

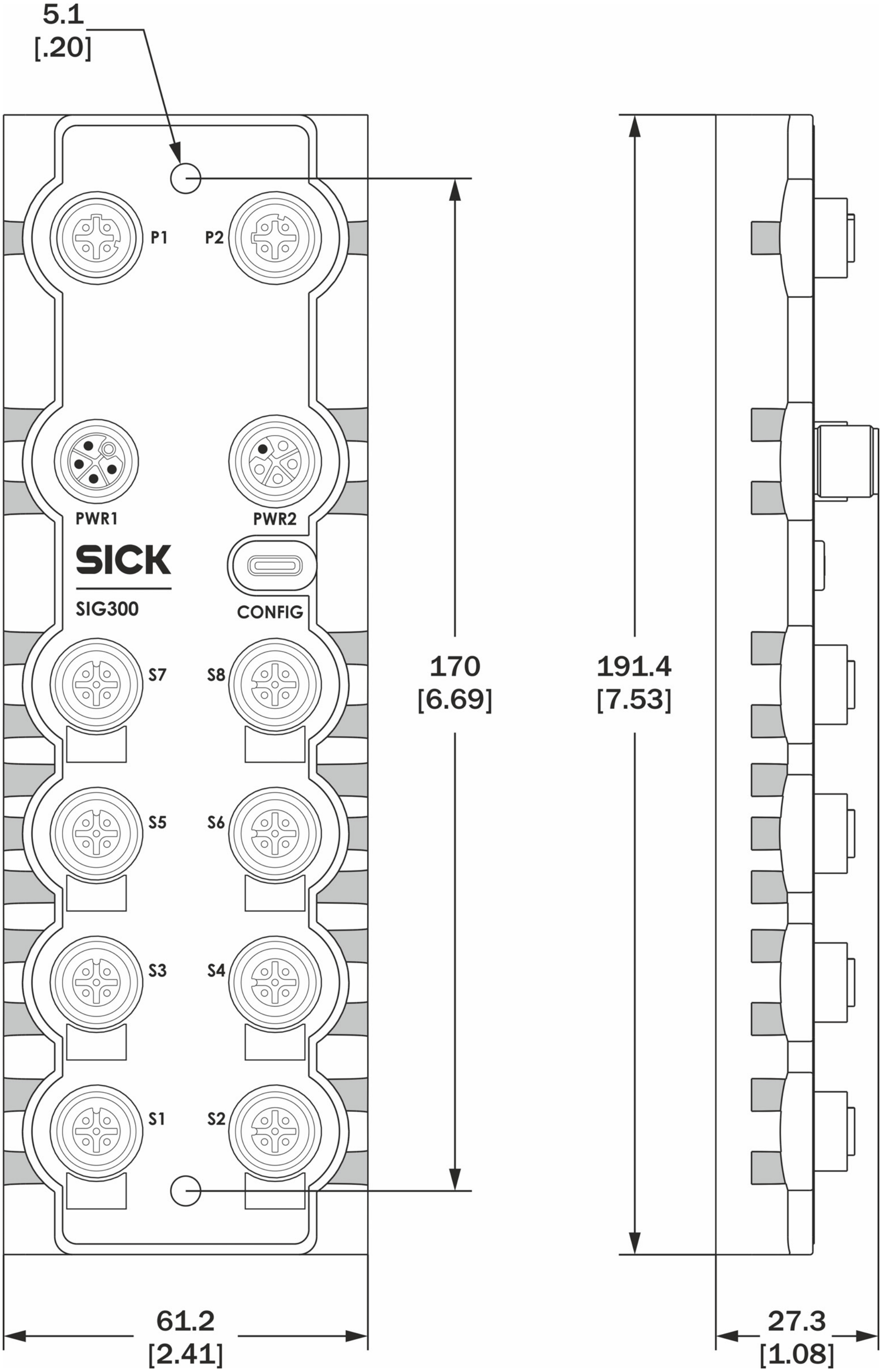


Figure 31: Dimensional drawing

Table 42: Mechanical data

Housing material	Plastic (ELEMELT PA514)

Enclosure rating as per IEC 60529	IP67 (only with connected cables)[1]
Dimensions (L x W x H)	191.4 mm x 61.2 mm x 27.3 mm
Mounting type	2-hole screw mounting
Weight	460 g

[1] If the appropriate cables are not connected to all ports, the free ports must be screwed tight with sealing plugs (part number 2136242) to ensure IP protection.

