

Manual

EtherCAT®

LioN-X IO-Link Master:
0980 XSL 3912-121-007D-00F
(8 x IO-Link Class A, Multiprotocol)

LioN-Xlight IO-Link Master:
0980 LSL 3211-121-0006-004
(8 x IO-Link Class A, EtherCAT®)

0980 LSL 3210-121-0006-004
(4 x IO-Link Class A + 8 x DI, EtherCAT®)



Contents

1 About this manual	8
1.1 General information	8
1.2 Explanation of symbols	9
1.2.1 Use of danger information	9
1.2.2 Use of general information	9
1.2.3 EtherCAT® trademark information	10
1.3 Version information	10
2 Safety instructions	11
2.1 Intended use	11
2.2 Qualified personnel	12
3 Designations and synonyms	13
4 System description	16
4.1 About LioN-X and LioN-Xlight	16
4.2 Device variants	17
4.3 I/O port overview	19
5 Overview of product features	21
5.1 EtherCAT® product features	21
5.2 I/O port features	22
5.3 Integrated Web server	23
5.4 Security features	24

5.5 Other features	25
--------------------	----

6 Assembly and wiring **26**

6.1 General information	26
6.2 Outer dimensions	27
6.2.1 LioN-X multiprotocol variants	27
6.2.2 LioN-Xlight variants with EtherCAT®	28
6.2.3 Notifications	30
6.3 Port assignments	31
6.3.1 Ethernet ports, M12 socket, 4-pin, D-coded	31
6.3.2 Power supply with M12 power L-coded	32
6.3.3 I/O ports as M12 sockets	33
6.3.3.1 IO-Link Class A	33

7 Starting operation **35**

7.1 ESI file	35
7.2 MAC addresses	36
7.3 Setting the rotary encoding switches	37
7.3.1 Factory reset	39

8 Configuration and operation in TwinCAT® 3 **40**

8.1 PDO assignments	40
8.1.1 Input data	40
8.1.2 Output data	46
8.1.3 Modular slots	50
8.2 Device parameters	53
8.2.1 Extended parameters	53
8.2.2 Failsafe replacement values	54
8.2.3 Failsafe mode for the digital output	55
8.2.4 General device settings	58
8.2.5 Surveillance timeout	59

8.2.6 Digital I/O mode, Channel B	61
8.2.7 Digital Input logic	63
8.2.8 Digital Input filter	65
8.2.9 Digital Output restart	66
8.2.10 Additional IO-Link port settings	68
8.2.11 IO-Link Parameterization	70
8.2.12 IO-Link configuration data	71
8.2.13 IO-Link serial number	72
8.2.14 IO-Link information data	73
8.2.15 IO-Link serial number of connected devices	74
8.3 Configuration example with TwinCAT® 3	75
8.3.1 Configuration of 0980 XSL 391x-1x1 devices	77
8.3.2 EoE IP address	79
8.3.3 Activate configuration	81

9 Diagnostics processing 82

9.1 Error of the system/sensor power supply	82
9.2 Error of the auxiliary/actuator power supply	82
9.3 Overload/short-circuit of the I/O port sensor supply outputs	82
9.4 Overload/short circuit of the digital outputs	83
9.5 IO-Link C/Q error	83
9.6 Diagnostic history object (0x10F3)	84
9.6.1 Maximum Messages	84
9.6.2 Newest Message	84
9.6.3 Newest Acknowledge Message	84
9.6.4 New Messages Available	86
9.6.5 Flags	86
9.6.6 Diagnosis Message Buffer	87
9.7 Diagnostic data object (0xA0n0)	88
9.8 Device status object (0xF100)	89
9.9 Emergency messages	90

10 IIoT functionality	91
10.1 MQTT	92
10.1.1 MQTT configuration	92
10.1.2 MQTT topics	95
10.1.2.1 Base topic	95
10.1.2.2 Publish topic	98
10.1.3 MQTT configuration - Quick start guide	103
10.1.3.1 MQTT configuration via JSON	103
10.2 OPC UA	105
10.2.1 OPC UA configuration	106
10.2.2 OPC UA address space	108
10.2.3 OPC UA configuration - Quick start guide	109
10.2.3.1 OPC UA configuration via JSON	109
10.3 REST API	111
10.3.1 Standard device information	113
10.3.2 Structure	114
10.3.3 Configuration and forcing	120
10.3.4 Reading and writing ISDU parameters	122
10.3.4.1 Reading ISDU	122
10.3.4.2 Writing ISDU	124
10.3.5 Example: Reading ISDU	126
10.3.6 Example: Writing ISDU	126
10.4 CoAP server	127
10.4.1 CoAP configuration	127
10.4.2 REST API access via CoAP	128
10.4.3 CoAP configuration - Quick start guide	131
10.4.3.1 CoAP configuration via JSON	131
10.5 Syslog	133
10.5.1 Syslog configuration	133
10.5.2 Syslog configuration - Quick start guide	136
10.5.2.1 Syslog configuration via JSON	136

11 The integrated Web server	138
11.1 LioN-X 0980 XSL... variants	139
11.1.1 The Status page	139
11.1.2 The Ports page	140
11.1.3 The System page	141
11.1.4 The User page	143
11.2 LioN-Xlight 0980 LSL... variants	144
11.2.1 The System page	144
12 Firmware update	146
12.1 Firmware update via FoE	146
13 Technical data	149
13.1 General	150
13.2 EtherCAT® protocol	151
13.3 Power supply of the module electronics/sensors	152
13.4 Power supply of the actuators	153
13.5 IO-Link Master ports Class A, Pin 4	154
13.5.1 Configured as a digital input	154
13.5.2 Configured as a digital output	155
13.5.3 Configured as an IO-Link port in COM mode	156
13.6 IO-Link Master ports Class A, Pin 2	157
13.6.1 Configured as a digital input	157
13.6.2 Configured as a digital output	158
13.7 LEDs	159
14 Accessories	161

15 References

162

1 About this manual

1.1 General information

Please read the assembly and operating instructions in this manual carefully before starting up the modules. Keep the manual where it is accessible to all users.

The texts, figures, diagrams, and examples used in this manual are used exclusively to explain how to operate and apply the modules.

Please contact us if you have any detailed questions on installing and starting up the devices.

Belden Deutschland GmbH
– Lumberg Automation™ –
Im Gewerbepark 2
D-58579 Schalksmühle
Germany
lumberg-automation-support.belden.com
www.lumberg-automation.com
catalog.belden.com

Belden Deutschland GmbH – Lumberg Automation™ – reserves the right to make technical changes or changes to this manual at any time without notice.

1.2 Explanation of symbols

1.2.1 Use of danger information

Danger information is denoted as follows:



Danger: Means that death, serious physical injury or substantial damage to property will occur if the required safety measures are not taken.



Warning: Means that death, serious physical injury or substantial damage to property can occur if the required safety measures are not taken.



Caution: Means that minor physical injury or damage to property can occur if the required safety measures are not taken.

1.2.2 Use of general information

General information is denoted as follows:



Attention: Contains important information on the product, on how to manage the product, or on the respective section of the documentation to which your special attention is being drawn.

1.2.3 EtherCAT® trademark information

EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

1.3 Version information

Index	Created	Changed
Version number	Version 1.0	Version 1.1
Date	03/2021	04/2021

Index	Changed	Changed
Version number	Version 1.2	
Date	05/2021	

Table 1: Overview of manual revisions

2 Safety instructions

2.1 Intended use

The products described in this manual are decentralized IO-Link Masters on an Industrial Ethernet Network.

We adhere to all safety standards when developing, producing, testing, and documenting our products. When you adhere to the handling specifications and safety instructions described for the configuration, assembly, and correct operation, there should not normally be any risks for people or equipment.

The modules fulfill the requirements of the EMC guidelines (89/336/EEC, 93/68/EEC and 93/44/EEC) and the low voltage guideline (73/23/EEC).

The IO-Link Masters are designed to be used in the industrial sector. The industrial environment is distinguished by the fact that the consumer is not connected directly to the public low voltage network. Additional measures are required for use in residential areas or in business and commercial sectors.



Attention: This equipment may cause radio interference in residential areas. In this case the operator may be requested to carry out appropriate measures.

The proper and safe operation of this product depends on proper transportation, storage, assembly, and installation, and careful operation.

A completely assembled device housing is required for the proper operation of the IO-Link Masters. Only connect devices that fulfill the requirements of EN 61558-2-4 and EN 61558-2-6 to the IO-Link Masters.

During the configuration, installation, start-up, maintenance, and testing of the devices, adhere to the safety and accident-prevention guidelines for the specific application.

Only install cables and accessories that fulfill the requirements and regulations for safety, electromagnetic compatibility, and, where applicable, telecommunication end devices, as well as the specification information.

Information on which cables and accessories are permitted for the installation can be obtained from Lumberg Automation™ or is contained in this manual.

2.2 Qualified personnel

The configuration, installation, start-up, maintenance, and testing of the devices may only be performed by a qualified electrician who is familiar with the safety standards of the automation technology.

The personnel requirements are based on the requirement profiles described by ZVEI, VDMA, or equivalent organizations.

Only electricians who are familiar with the content of this manual are authorized to install and maintain the devices described. These are persons who

- ▶ based on their technical training, knowledge, and experience, and their knowledge of the pertinent standards, can evaluate the work to be carried out and identify any potential risks or
- ▶ based on working for several years in a related sector, have the same level of knowledge as they would have from the relevant technical training.

Only Belden Deutschland GmbH – Lumberg Automation™ – is permitted to make changes to the hardware or software of the products that go beyond the scope of this manual.



Warning: Making unqualified changes to the hardware or software, or non-adherence to the warning information contained in this manual, can result in serious personal injury or damage to equipment.



Attention: Belden accepts no liability for any damage caused by unqualified personnel or improper use. This automatically voids the warranty.

3 Designations and synonyms

API	Application Programming Interface
BF	Bus Fault LED
Big Endian	Data format with High-B on first place (PROFINET and IO-Link)
BUI	Back-Up Inconsistency (EIP diagnostics)
C/Q	I/O port pin 4 mode, IO-Link communication/switching signal
Ch. A	Channel A (Pin 4) of I/O port
Ch. B	Channel B (Pin 2) of I/O port
CIP	Common Industrial Protocol (media independent protocol)
Class A	IO-Link port specification (Class A)
Class B	IO-Link port specification (Class B)
CoAP	Constrained Application Protocol
DCP	Discovery and Configuration Protocol
DevCom	Device Communicating (EIP diagnostics)
DevErr	Device Error (EIP diagnostics)
DI	Digital Input
DIA	Diagnostic LED
DO	Digital Output
DIO	Digital Input/Output
DTO	Device Temperature Overrun (EIP diagnostics)
DTU	Device Temperature Underrun (EIP diagnostics)
DUT	Device under test
EIP	Ethernet/IP
EIS	EtherNet/IP string
ERP	Enterprise Resource Planning system
ETH	ETHERNET
FE	Functional Earth
FME	Force Mode Enabled (EIP diagnostics)

FSU	Fast Start-Up
GSDML	General Station Description Markup Language
High-B	High-Byte
ICE	IO-Link port COM Error (EIP diagnostics)
ICT	Invalid Cycle Time (EIP diagnostics)
IDE	IO-Link port Device Error (EIP diagnostics)
IDN	IO-Link port Device Notification (EIP diagnostics)
IDW	IO-Link port Device Warning (EIP diagnostics)
IIoT	Industrial Internet of Things
ILE	Input process data Length Error (EIP diagnostics)
IME	Internal Module Error (EIP diagnostics)
I/Q	I/O port pin 2 mode, Digital Input/switching signal
I/O	Input / Output
I/O port	X1 - X8
I/O port pin 2	Channel B of X1 - X8
I/O port pin 4 (C/Q)	Channel A of X1 - X8
IOL or IO-L	IO-Link
ISDU	Indexed Service Data Unit
IVE	IO-Link port Validation Error (EIP diagnostics)
I&M	Identification & Maintenance
JSON	JavaScript Object Notation (platform independent data format)
L+	I/O port pin 1, sensor power supply
LioN-X 60	LioN-X variants with a width of 60mm
Little Endian	Data format with Low-B on first place (Ethernet/IP)
LLDP	Link Layer Discovery Protocol
Low-B	Low-Byte
LSB	Least Significant Bit
LVA	Low Voltage Actuator Supply (EIP diagnostics)
LVS	Low Voltage System/Sensor Supply (EIP diagnostics)
MIB	Management Information Base

3 Designations and synonyms

MP	Multi Protocol (PROFINET + EtherNet/IP + EtherCAT® + Modbus TCP)
MQTT	Message Queuing Telemetry Transport (open networking protocol)
MSB	Most Significant Bit
M12	Metric thread according to DIN 13-1 with 12 mm diameter
OLE	Output process data Length Error (EIP diagnostics)
OPC UA	Open Platform Communications Unified Architecture (platform independent, service-oriented architecture)
PLC	Programmable Logic Controller
PN	PROFINET
PNS	PROFINET string
PWR	Power
REST	REpresentational State Transfer
RFC	Request for Comments
RPI	Requested Packet Interval
SCA	Short Circuit Actuator/ U_L/U_{Aux} (EIP diagnostics)
SCS	Short Circuit Sensor (EIP diagnostics)
SNMP	Simple Network Management Protocol
SP	Single Protocol (PROFINET, EtherNet/IP, EtherCAT® or Modbus TCP)
SPE	Startup Parameterization Error (EIP diagnostics)
U_{Aux}	$U_{Auxiliary}$
UDP	User Datagram Protocol
U_L	U_{Load} , supply voltage for the load circuit (Actuator supply on Class A IO-Link Master)
UL	Underwriters Laboratories Inc. (certification company)
UINT16	Unsigned integer with 16 bits or word in PLC (IW, QW)
UINT8	Byte in PLC (IB, QB)

Table 2: Designations and synonyms

4 System description

The LioN modules (Lumberg Automation™ **I**nput/**O**utput **N**etwork) function as the interface in an industrial Ethernet system: A central controller on the management level is able to communicate with the decentralized sensors and actuators on the field level. The line or ring topologies for which LioN modules can be used ensure not only reliable data communication but also significantly reduce the number of cables required and thus also the costs for and maintenance. They additionally enable easy and quick extension.

4.1 About LioN-X and LioN-Xlight

LioN-X and the LioN-Xlight variants are IO-Link Masters which convert standard input, standard output or IO-Link signals from sensors & actuators into an industrial Ethernet protocol (PROFINET, EtherNet/IP, EtherCAT®, Modbus TCP) and/or into a cloud protocol (REST API, OPC UA, MQTT). For the first time, there is now Syslog on board. The robust 8 port housing design allows the use even in harsh environments where e.g. weld field immunity, high temperature ranges or protection class IP67 & IP69K are needed. There are also LioN-Xlight single protocol versions available with a limited feature set at a highly attractive price point.

Use all benefits of the Lumberg Automation™ product solution by additionally downloading the configuration tool *LioN-Management Suite V2.0* from www.belden.com to enable e.g. a fast and easy parameterization of the connected IO-Link devices via the embedded IODD interpreter.

4.2 Device variants

The following IO-Link Masters are available in the LioN-X and the LioN-Xlight family:

Article number	Product designation	Description	I/O port functionality
935700001	0980 XSL 3912-121-007D-00F	LioN-X M12-60 mm, IO-Link Master Multiprotocol Security	8 x IO-Link Class A
935701001	0980 LSL 3011-121-0006-001	LioN-Xlight M12-60 mm, IO-Link Master PROFINET	8 x IO-Link Class A
935702001	0980 LSL 3010-121-0006-001	LioN-Xlight M12-60 mm, IO-Link Master PROFINET	4 x IO-Link Class A + 8 x DI
935701002	0980 LSL 3111-121-0006-002	LioN-Xlight M12-60 mm, IO-Link Master EtherNet/IP	8 x IO-Link Class A
935702002	0980 LSL 3110-121-0006-002	LioN-Xlight M12-60 mm, IO-Link Master EtherNet/IP	4 x IO-Link Class A + 8 x DI
935701004	0980 LSL 3311-121-0006-008	LioN-Xlight M12-60 mm, IO-Link Master Modbus TCP	8 x IO-Link Class A
935702004	0980 LSL 3310-121-0006-008	LioN-Xlight M12-60 mm, IO-Link Master Modbus TCP	4 x IO-Link Class A + 8 x DI
935701003	0980 LSL 3211-121-0006-004	LioN-Xlight M12-60 mm, IO-Link Master EtherCAT®	8 x IO-Link Class A

Article number	Product designation	Description	I/O port functionality
935702003	0980 LSL 3210-121-0006-004	LioN-Xlight M12-60 mm, IO-Link Master EtherCAT®	4 x IO-Link Class A + 8 x DI

Table 3: Overview of LioN-X and LioN-Xlight variants

4.3 I/O port overview

The following tables show the main I/O port differences of the LioN-X IO-Link Master family. Pin 4 and Pin 2 of the I/O ports can be configured partly to IO-Link, Digital Input or Digital Output.

LioN-X

Device variant:	Port	Pin 1 U _S	Pin 4 / Ch. A (C/Q)				Pin 2 / Ch. B (I/Q)	
0980 XSL 3x12...	Info:	–	Class A	Type 1	Supply by U _S ¹⁾	Supply by U _L ²⁾	Type 1	Supply by U _L ²⁾
	X8:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
	X7:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
	X6:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
	X5:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
	X4:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
	X3:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
	X2:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)
X1:	Out (4 A)	IOL	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)	

Table 4: Port configuration of 0980 XSL 3x12... variants

¹⁾ DO switch mode configured as "Push-Pull" (description in the configuration chapters).

²⁾ DO switch mode configured as "High-Side" (description in the configuration chapters).

LioN-Xlight

Device variant:	Port	Pin 1 U _S	Pin 4 / Ch. A (C/Q)			Pin 2 / Ch. B (I/Q)
0980 LSL 3x11...	Info:	–	Class A	Type 1	Supply by U _S ¹⁾	Type 1
	X8:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X7:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X6:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X5:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X4:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X3:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X2:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
X1:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI	

Table 5: Port configuration of 0980 LSL 3x11... variants

Device variant:	Port	Pin 1 U _S	Pin 4 / Ch. A (C/Q)			Pin 2 / Ch. B (I/Q)
0980 LSL 3x10...	Info:	–	Class A	Type 1	Supply by U _S ¹⁾	Type 1
	X8:	Out (0.7 A)	–	DI	–	DI
	X7:	Out (0.7 A)	–	DI	–	DI
	X6:	Out (0.7 A)	–	DI	–	DI
	X5:	Out (0.7 A)	–	DI	–	DI
	X4:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X3:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X2:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI
	X1:	Out (2 A)	IOL	DI	DO (0.5 A*)	DI

Table 6: Port configuration of 0980 LSL 3x10... variants

¹⁾ With DO Switch Mode configured as "Push-Pull" (see description in the configuration chapters).

* For **UL applications**: Max. 0.25 A DO.

5 Overview of product features

5.1 EtherCAT® product features

Data connection

The connection option provided by LioN-X is the widely-used M12 connector with D-coding for the EtherCAT® network.

The connectors are also color-coded to prevent the ports from being mixed up.

Data transmission rates

Support of 100Mbit/s with auto crossover and auto negotiation corresponding to IEEE 802.3.

Integrated switch

The integrated Ethernet switch with Conformance Class C (CC-C) has two EtherCAT® ports and thus supports the establishment of a line or ring topology for the EtherCAT® network.

Alarm and diagnostic messages

The devices support messages and alarms via a ring buffer based *Diagnosis History Object*.

5.2 I/O port features

IO-Link specification.

LioN-X is ready for IO-Link specification v1.1.3.

8 x IO-Link Master ports

Depending on the variant, the Master device has 4 Class A or 8 Class A ports with an additional hard-wired digital input on pin 2 of the I/O port. For detailed information see chapter [I/O port overview](#) on page 19.



Warning: If devices with electric isolation and devices without electric isolation are used within the same system, the electric isolation of all connected devices is annulled.

IO-Link port connections

The IO-Link port connection option provided by LioN-X devices is the 5-pin M12 connector. Pin 5 is not assigned for IO-Link Class A ports.

Validation & Backup

The Validation & Backup function checks if the right device is connected and stores the parameters of the IO-Link Device. The function thus gives you an easy option for replacing the IO-Link Device.

This is possible as of IO-Link specification V1.1 and only if the IO-Link Device **and** the IO-Link Master support the function.

IO-Link Device parameterization

IO-Link Device parameterization in an EtherCAT® network is possible via TwinCAT®.

LED

You can see the status of a port by the color of the matching LEDs and their flash pattern. For details on the

meanings of the LED colors, please see section [LEDs](#) on page 159.

5.3 Integrated Web server

Network parameter display

Get an overview of network parameters such as the IP address, subnet mask and gateway.

Displaying diagnostics

View diagnostics via the integrated Web server.

User management

Use the integrated Web server for convenient management of all users.

IO-Link Device parameters

You can read the IO-Link Device parameters and write new parameters in single write mode (single write mode doesn't activate the automated mechanism of the "Validation and Backup" function) to the IO-Link Device.

5.4 Security features

Firmware signature

All official firmware update packages contain a signature which prevents the system against manipulated firmware updates.

Syslog

The LioN-X multiprotocol variants support the traceability of messages centrally managed and logged via Syslog.

User manager

The Web server provides a user manager to protect the Web interface against unauthorized access. You can manage the allowed users by different access levels “Admin” or “Write”.

Default user settings:

User: admin

Password: private



Attention: Change the default settings to protect the device against unauthorized access.

5.5 Other features

Interface protection

The modules have reverse polarity, short-circuit and overload protection for all interfaces.

For more details, see section [Port assignments](#) on page 31.

Failsafe

The modules support a failsafe function. This allows you to define the behavior of every single channel configured as an output in the case of a loss of the PLC communication.

Industrial Internet of Things

LioN-X is industry 4.0 ready and supports the integration in IIoT networks via REST API and the IIoT-relevant protocols MQTT, OPC UA and CoAP.

Color-coded connectors

The green-colored connectors help you avoid confusion in your cabling.

IP protection classes: IP65 / IP67 / IP69k




The IP protection class describes environmental influences that the devices can be exposed to without risk and without suffering damage, or causing a risk for the user.

The whole LioN-X family offers IP65, IP67 and IP69k.