Table 43: Operating conditions

Operating temperature	-40 °C ... +55 °C[1]
Storage temperature	-40 °C ... +75 °C[1]
EMC	EN 61000-6-2 EN 61000-6-4
Impact load	EN 60068-2-27

[1] Permissible relative humidity 0% ... 95% (non-condensing)

NOTE

This equipment is not intended for use in residential areas and may not provide adequate protection against radio reception in these environments.

Table 44: Electrical data

Voltage supply U_A	20 ... 30 V DC	
Voltage supply U_S	20 ... 30 V DC	With IO-Link devices
	11.5 ... 30 V DC	without IO-Link-devices.
“Voltage supply in” (PWR1)	U_S Current consumption	≤ 180 mA @ 24 V[1]
	U_A Current consumption	≤ 20 mA @ 24 V[1]
	Max. current carrying capacity	≤ 16 A, U_S[2] ≤ 16 A, U_A[2]
“Voltage supply out” (PWR2)	Max. current carrying capacity	≤ 16 A, U_S[2] ≤ 16 A, U_A[2]
Port (S1-S8)	Pin 1 max. supply current (Class A)	2 A
	Pin 1 max. supply current (Class B)	2 A
	Pin 2 max. supply current (DO) (Class A)	0.2 A
	Pin 2 max. supply current (Class B)	2 A
	Pin 4 max. supply current (DO) (Class A)	0.2 A
	Pin 4 max. supply current (DO) (Class B)	0.2 A
	Pin 5 max. supply current (DO) (Class A)	0.2 A
	Max. current carrying capacity per port	4.2 A
	Max. current carrying capacity of all IO-Link ports (S1-S8)[3]	12 A
	Input characteristics	EN 61131-2 type 1

Protection class	III[4]

[1] Without load, sensors and outputs switched off

[2] U_S + U_A total <= 25 A.

[3] Max. current per port includes both the output current (pin 4, pin 2 and pin 5, if applicable) and the current consumption of the connected device (pin 1).

[4] When using a SELV or PELV power supply unit

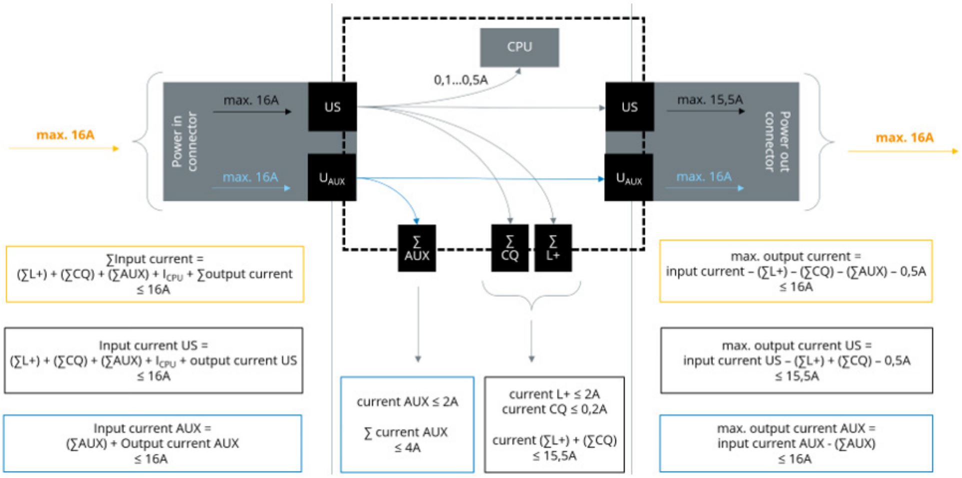

CPU
0,1...0,5A
max. 16A
Power in connector
max. 16A
US
max. 16A
U_{AUX}
US
max. 15,5A
U_{AUX}
max. 16A
Power out connector
max. 16A
∑ AUX
∑ CQ
∑ L+
∑Input current = (∑L+) + (∑CQ) + (∑AUX) + I_{CPU} + ∑output current ≤ 16A
Input current US = (∑L+) + (∑CQ) + (∑AUX) + I_{CPU} + output current US ≤ 16A
Input current AUX = (∑AUX) + Output current AUX ≤ 16A
current AUX ≤ 2A
∑ current AUX ≤ 4A
current L+ ≤ 2A
current CQ ≤ 0,2A
current (∑L+) + (∑CQ) ≤ 15,5A
max. output current = input current – (∑L+) – (∑CQ) – (∑AUX) – 0,5A ≤ 16A
max. output current US = input current US – (∑L+) + (∑CQ) – 0,5A ≤ 15,5A
max. output current AUX = input current AUX - (∑AUX) ≤ 16A