6 Assembly and wiring

6.1 General information

Mount the device on a flat surface using 2 screws (M4x 25/30). The torque required here is 1 Nm. Use washers for all fastening methods as per DIN 125.

-  **Attention:** The devices have a ground connection with an M4 thread for the conduction of interference currents and the EMC immunity. This is labeled with the symbol for the ground and the designation "FE"
-  **Attention:** Use a low-impedance connection to connect the device to the reference ground. When using a grounded mounting surface, you can make the connection directly via the fixing screws.
-  **Attention:** If the mounting surface is ground-free, use a ground strap or a suitable FE line (FE = Functional Earth). Use an M4 screw to connect the ground strap or the FE line to the ground point and if possible put a washer and a toothed washer below the fixing screw.

6.2 Outer dimensions

6.2.1 Lion-X multiprotocol variants

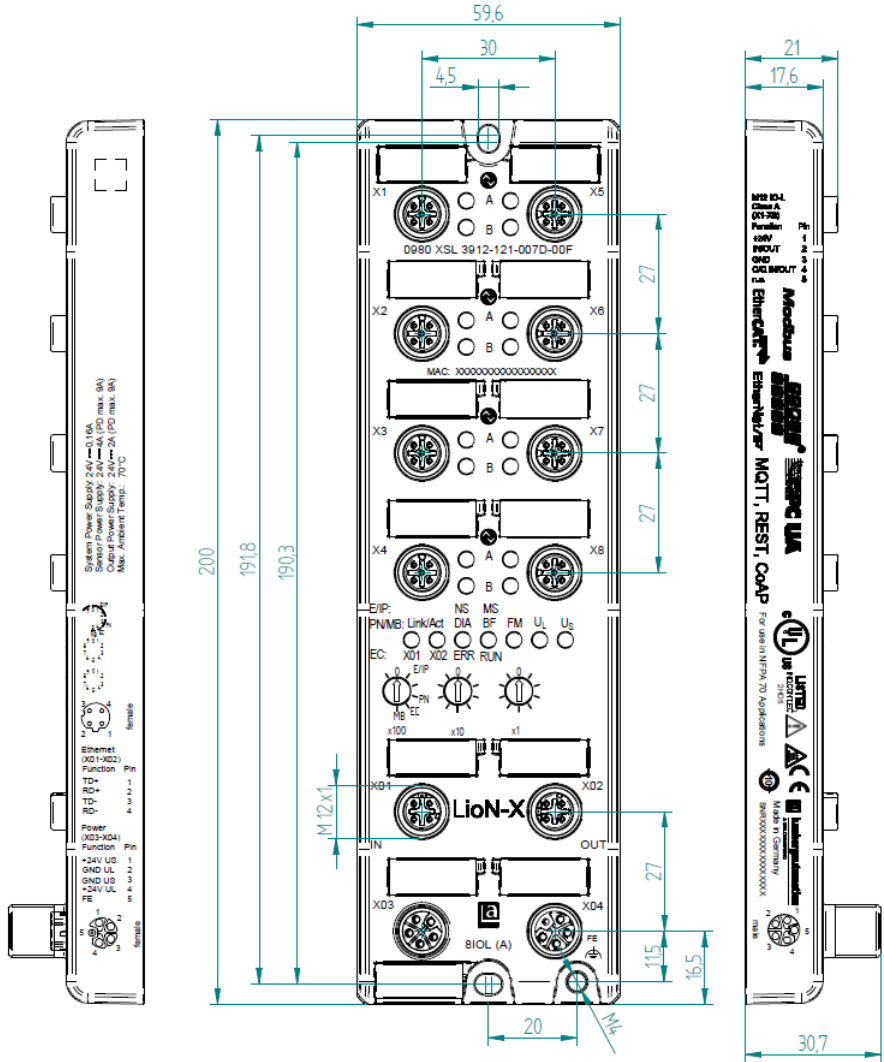


Figure 1: 0980 XSL 3912-121-007D-00F

6.2.2 Lion-Xlight variants with EtherCAT®

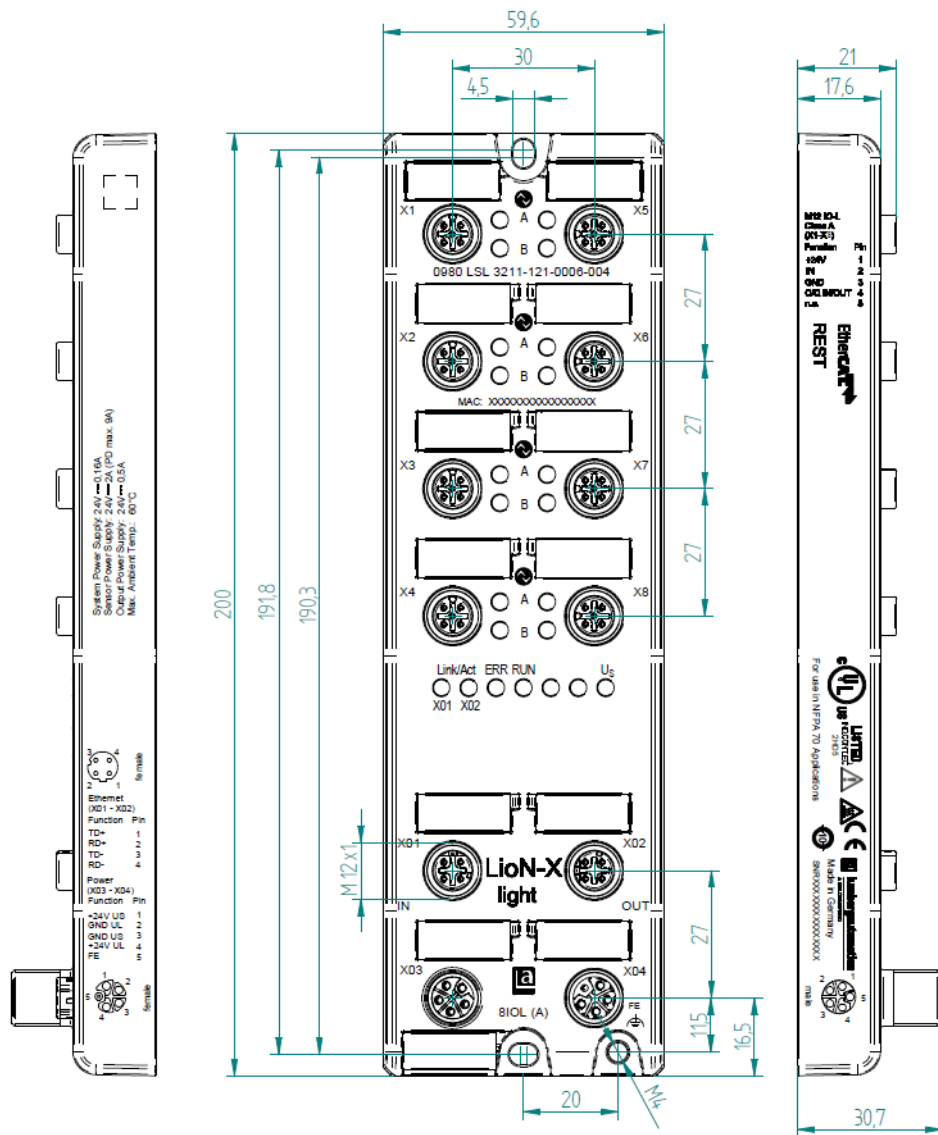


Figure 2: 0980 LSL 3211-121-0006-004

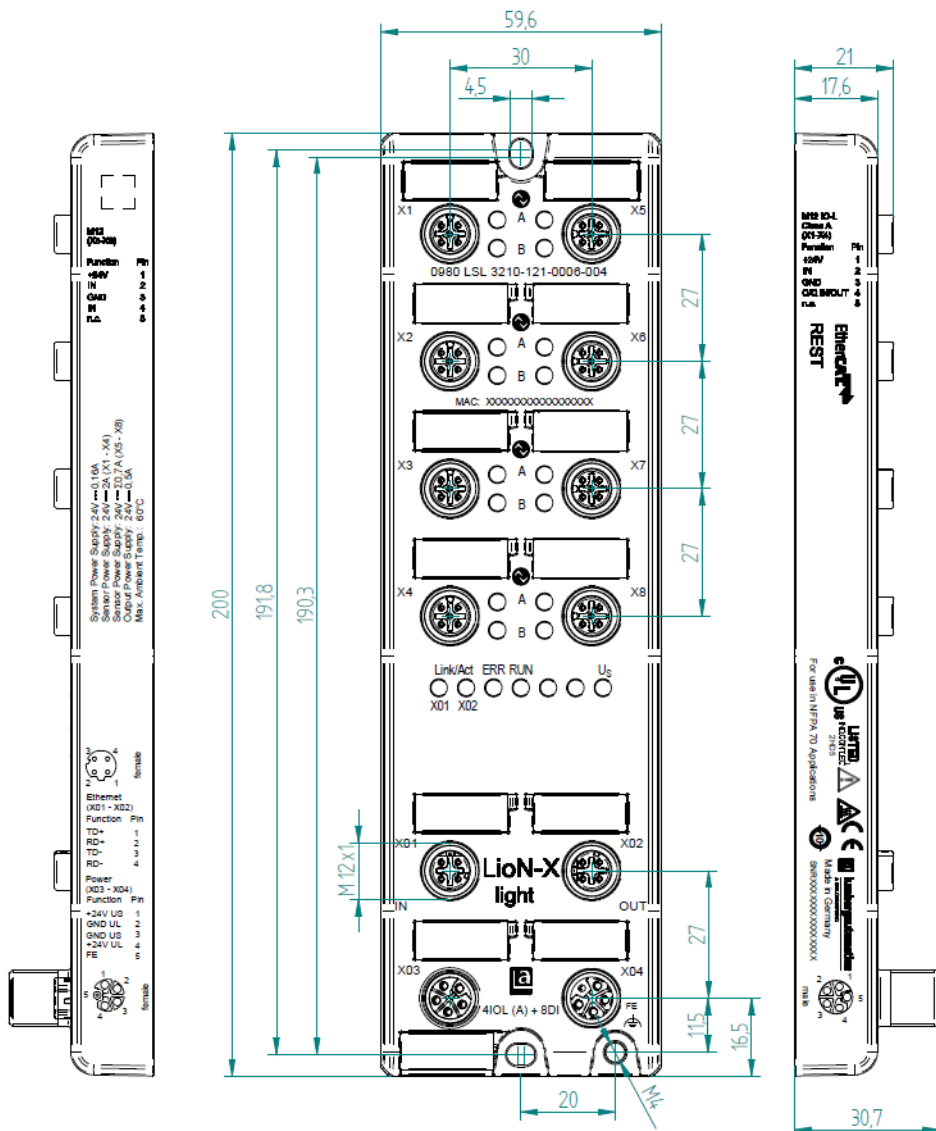


Figure 3: 0980 LSL 3210-121-0006-004

6.2.3 Notifications

**Attention:**

For **UL applications**, be sure to use a UL-certified cable with a suitable evaluation to connect the devices (CYJV or PVVA). To program the control, please refer to the OEM information, and only use suitable accessories.

Only approved for interior use. Please note the maximum elevation of 2000 meters. Approved up to a maximum soiling level of 2.



Warning: Terminals, housings field-wired terminal boxes or components can exceed temperatures of +60 °C (140 °F).



Warning: For **UL applications** at a maximum ambient temperature of +70 °C (158 °F):

Use temperature-resistant cables with heat resistance up to at least +115 °C (239 °F) for all LioN-X and LioN-Xlight variants.



Warning: Observe the following maximum output power for the sensor supply:

Max. 2.0 A per channel; max. 6.5 A in total (for **UL applications** max. 5 A in total) for every port pair (X1/X2, X3/X4, X5/X6, X7/X8); max. 9.0 A in total (with derating) for the whole port group (X1 .. X8).

6.3 Port assignments

All the contact arrangements shown in this chapter show the frontal view of the connection area for the connectors.

6.3.1 Ethernet ports, M12 socket, 4-pin, D-coded

Color coding: green

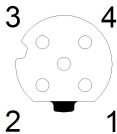


Figure 4: Schematic drawing, ports X01, X02

Port	Pin	Signal	Function
Ethernet Ports X01, X02	1	TD+	Transmit data plus
	2	RD+	Receive data plus
	3	TD-	Transmit data minus
	4	RD-	Receive data minus

Table 7: Assignment of ports X01, X02



Caution: Risk of destruction! Never connect the power supply to the data cables.

6.3.2 Power supply with M12 power L-coded

Color coding: gray

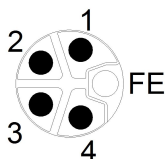


Figure 5: Schematic diagram of the M12 L-coding (connector X03 for Power In)

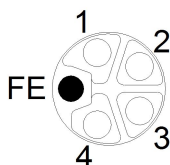


Figure 6: Schematic diagram of the M12 L-coding (socket X04 for Power Out)

Power supply	Pin	Signal	Function
	1	U_S (+24 V)	Sensor/system power supply
	2	GND_ U_L	Ground/reference potential U_L
	3	GND_ U_S	Ground/reference potential U_S ¹
	4	U_L (+24 V)	Load supply (NOT electrically isolated to U_S internally in device)
	5	FE (PE)	Functional ground

Table 8: Power supply with M12-Power

i Attention: Only use power supply units for the system/sensor and actuator supply that correspond to PELV (Protective Extra Low Voltage) or SELV (Safety Extra Low Voltage). Power supplies

¹ U_L and U_S ground connected in device

according to EN 61558-2-6 (transformers) or EN 60950-1 (switching power supply units) fulfill these requirements.

6.3.3 I/O ports as M12 sockets

6.3.3.1 IO-Link Class A

Color coding: black

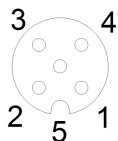


Figure 7: Schematic drawing I/O port as M12 socket IO-Link Class A

0980 XSL 3x12-121...	Pin	Signal	Function
IO-Link Class A, ports X1 - X8	1	+24 V	power supply +24 V
	2	IN/OUT	Ch. B: Digital inputs or digital outputs
	3	GND	Ground/reference potential
	4	C/Q IN/OUT	Ch. A: IO-Link data communication, digital input or digital output
	5	n.c.	not connected
0980 LSL 3x11-121...	Pin	Signal	Function
IO-Link Class A, ports X1 - X8	1	+24 V	power supply +24 V
	2	IN	Ch. B: Digital inputs
	3	GND	Ground/reference potential
	4	C/Q IN/OUT	Ch. A: IO-Link data communication, digital input or digital output
	5	n.c.	not connected
0980 LSL 3x10-121...	Pin	Signal	Function
IO-Link Class A, ports X1 - X4	1	+24 V	power supply +24 V
	2	IN	Ch. B: Digital inputs
	3	GND	Ground/reference potential
	4	C/Q IN/OUT	Ch. A: IO-Link data communication, digital input or digital output
	5	n.c.	not connected
IO-Link Class A, ports X5 - X8	1	+24 V	power supply +24 V
	2	IN	Ch. B: Digital inputs
	3	GND	Ground/reference potential
	4	C/Q IN	Ch. A: IO-Link data communication, digital input
	5	n.c.	not connected

Table 9: I/O ports as M12 socket IO-Link Class A

7 Starting operation

7.1 ESI file

An ESI file in XML format is required to configure the LioN-X and LioN-Xlight devices. All device variants are grouped in a single ESI file. The file can be downloaded from the product pages on our online catalog: catalog.belden.com

On request, the ESI file is also sent by the support team.

The file for the LioN-X EtherCAT® devices is named:

LumbergAutomation-LioN-X-IO-Link-Master.xml

Download this file and unpack it.

Install the ESI file for the device variant used via the hardware or network configuration tool of your controller manufacturer.

For TwinCAT®, the ESI file has to be copied into the installation folder, e.g.:
C:\TwinCAT\3.1\Config\Io\EtherCAT

After installation, TwinCAT® needs a system restart. Alternatively, use the menu bars in TwinCAT® to reload the application:

TWINCAT > EtherCAT Devices > Reload Device Descriptions.

The LioN-X and LioN-Xlight devices are then available in the hardware catalog.

7.2 MAC addresses

Every device has three unique MAC address assigned by the manufacturer that cannot be changed by the user. The first assigned MAC address is printed onto the device.

For EtherCAT[®], the MAC address has no function. For EoE (Ethernet over EtherCAT[®]), a virtual MAC address will be assigned to the device.

7.3 Setting the rotary encoding switches

i Attention: Only applicable for LioN-X multiprotocol variants; not applicable for LioN-Xlight variants.

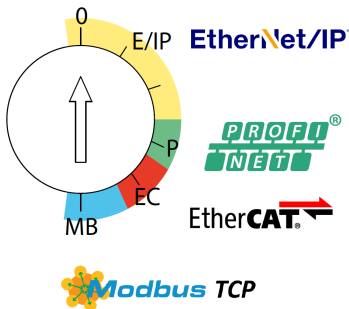
The LioN-X multiprotocol variants allow you to select different protocols for communication within an industrial Ethernet system. In this way the IO-Link Masters with multiprotocol function can be integrated into different networks without it being necessary to purchase products specific for each protocol. This technology also gives you the option to use the same IOL-Master in different environments.

Using rotary encoding switches at the lower front of the devices, you can easily and conveniently set both the protocol and the address of the device, if the protocol to be used supports this. Once you have made a protocol selection and started the cyclical communication, the device stores this setting permanently and uses the selected protocol from this point on. To use another supported protocol with this device, perform a factory reset.

The following LioN-X IO-Link Master variants support multiprotocol application for the protocols EtherNet/IP (E/IP), PROFINET (P), EtherCAT® (EC) and Modbus TCP (MB):

► 0980 XSL 3912-121-007D-00F

The multiprotocol devices have a total of three rotary encoding switches. With the first rotary encoding switch (x100) you set the protocol by using the corresponding switch position. Additionally, x100 is used to set the third last digit of the IP address for EIP.



With the other rotary encoding switches (x10 / x1), you set the last two digits of the IP address when you are using EtherNet/IP or Modbus TCP.

Protocol	x100	x10	x1
EtherNet/IP	0-2	0-9	0-9
PROFINET	P	–	–
EtherCAT®	EC	–	–
Modbus TCP	MB	0-9	0-9

Table 10: Assignment of the rotary encoding switches for each protocol

The setting you make to select a protocol is described detailed in the protocol-specific sections.

In delivery state no protocol settings are stored in the device. In this case only the desired protocol has to be chosen. To take over a changed rotary encoding switch setting (protocol setting), a power cycle or “Reset” from the Web interface is necessary.

Once you have set the protocol using the rotary encoding switches, the device stores this setting when it starts in cyclic communication. Changing the protocol using the rotary encoding switch is no longer possible after this point. The device will always start using the stored protocol from that point on. The IP address can be changed depending on the selected protocol.

To change the protocol, carry out a factory reset. In this way you restore the factory settings of the respective device. How you perform the factory reset for your device is described in chapter [Factory reset](#) on page 39.

If you position the rotary encoding switch in a manner that is invalid, the device signals this to you with a blink code (the BF/MS LED blinks in red three times).

7.3.1 Factory reset

A factory reset restores the original factory settings and thus resets the changes and settings you have made up to that point. It also resets the protocol selection. To perform a factory reset, set the first rotary encoding switch (x100) to 9, the second (x10) to 7, and the third (x1) also to 9.

Afterwards perform a power cycle and wait 10 seconds due to internal memory write processes.

During the factory reset, the U_S LED is blinking red. After the internal memory write processes have finished, the U_S LED returns to display static green or red light, in dependency of the actual U_S voltage.

	x100	x10	x1
Factory Reset	9	7	9

Follow the steps from section [Setting the rotary encoding switches](#) on page 37 again to select a new protocol.

For performing a factory reset via software configuration, see chapter [OPC UA configuration](#) on page 106 and the configuration section.

8 Configuration and operation in TwinCAT® 3

8.1 PDO assignments

The LioN-X and LioN-Xlight devices support different PDO (Process Data Object) assignments for input and output data.

By selecting the relevant PDO, you can decide your preferred I/O data content. The devices feature a dynamic, slot-based PDO assignment. The following PDO assignments are provided:

8.1.1 Input data

PDO 0x1A00

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A00	32	0x6000:1	1	UINT32	1st byte of IO-Link input data
	
		0x6000:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A01

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A01	32	0x6010:1	1	UINT32	1st byte of IO-Link input data
	
		0x6010:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A02

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A02	32	0x6020:1	1	UINT32	1st byte of IO-Link input data
	
		0x6020:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A03

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A03	32	0x6030:1	1	UINT32	1st byte of IO-Link input data
	
		0x6030:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A04

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A04	32	0x6040:1	1	UINT32	1st byte of IO-Link input data
	
		0x6040:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A05

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A05	32	0x6050:1	1	UINT32	1st byte of IO-Link input data
	
		0x6050:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A06

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A06	32	0x6060:1	1	UINT32	1st byte of IO-Link input data
	
		0x6060:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A07

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A07	32	0x6070:1	1	UINT32	1st byte of IO-Link input data
	
		0x6070:32	1	UINT32	32nd byte of IO-Link input data

PDO 0x1A08

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A08	4	0x2080:1	1	UINT32	Digital input status of ports X1 - X4 Bitwise
		0x2080:2	1	UINT32	Digital input status of ports X5 – X8 Bitwise
		0x2080:3	1	UINT32	State of IO-Link Communication Bitwise
		0x2080:4	1	UINT32	Status of IO-Link Process Data valid Bitwise

PDO 0x1A09

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A09	1	0x10F3: 4	1	UINT32	Flag "New Messages Available" from Diagnostic Object 0x10F3

PDO 0x1A0A

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A0A	1	0x10F8: 0	1	UINT32	Timestamp Object 0x10F8

PDO 0x1A80

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A80	8	0xF100:1	1	UINT32	Status IO-Link port 1
		0xF100:2	1	UINT32	Status IO-Link port 2
		0xF100:3	1	UINT32	Status IO-Link port 3
		0xF100:4	1	UINT32	Status IO-Link port 4
		0xF100:5	1	UINT32	Status IO-Link port 5
		0xF100:6	1	UINT32	Status IO-Link port 6
		0xF100:7	1	UINT32	Status IO-Link port 7
		0xF100:8	1	UINT32	Status IO-Link port 8