Figure 32: Absolute current limitations

Table 45: Additional information

Max. number of I/O s that can be connected directly to SIG300	Max 16: Port 1...4: 2 I/O s Port 5...6: 3 I/O s Port 7...8: 1 I/O
Max. number of I/O-Link signals that can be connected:	8
Ethernet ports:	2
Max. switching frequency Digital input to digital Output connected through logic editor connection	50 Hz

Table 46: IO-Link

Specification	V1.1.3
Connection class	Class A Class B
Transmission rate	COM1 / COM2 / COM3
Min. IO-Link cycle time	0.4 ms
Detection of transmission rate	Automatic

8029286/1SL0/2025-06-30/en

Australia
Phone +61 (3) 9457 0600
1800 33 48 02 – tollfree
E-Mail sales@sick.com.au

Austria
Phone +43 (0) 2236 62288-0
E-Mail office@sick.at

Belgium/Luxembourg
Phone +32 (0) 2 466 55 66
E-Mail info@sick.be

Brazil
Phone +55 11 3215-4900
E-Mail comercial@sick.com.br

Canada
Phone +1 905.771.1444
E-Mail cs.canada@sick.com

Czech Republic
Phone +420 234 719 500
E-Mail sick@sick.cz

Chile
Phone +56 (2) 2274 7430
E-Mail chile@sick.com

China
Phone +86 20 2882 3600
E-Mail info.china@sick.net.cn

Denmark
Phone +45 45 82 64 00
E-Mail sick@sick.dk

Finland
Phone +358-9-25 15 800
E-Mail sick@sick.fi

France
Phone +33 1 64 62 35 00
E-Mail info@sick.fr

Germany
Phone +49 (0) 2 11 53 010
E-Mail info@sick.de

Greece
Phone +30 210 6825100
E-Mail office@sick.com.gr

Hong Kong
Phone +852 2153 6300
E-Mail ghk@sick.com.hk

Hungary
Phone +36 1 371 2680
E-Mail ertekesites@sick.hu

India
Phone +91-22-6119 8900
E-Mail info@sick-india.com

Israel
Phone +972 97110 11
E-Mail info@sick-sensors.com

Italy
Phone +39 02 27 43 41
E-Mail info@sick.it

Japan
Phone +81 3 5309 2112
E-Mail support@sick.jp

Malaysia
Phone +603-8080 7425
E-Mail enquiry.my@sick.com

Mexico
Phone +52 (472) 748 9451
E-Mail mexico@sick.com

Netherlands
Phone +31 (0) 30 204 40 00
E-Mail info@sick.nl

New Zealand
Phone +64 9 415 0459
0800 222 278 – tollfree
E-Mail sales@sick.co.nz

Norway
Phone +47 67 81 50 00
E-Mail sick@sick.no

Poland
Phone +48 22 539 41 00
E-Mail info@sick.pl

Romania
Phone +40 356-17 11 20
E-Mail office@sick.ro

Singapore
Phone +65 6744 3732
E-Mail sales.gsg@sick.com

Slovakia
Phone +421 482 901 201
E-Mail mail@sick-sk.sk

Slovenia
Phone +386 591 78849
E-Mail office@sick.si

South Africa
Phone +27 10 060 0550
E-Mail info@sickautomation.co.za

South Korea
Phone +82 2 786 6321/4
E-Mail infokorea@sick.com

Spain
Phone +34 93 480 31 00
E-Mail info@sick.es

Sweden
Phone +46 10 110 10 00
E-Mail info@sick.se

Switzerland
Phone +41 41 619 29 39
E-Mail contact@sick.ch

Taiwan
Phone +886-2-2375-6288
E-Mail sales@sick.com.tw

Thailand
Phone +66 2 645 0009
E-Mail marcom.th@sick.com

Turkey
Phone +90 (216) 528 50 00
E-Mail info@sick.com.tr

United Arab Emirates
Phone +971 (0) 4 88 65 878
E-Mail contact@sick.ae

United Kingdom
Phone +44 (0)17278 31121
E-Mail info@sick.co.uk

USA
Phone +1 800.325.7425
E-Mail info@sick.com

Vietnam
Phone +65 6744 3732
E-Mail sales.gsg@sick.com

Detailed addresses and further locations at **www.sick.com**

SICK AG | Waldkirch | Germany | www.sick.com

SICK
Sensor Intelligence.