8.1.2 Output data

PDO 0x1600

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A00	32	0x7000:1	1	UINT32	1st byte of IO-Link output data
	
		0x7000:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1601

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A01	32	0x7010:1	1	UINT32	1st byte of IO-Link output data
	
		0x7010:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1602

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A02	32	0x7020:1	1	UINT32	1st byte of IO-Link output data
	
		0x7020:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1603

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1A03	32	0x7030:1	1	UINT32	1st byte of IO-Link output data
	
		0x7030:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1604

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1604	32	0x7040:1	1	UINT32	1st byte of IO-Link output data
	
		0x7040:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1605

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1605	32	0x7050:1	1	UINT32	1st byte of IO-Link output data
	
		0x7050:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1606

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1606	32	0x7060:1	1	UINT32	1st byte of IO-Link output data
	
		0x7060:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1607

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1607	32	0x7070:1	1	UINT32	1st byte of IO-Link output data
	
		0x7070:32	1	UINT32	32nd byte of IO-Link output data

PDO 0x1608

PDO		PDO content			
Index	Size	Index	Size	Type	Name
0x1608	4	0x2280:1	1	UINT32	Digital output mapping for ports X1 - X4 Bitwise
		0x2280:2	1	UINTtwinCAT32	Digital output mapping for ports X5 – X8 Bitwise
		0x2280:3	1	UINT32	Control of IO-Link COM mode Bitwise
		0x2280:4	1	UINT32	Reserved

8.1.3 Modular slots

The ESI file features a modular slot-based PDO configuration for different IO-Link configurations. The following slots are available:

Slot name	Description
STD_IN_1_bit	IO-Link port as standard Digital Input
IOL_I_1byte	IO-Link, 1 byte of process data input
IOL_I_2byte	IO-Link, 2 bytes of process data input
IOL_I_4byte	IO-Link, 4 bytes of process data input
IOL_I_6byte	IO-Link, 6 bytes of process data input
IOL_I_8byte	IO-Link, 8 bytes of process data input
IOL_I_10byte	IO-Link, 10 bytes of process data input
IOL_I_16byte	IO-Link, 16 bytes of process data input
IOL_I_24byte	IO-Link, 24 bytes of process data input
IOL_I_32byte	IO-Link, 32 bytes of process data input
STD_OUT_1_bit	IO-Link port as standard Digital Output
IOL_O_1byte	IO-Link, 1 bytes of process data output
IOL_O_2byte	IO-Link, 2 bytes of process data output
IOL_O_4byte	IO-Link, 4 bytes of process data output
IOL_O_6byte	IO-Link, 6 bytes of process data output
IOL_O_8byte	IO-Link, 8 bytes of process data output
IOL_O_10byte	IO-Link, 10 bytes of process data output
IOL_O_16byte	IO-Link, 16 bytes of process data output
IOL_O_24byte	IO-Link, 24 bytes of process data output
IOL_O_32byte	IO-Link, 32 bytes of process data output
IOL_I/O_1/1byte	IO-Link, 1 byte of process data input 1 byte of process data output

Slot name	Description
IOL_I/O_2/2byte	IO-Link, 2 bytes of process data input 2 bytes of process data output
IOL_I/O_2/4byte	IO-Link, 2 bytes of process data input 4 bytes of process data output
IOL_I/O_4/4byte	IO-Link, 4 bytes of process data input 4 bytes of process data output
IOL_I/O_4/2byte	IO-Link, 4 bytes of process data input 2 bytes of process data output
IOL_I/O_2/8byte	IO-Link, 2 bytes of process data input 8 bytes of process data output
IOL_I/O_4/8byte	IO-Link, 4 bytes of process data input 8 bytes of process data output
IOL_I/O_8/2byte	IO-Link, 8 bytes of process data input 2 bytes of process data output
IOL_I/O_8/4byte	IO-Link, 8 bytes of process data input 4 bytes of process data output
IOL_I/O_8/8byte	IO-Link, 8 bytes of process data input 8 bytes of process data output
IOL_I/O_4/32byte	IO-Link, 4 bytes of process data input 32 bytes of process data output
IOL_I/O_32/4byte	IO-Link, 32 bytes of process data input 4 bytes of process data output

Slot name	Description
IOL_I/O_16/16byte	IO-Link, 16 bytes of process data input 16 bytes of process data output
IOL_I/O_24/24byte	IO-Link, 24 bytes of process data input 24 bytes of process data output
IOL_I/O_32/32byte	IO-Link, 32 bytes of process data input 32 bytes of process data output

8.2 Device parameters

The LioN-X and LioN-Xlight devices support different parameters. The parameters must be transferred from the controller to the device during startup. The following blocks of parameters can be adjusted.

8.2.1 Extended parameters

The following table represents extended parameters of the IO-Link ports.

SDO		SDO content			
Index	Size	Index	Size	Type	Name (specification)
0x23n0	2	0x23n0:1	1	UINT8	Data Storage 0 = Data Storage disabled (stored data set is preserved) 1 = Data Storage download only (IOLM → IOLD) 2 = Data Storage upload only (IOLD → IOLM) 3 = Data Storage download and upload 4 = Data Storage disabled and cleared (a previously stored data set is cleared) Others: reserved
		0x23n0:2	1	UINT8	Fail Safe Mode 0 = Set Low 1 = Set High 2 = Hold Last Value 3 = Replacement value (refer object 0x2301) 4 = IO-Link master command Others: reserved

Key:

n = number between 0..7 (port number - 1)

8.2.2 Failsafe replacement values

The firmware of the devices provides a fail-safe function for the port while in IO-Link mode. During device configuration, you have the option to define the replacement value of the IO-Link output data.

The following table represents failsafe replacement values of IO-Link output data while the port is in IO-Link mode.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x23n1	2	0x23n1:1	1	UINT8	Failsafe replacement value byte 1
	
		0x23n1:32	32	UINT8	Failsafe replacement value byte 32

Key:

n = number between 0..7 (port number - 1)

8.2.3 Failsafe mode for the digital output

The firmware of the devices provides a fail-safe function for ports in digital output mode. During device configuration, you have the option to define the status of channels A and B for ports in digital output mode in the case of an interruption or the loss of communication.

The following table represents possible failsafe replacement values of ports in digital output mode.

The following options are available:

- Set low (default value)** Deactivate the output channel
(value = 0)
- Set high** Activate the output channel
(value = 1)
- Hold last** Hold the last output status
(value = 2)

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2380	16	0x2380 : 1	1	UINT8	Port 1 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 2	1	UINT8	Port 1 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 3	1	UINT8	Port 2 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved

SDO		SDO content			
		0x2380 : 4	1	UINT8	Port 2 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 5	1	UINT8	Port 3 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 6	1	UINT8	Port 3 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 7	1	UINT8	Port 4 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 8	1	UINT8	Port 4 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 9	1	UINT8	Port 5 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 10	1	UINT8	Port 5 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved

SDO		SDO content			
		0x2380 : 11	1	UINT8	Port 6 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 12	1	UINT8	Port 6 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 13	1	UINT8	Port 7 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 14	1	UINT8	Port 7 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 15	1	UINT8	Port 8 Channel A 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved
		0x2380 : 16	1	UINT8	Port 8 Channel B 0 = Set Low 1 = Set High 2 = Hold Last Others: reserved

8.2.4 General device settings

The device supports the setting of different parameters. The following blocks of parameters can be adjusted:

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2381	7	0x2381: 1	1	BOOL	Web Interface Locked 0 = false, 1 = true
		0x2381: 2	1	BOOL	Force Mode Locked
		0x2381: 3	1	BOOL	Disable All Emergency Messages 0 = false, 1 = true
		0x2381: 4	1	BOOL	Disable UL Emergency Messages 0 = false, 1 = true
		0x2381: 5	1	BOOL	Disable Actuator Emergency Messages 0 = false, 1 = true
		0x2381: 6	1	BOOL	Enable External Configuration 0 = false, 1 = true
		0x2381: 7	1	BOOL	Automatic Output Restart after failure 0 = false, 1 = true

8.2.5 Surveillance timeout

The firmware of the devices allows you to define a delay time before the automatic monitoring of the output currents. This is known as the surveillance timeout.

You can define the surveillance timeout for every individual output channel.

The delay time begins after the output channel has been activated (after a rising edge) or deactivated (after a falling edge). After the surveillance timeout has elapsed, the monitoring of the output begins and the diagnosis will report error states.

The value of the surveillance timeout is 0 to 255ms. The default value is 80ms. While an output channel is in static state (permanently switched on or off), the respective value is 100ms.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2382	16	0x2382 : 1	1	UINT8	Surveillance Timeout Port 1 Channel A
		0x2382 : 2	1	UINT8	Surveillance Timeout Port 1 Channel B
		0x2382 : 3	1	UINT8	Surveillance Timeout Port 2 Channel A
		0x2382 : 4	1	UINT8	Surveillance Timeout Port 2 Channel B
		0x2382 : 5	1	UINT8	Surveillance Timeout Port 3 Channel A
		0x2382 : 6	1	UINT8	Surveillance Timeout Port 3 Channel B
		0x2382 : 7	1	UINT8	Surveillance Timeout Port 4 Channel A
		0x2382 : 8	1	UINT8	Surveillance Timeout Port 4 Channel B

SDO		SDO content			
		0x2382: 9	1	UINT8	Surveillance Timeout Port 5 Channel A
		0x2382: 10	1	UINT8	Surveillance Timeout Port 5 Channel B
		0x2382: 11	1	UINT8	Surveillance Timeout Port 6 Channel A
		0x2382: 12	1	UINT8	Surveillance Timeout Port 6 Channel B
		0x2382: 13	1	UINT8	Surveillance Timeout Port 7 Channel A
		0x2382: 14	1	UINT8	Surveillance Timeout Port 7 Channel B
		0x2382: 15	1	UINT8	Surveillance Timeout Port 8 Channel A
		0x2382: 16	1	UINT8	Surveillance Timeout Port 8 Channel B

8.2.6 Digital I/O mode, Channel B

The device supports the configuration of channel B (Pin 2) of the IO-Link port via SDO 0x2383 in input or output mode.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2383	8	0x2383 : 1	1	UINT8	Digital I/O Mode Port 1 Channel B 1 = Input 2 = Output Others = reserved
		0x2383: 2	1	UINT8	Digital I/O Mode, Port 2 Channel B 1 = Input 2 = Output Others = reserved
		0x2383: 3	1	UINT8	Digital I/O Mode, Port 3 Channel B 1 = Input 2 = Output Others = reserved
		0x2383: 4	1	UINT8	Digital I/O Mode, Port 4 Channel B 1 = Input 2 = Output Others = reserved
		0x2383: 5	1	UINT8	Digital I/O Mode, Port 5 Channel B 1 = Input 2 = Output Others = reserved
		0x2383: 6	1	UINT8	Digital I/O Mode, Port 6 Channel B 1 = Input 2 = Output Others = reserved
		0x2383: 7	1	UINT8	Digital I/O Mode, Port 7 Channel B 1 = Input 2 = Output Others = reserved

SDO		SDO content			
		0x2383: 8	1	UINT8	Digital I/O Mode, Port 8 Channel B 1 = Input 2 = Output Others = reserved

8.2.7 Digital Input logic

The device supports the configuration of digital input logic for Channel A (Pin 4) and Channel B (Pin 2) of the IO-Link port.

The following values are only applicable for ports in **digital input mode**.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2384	16	0x2384 : 1	1	UINT8	Digital Input logic Port 1 Channel A 0: NO 1: NC
		0x2384: 2	1	UINT8	Digital Input logic Port 1 Channel B 0: NO 1: NC
		0x2384: 3	1	UINT8	Digital Input logic Port 2 Channel A 0: NO 1: NC
		0x2384: 4	1	UINT8	Digital Input logic Port 2 Channel B 0: NO 1: NC
		0x2384: 5	1	UINT8	Digital Input logic Port 3 Channel A 0: NO 1: NC
		0x2384: 6	1	UINT8	Digital Input logic Port 3 Channel B 0: NO 1: NC
		0x2384: 7	1	UINT8	Digital Input logic Port 4 Channel A 0: NO 1: NC

SDO		SDO content			
		0x2384: 8	1	UINT8	Digital Input logic Port 4 Channel B 0: NO 1: NC
		0x2384: 9	1	UINT8	Digital Input logic Port 5 Channel A 0: NO 1: NC
		0x2384: 10	1	UINT8	Digital Input logic Port 5 Channel B 0: NO 1: NC
		0x2384: 11	1	UINT8	Digital Input logic Port 6 Channel A 0: NO 1: NC
		0x2384: 12	1	UINT8	Digital Input logic Port 6 Channel B 0: NO 1: NC
		0x2384: 13	1	UINT8	Digital Input logic Port 7 Channel A 0: NO 1: NC
		0x2384: 14	1	UINT8	Digital Input logic Port 7 Channel B 0: NO 1: NC
		0x2384: 15	1	UINT8	Digital Input logic Port 8 Channel A 0: NO 1: NC
		0x2384: 16	1	UINT8	Digital Input logic Port 8 Channel B 0: NO 1: NC

8.2.8 Digital Input filter

The device supports the configuration of a digital input filter for Channel A (Pin 4) and Channel B (Pin 2) of the IO-Link port.

The following values are only applicable for ports in **digital input mode**.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2385	16	0x2385 : 1	1	UINT8	Digital Input Filter Port 1 Channel A
		0x2385: 2	1	UINT8	Digital Input Filter Port 1 Channel B
		0x2385: 3	1	UINT8	Digital Input Filter Port 2 Channel A
		0x2385: 4	1	UINT8	Digital Input Filter Port 2 Channel B
		0x2385: 5	1	UINT8	Digital Input Filter Port 3 Channel A
		0x2385: 6	1	UINT8	Digital Input Filter Port 3 Channel B
		0x2385: 7	1	UINT8	Digital Input Filter Port 4 Channel A
		0x2385: 8	1	UINT8	Digital Input Filter Port 4 Channel B
		0x2385: 9	1	UINT8	Digital Input Filter Port 5 Channel A
		0x2385: 10	1	UINT8	Digital Input Filter Port 5 Channel B
		0x2385: 11	1	UINT8	Digital Input Filter Port 6 Channel A
		0x2385: 12	1	UINT8	Digital Input Filter Port 6 Channel B
		0x2385: 13	1	UINT8	Digital Input Filter Port 7 Channel A
		0x2385: 14	1	UINT8	Digital Input Filter Port 7 Channel B
		0x2385: 15	1	UINT8	Digital Input Filter Port 8 Channel A
		0x2385: 16	1	UINT8	Digital Input Filter Port 8 Channel B

8.2.9 Digital Output restart

The device supports the configuration of a digital output timeout before restart for Channel A (Pin 4) and Channel B (Pin 2) of the IO-Link port.

The following values are only applicable for ports in **digital output mode**.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x2386	16	0x2386 : 1	1	BOOL	Digital output timeout before restart in ms Port 1 Channel A
		0x2386: 2	1	BOOL	Digital output timeout before restart in ms Port 1 Channel B
		0x2386: 3	1	BOOL	Digital output timeout before restart in ms Port 2 Channel A
		0x2386: 4	1	BOOL	Digital output timeout before restart in ms Port 2 Channel B
		0x2386: 5	1	BOOL	Digital output timeout before restart in ms Port 3 Channel A
		0x2386: 6	1	BOOL	Digital output timeout before restart in ms Port 3 Channel B
		0x2386: 7	1	BOOL	Digital output timeout before restart in ms Port 4 Channel A
		0x2386: 8	1	BOOL	Digital output timeout before restart in ms Port 4 Channel B
		0x2386: 9	1	BOOL	Digital output timeout before restart in ms Port 5 Channel A
		0x2386: 10	1	BOOL	Digital output timeout before restart in ms Port 5 Channel B
		0x2386: 11	1	BOOL	Digital output timeout before restart in ms Port 6 Channel A

SDO		SDO content			
		0x2386: 12	1	BOOL	Digital output timeout before restart in ms Port 6 Channel B
		0x2386: 13	1	BOOL	Digital output timeout before restart in ms Port 7 Channel A
		0x2386: 14	1	BOOL	Digital output timeout before restart in ms Port 7 Channel B
		0x2386: 15	1	BOOL	Digital output timeout before restart in ms Port 8 Channel A
		0x2386: 16	1	BOOL	Digital output timeout before restart in ms Port 8 Channel B

8.2.10 Additional IO-Link port settings

The device firmware provides the following additional setting options for each individual IO-Link port via SDO.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x30n0	6	0x30n0:1	1	UINT8	Swap Mode
		0x30n0:2	1	UINT8	Swap Length
		0x30n0:3	1	UINT8	Swap Offset
		0x30n0:4	1	BOOL	Sensor Supply Enabled 0 = false, 1 = true
		0x30n0:5	1	BOOL	Pin 2 LED Enabled 0 = false, 1 = true
		0x30n0:6	1	BOOL	Supress All Diagnosis 0 = false, 1 = true
		0x30n0:7	1	BOOL	Pin 4 DO use of Push-Pull 0 = Use High-Side switches 1 = Use Push-Pull
		0x30n0:8	1	UINT16	Pin 4 current limit in mA (Maximum current limit till Pin 4 is turned off) Default: 2000
		0x30n0:9	1	UINT16	Pin 2 DO use of Push-Pull in mA (Maximum current limit till Pin 2 is turned off) Default: 2000

Key:

n = number between 0..7 (port number - 1)

Swap Mode

The byte order swapping will be made for the selected count of data types or for the complete length of I/O-Data with the selected Data Type (Word = 2 Bytes or DWord = 4 Bytes).

Swap Length

The swapping can be setup to Word (2 Bytes) or DWord (4 Bytes)

→ Word Swapping: Byte 1 - Byte 2 => Byte 2 - Byte 1

→ DWord Swapping: Byte 1 - Byte 4 => Byte 4 - Byte 1

Swapping Offset

A swapping offset in bytes can be setup in dependency of the configured I/O-Data length. When setup to "2", the swapping will be processed from the 3rd Byte.

8.2.11 IO-Link Parameterization

The device features read or write of ISDU (IO-Link Service Data) parameters via SDO 0x40n0.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x40n0	7	0x40n0:1	1	UINT8	Control 0x00: no action 0x02: write 0x03: read
		0x40n0:2	1	UINT8	Status 0x00: no activity 0x01: active / busy 0x02: access 0x04: error 0xFF: failure
		0x40n0:3	1	UINT8	Index
		0x40n0:4	2	UINT16	Subindex
		0x40n0:5	1	UINT8	Length
		0x40n0:6	232	UINT232	Data
		0x40n0:6	2	UINT16	Error Code

Key:

n = number between 0..7 (port number - 1)

8.2.12 IO-Link configuration data

This is the standard 0x80n0 object for the IO-Link configuration data as per the IO-Link device profile.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x80n0	7	0x80n0:4	4	UINT32	Device ID
		0x80n0:5	4	UINT32	Vendor ID
		0x80n0:32	1	UINT8	IO LINK revision Version of the implemented IO-Link specification (of the connected IO-Link device) according to version 1.0 of the IO-Link specification. Bit 0...3: Minor Rev Bit 4...7: Major Rev
		0x80n0:33	1	UINT8	Frame Capability
		0x80n0:34	1	UINT8	Cycle Time
		0x80n0:35	1	UINT8	Offset Time
		0x80n0:36	1	UINT8	Process data IN length Number and structure of input data This value is transmitted in IO-Link format as "ProcessDataIn" according to version 1.0 of the IO-Link specification. Bit 0...4: Length Bit 5: reserved Bit 6: SIO Indicator (if device supports standard I/O mode) Bit 7: Byte Indicator (if value of length is interpreted as bit length or as byte length +1)
		0x80n0:37	1	UINT8	Process data OUT length Number and structure of input data This value is transmitted in IO-Link format as "ProcessDataOut" according to version 1.0 of the IO-Link specification. Bit 0...4: Length Bit 5: reserved Bit 6: SIO Indicator (if device supports standard I/O mode) Bit 7: Byte Indicator (if value of length is interpreted as bit length or as byte length +1)

SDO		SDO content			
		0x80n0:38	2	UINT16	Compatible ID
		0x80n0:39	2	UINT16	Reserved
		0x80n0:40	2	UINT16	Master Control Control of the IO-Link master port. Defines the different operating modes of the IO-Link master. Bits 0...3: 0: Inactive 1: Digital Input Port 2: Digital Output Port 3: Communication over IO-Link Protocol 4: Digital input with support for acyclic IO-Link

Key:

n = number between 0..7 (port number - 1)

8.2.13 IO-Link serial number

This is the standard 0x80n1 object for the IO-Link serial number as per the IO-Link device profile.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x80n1	20	0x80n1:1	1	VISIBLE_STRING	Serial Number

Key:

n = number between 0..7 (port number - 1)

8.2.14 IO-Link information data

This is standard 0x90n0 object for the IO-Link information data as per the IO-Link device profile.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x90n0	7	0x90n0:4	4	UINT32	Device ID
		0x90n0:5	4	UINT32	Vendor ID
		0x90n0:32	1	UINT8	IO LINK revision Version of the implemented IO-Link specification (of the connected IO-Link device) according to version 1.0 of the IO-Link specification. Bit 0...3: Minor Rev Bit 4...7: Major Rev
		0x90n0:33	1	UINT8	Frame Capability
		0x90n0:34	1	UINT8	Cycle Time
		0x90n0:35	1	UINT8	Offset Time
		0x90n0:36	1	UINT8	Process data IN length Number and structure of input data This value is transmitted in IO-Link format as "ProcessDataIn" according to version 1.0 of the IO-Link specification. Bit 0...4: Length Bit 5: reserved Bit 6: SIO Indicator (if device supports standard I/O mode) Bit 7: Byte Indicator (if value of length is interpreted as bit length or as byte length +1)

SDO		SDO content			
		0x90n0:37	1	UINT8	Process data OUT length Number and structure of input data This value is transmitted in IO-Link format as "ProcessDataOut" according to version 1.0 of the IO-Link specification. Bit 0...4: Length Bit 5: reserved Bit 6: SIO Indicator (if device supports standard I/O mode) Bit 7: Byte Indicator (if value of length is interpreted as bit length or as byte length +1)

Key:

n = number between 0..7 (port number - 1)

8.2.15 IO-Link serial number of connected devices

This is the standard 0x90n1 object for the IO-Link serial number of connected devices as per the IO-Link device profile.

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x9n01	20	0x90n1:1	1	VISIBLE_STRING	Serial Number

Key:

n = number between 0..7 (port number - 1)

8.3 Configuration example with TwinCAT® 3

The configuration and start-up of the devices described below refer to the TwinCAT® 3 software by Beckhoff Automation GmbH & Co. KG. If you use the control system of another provider, refer to the corresponding documentation.

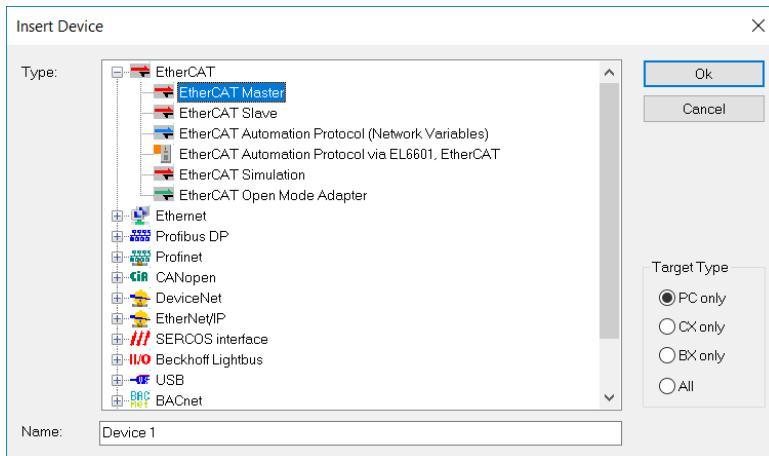
1. Install the ESI file of the device family in TwinCAT®. For TwinCAT®, the ESI file must be copied into the installation folder, e.g.: C:\TwinCAT\3.1\Config\Io\EtherCAT.

2. After installation, TwinCAT® needs a system restart. Alternatively, use the menu bars in TwinCAT® to reload the application: **TWINCAT > EtherCAT Devices > Reload Device Descriptions**.

The devices are now available in the hardware catalog.

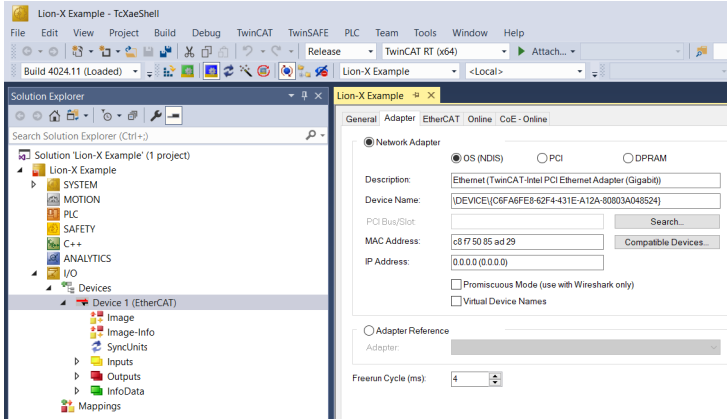
3. Start TwinCAT® and open a new project.

4. Browse to **Solution Explorer > I/O > Devices** in the left workspace window. Right-click on **Devices** and choose the option **Add New Item ... > EtherCAT Master**.



5. If not already done, choose the network adapter and install the driver for EtherCAT® real time communication.

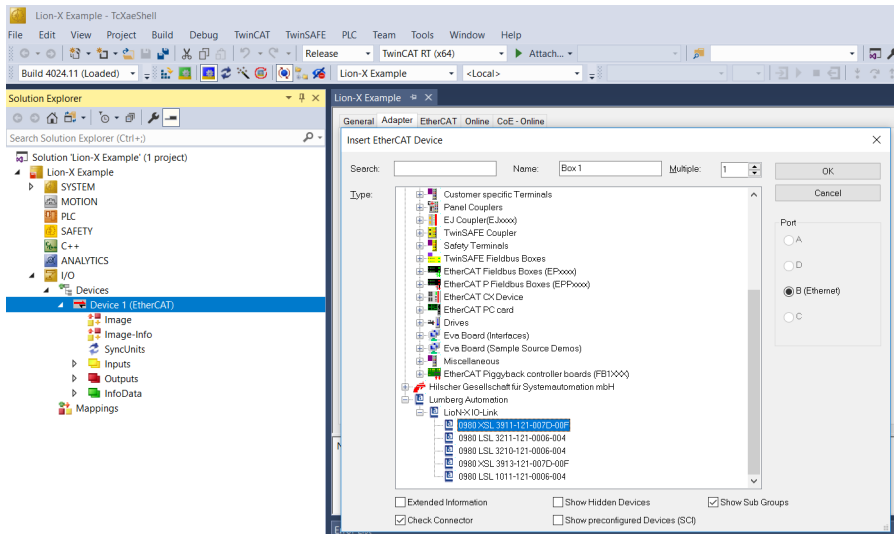
Browse to **Adapter** in the right workspace window and click on **Compatible Devices...** to choose the driver and start the installation.



8.3.1 Configuration of 0980 XSL 391x-1x1 devices

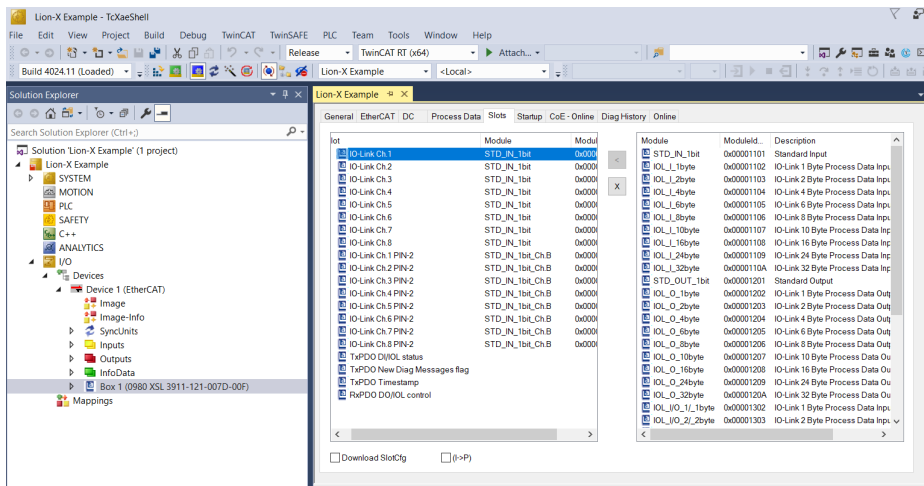
1. Selecting the I/O device from the hardware catalog:

Browse to **Solution Explorer > I/O > Devices** in the left workspace window. Right-click on **Device 1 (EtherCAT)** and choose the option **Add New Item** Select the device and click on **OK**.



2. Configuring the “Slots”:

Browse to **Slots** in the right workspace window and configure the IO-Link channels. You can change, for example, the Input/Output length, the channel mode, or the I/O mode of Channel B (Pin 2). Additional PDOs like "TXPDO" for the DI/IOL status, "TxPDO" for a new diagnostic messages flag, "TxPDO" for a timestamp and "RxPDO" for DO/IOL control can be set as well.



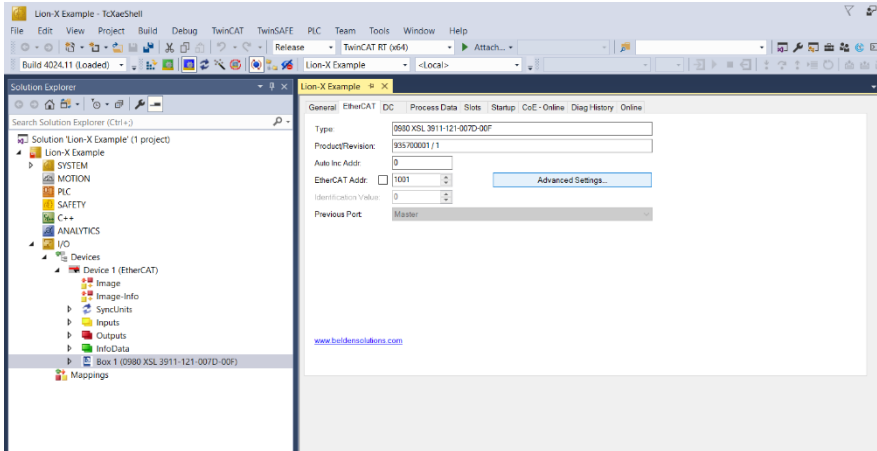
3. Configure the Process Data:

Browse to **Process Data** in the right workspace window and choose the PDOs for Inputs and Outputs.

8.3.2 EoE IP address

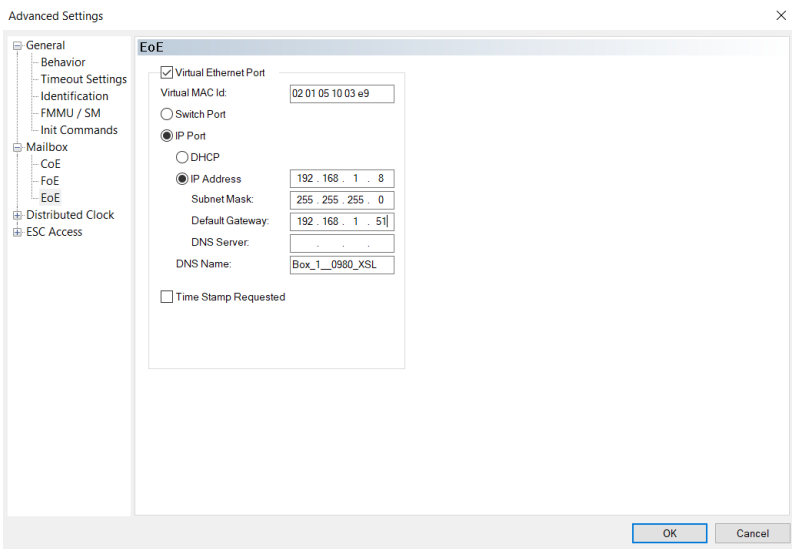
1. Set the IP address for the EoE (Ethernet over EtherCAT®) protocol:

For using the Web interface of the device, the IP address must be set. Click on **EtherCAT > Advanced Settings...** in the right workspace window and navigate to **Mailbox > EoE**.



2. Disable the option **Virtual Ethernet Port** when using no Web services.

3. Activate IP Port and IP Address when using Web services. Enter your IP settings depending from your local network adapter settings.



8.3.3 Activate configuration



Warning: Risk of personal injury or damage of the equipment. Keep away from moveable machine parts during setting up the inputs or outputs of the device.

1.

When the device is connected to the EtherCAT® network, click on **TwinCAT** in the top ribbon and choose **Activate Configuration** in the upcoming window.

2. Click again on **TwinCAT** in the top ribbon and choose **Restart TwinCAT (Config Mode)**. Accept the dialog boxes by clicking on **Yes**. The device will now be changed to “OP” state and will be transferring I/O data.

3. Click on **Write...** to set up any output of the device.

Set Value Dialog

Dec:

Hex:

Float:

Bool:

Binary:

Bit Size: 1 8 16 32 64 ?

9 Diagnostics processing

9.1 Error of the system/sensor power supply

The voltage value for the incoming system/sensor power supply is also monitored globally. If the voltage drops below approx. 18 V, or exceeds approx. 30 V, an error message is generated.

The green U_S indicator is off.

The error message has no effect on the outputs.



Caution: It must definitely be ensured that the supply voltage, measured at the most remote participant is not below 18 V DC from the perspective of the system power supply.

9.2 Error of the auxiliary/actuator power supply

The voltage value for the incoming auxiliary/actuator power supply is also monitored globally. If U_{Aux} diagnostic messaging is enabled, an error message is generated if the voltage drops below approx. 18 V or exceeds approx. 30 V.

The U_{Aux} indicator shows red. If output channels are enabled, additional error messages caused by the voltage failure are generated on the I/O ports. U_{Aux} diagnostic messaging is disabled by default and must be enabled via parameterization.

9.3 Overload/short-circuit of the I/O port sensor supply outputs

In case of an overload or a short circuit between pin 1 and pin 3 on the ports (X1 - X8), the following channel-specific diagnostic messages are generated:

9.4 Overload/short circuit of the digital outputs

A channel error is determined by comparing the target value set by a controller and the actual value of an output channel.

When an output channel is activated (rising edge of the channel state), the channel errors are filtered for the period that you set using the "Surveillance-Timeout" parameter during the configuration of the module. The value of this parameter can range from 0 to 255 ms; the factory setting is 80 ms.

The filter is used to avoid premature error messages when a capacitive load is activated or an inductive load is deactivated, and during other voltage peaks when a status changes.

In static state of the output channel, that is, while the channel is permanently switched on, the filter time between error detection and the diagnostic message is typically 5 ms.

9.5 IO-Link C/Q error

If an IO-Link Device in COM mode is unplugged, an incorrect IO-Link Device is plugged in, or an electrical fault occurs on the C/Q (Pin 4) line, for example, due to a short circuit, an error message is generated.

9.6 Diagnostic history object (0x10F3)

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0x10F3	69	0x10F3: 1	1	UINT8	Maximum Messages
		0x10F3: 2	1	UINT8	Newest Message
		0x10F3: 3	1	UINT8	Newest Acknowledged Message
		0x10F3: 4	1	BOOL	New Messages Available
		0x10F3: 5	1	UINT16	Flags
		0x10F3: 6-69	1	OCTET_STRING	Diagnosis Message Buffer

9.6.1 Maximum Messages

Number of diagnosis messages which can be stored in the diagnosis history (subindex 6 onwards).

9.6.2 Newest Message

Sub-index of the newest diagnosis message (6-69).

9.6.3 Newest Acknowledge Message

Overwrite mode (SI5, bit 4 = 0)

Read = 0:

When the message queue will be overwritten, the slave sets SI3 to 0.

Writing = 0:

The slave clears all messages, i.e. resets SI2, SI3, SI4 and SI5 bit 5. Messages are deleted even if they were not acknowledged or read before.

Writing = 1...5:

The slave returns SDO abort with code 0x06090032 (value of parameter written too low).

Writing = 6...69:

SI3 = written value (without checking)

All messages up to the age of the message which is in the written sub-index are acknowledged. The slave does not check if those messages have been read before.

The slave returns SDO abort with code 0x06090030 (value range of parameter exceeded) in the following case: SI3 is written with a value of a sub-index which does not hold a message.

Writing = 69...255:

The slave returns SDO abort with code 0x06090031 (value of parameter written too high).

Acknowledge mode (SI5, bit 4 = 1)**Read = 0:**

No messages have been acknowledged so far.

Read = 0:

Sub-index of latest acknowledged diagnosis message (6-69)

Writing = 0:

All acknowledged messages will be deleted.

Writing = 1...5:

The slave returns SDO abort with code 0x06090032 (value of parameter written too low).

Writing = 6...69:

Messages are acknowledged

All messages up to the age of the message which is in the written sub-index are acknowledged. The slave does not check if those messages have been read before. The slave returns SDO abort with

code 0x06090030 (value range of parameter exceeded) in the following case: SI3 is written with a value of a sub-index which does not hold a message.

Writing = 69...255:

The slave returns SDO abort with code 0x06090031 (value of parameter written too high).

9.6.4 New Messages Available

Overwrite mode:

0: newest message was read

1: newest message was not read

Acknowledge mode:

0: no unacknowledged message

1: diagnosis messages are available which can be acknowledged (SI2 != SI3).

9.6.5 Flags

Flags to control sending and storing of diagnosis messages:

Bit 0: Enable Emergency sending

0: Default, if device does not support emergency sending.

1: New diagnosis messages shall be sent as emergency message.

Bit 1: Disable info messages

0: Info messages are stored in the diagnosis message queue (default).

1: Info messages will not be stored in the diagnosis message queue.

Bit 2: Disable warning messages

0: Warning messages are stored in the diagnosis message queue (default).

1: Warning messages will not be stored in the diagnosis message queue.

Bit 3: Disable error messages

0: Error messages are stored in the diagnosis message queue (default).

1: Error messages will not be stored in the diagnosis message queue.

Bit 4: Mode selection for diagnosis history handling

0: Overwrite mode. Old messages are overwritten by new ones when buffer is full.

1: Acknowledge mode. New messages do only overwrite messages which were acknowledged before.

Bit 5: Overwrite/Discard Information (read only)

In Overwrite mode:

1: Unacknowledged messages have been overwritten (= buffer overrun) (SI3 is set to 0, too).

In Acknowledge mode:

1: Message buffer is full with acknowledged messages and a new message is discarded.

Bit 6 – 15: reserved

9.6.6 Diagnosis Message Buffer

Depending on SI1, the EtherCAT® slave can store up to 64 messages; the first message is stored in sub-index 6, the second in sub-index 7 and so on. When the queue is full, the EtherCAT® slave shall overwrite sub-index 6 and the following, in a way that always the latest maximum messages (SI1) shall be accessible by the EtherCAT® master.

9.7 Diagnostic data object (0xA0n0)

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0xA0n0	2	0xA0n0: 1	1	UINT8	IO-Link State
		0xA0n0: 2	1	UINT8	Lost Frames

IO-Link State

State of the IO-Link Master (state machine of the IO-Link port)

0x0 = Port inactive

0x01 = Digital input

0x02 = Digital output

0x03 = Establish communication

0x04-0x07: reserved

0x08 = Communication OPERATE

0x09 = Communication STOP

Lost Frames

Counter of lost IO-Link telegrams. Reset counter during start-up.

9.8 Device status object (0xF100)

SDO		SDO content			
Index	Size	Index	Size	Type	Name
0xF100	8	0xF100: 1	1	UINT8	Status Port 1
		0xF100: 2	1	UINT8	Status Port 2
		0xF100: 3	1	UINT8	Status Port 3
		0xF100: 4	1	UINT8	Status Port 4
		0xF100: 5	1	UINT8	Status Port 5
		0xF100: 6	1	UINT8	Status Port 6
		0xF100: 7	1	UINT8	Status Port 7
		0xF100: 8	1	UINT8	Status Port 8

Status Ports 1 - 8

Status of the IO-Link port (state machine of the IO-Link port)

0x0 = Port inactive

0x01 = Digital input

0x02 = Digital output

0x03 = Establish communication

0x04-0x07 = reserved

0x08 = Communication OPERATE

0x09 = Communication STOP

9.9 Emergency messages

If set in the parameters, the Device sends emergency messages to the Master in case of a detected diagnosis on the Device.

Emergency error code	B7	B6	B5	B4	B3	B2	B1	B0	Error description
0x0000	0	0	0	0	0	0	0	0	No error
0x2300	–	0	0	0	0	0	1	1	Sensor short circuit
0x3100	–	0	0	0	0	1	–	1	U _S Voltage error
0x3300	–	0	0	0	0	1	–	0	U _L Voltage error
0xF000	1	0	0	0	0	0	–	1	Additional function forcing
0xFF00	1	0	0	0	0	0	–	1	Additional function parameter error

10 IloT functionality

The Lion-X variants offer a number of new interfaces and functions for the optimal integration into existing or future IloT (Industrial Internet of Things) networks. The devices continue to work as field bus devices which communicate with and are controlled by a PLC (Programmable Logic Controller).

In addition, the devices offer common IloT interfaces, which enable new communication channels besides the PLC. The communication is performed via IloT-relevant protocols MQTT and OPC UA. With the help of these interfaces not only all information in a Lion-X device can be read. They also enable its configuration and control, if the user wishes. All interfaces can be configured extensively and offer read-only functionality.

All Lion-X variants provide user administration, which is also applicable for accessing and configuring the IloT protocols. This allows you to manage all modification options for the device settings via personalized user authorizations.

All IloT protocols can be used and configured independently of the field bus. It is also possible to use the devices completely without the help of a PLC and control them via IloT protocols.



Attention: When using the IloT functionality, a protected local network environment without direct access to the Internet is recommended.

10.1 MQTT

MQTT functions are **only** applicable for the following LioN-X variant:

- ▶ 0980 XSL 3912-121-007D-00F

The MQTT (Message Queuing Telemetry Transport) protocol is an open network protocol for machine-to-machine communication, which provides the transmission of telemetric data messages between devices. The integrated MQTT client allows the device to publish a specific set of information to an MQTT broker.

The publishing of messages can either occur periodically or be triggered manually.

10.1.1 MQTT configuration

In **delivery state**, MQTT functions are **disabled**. The MQTT client can be configured either using the Web interface or directly via a JSON object sent in an HTTP request. For more information see chapter [MQTT configuration - Quick start guide](#) on page 103.

The configuration URL is:

```
http://[ip-address]/w/config/mqtt.json
```

The configuration can also read back as a JSON file:

```
http://[ip-address]/r/config/mqtt.json
```

The configuration is a JSON object. Each JSON member is a configuration element. The object must not contain all elements. Only the provided elements will be changed. The configuration changes apply only after a device restart.

The following configuration elements are available (default values in bold):

Element	Data type	Description	Example data
mqtt-enable	boolean	Master switch for the MQTT client.	true / false
broker	string	IP address of the MQTT Broker	" 192.168.1.1 "
login	string	Username for MQTT Broker	"admin" (Default: null)
password	string	Password for MQTT Broker	"private" (Default: null)
port	number	Broker port	1883
base-topic	string	Base topic	"iomodule_[mac]" (Default: " lionx ")
will-enable	boolean	If true, the device provides a last will message to the broker	true / false
will-topic	string	The topic for the last will message.	(Default: null)
auto-publish	boolean	If true, all enabled domains will be published automatically in the specified interval.	true / false
publish-interval	number	The publish interval in ms if auto-publish is enabled. Minimum is 250 ms.	2000
publish-identity	boolean	If true, all identity domain data will be published	true / false
publish-config	boolean	If true, all config domain data will be published	true / false
publish-status	boolean	If true, all status domain data will be published	true / false
publish-process	boolean	If true, all process domain data will be published	true / false
publish-devices	boolean	If true, all IO-Link Device domain data will be published	true / false
commands-allowed	boolean	Master switch for MQTT commands. If false, the device will not subscribe to any command topic, even if specific command topics are activated below.	true / false
force-allowed	boolean	If true, the device accepts force commands via MQTT.	true / false
reset-allowed	boolean	If true, the device accepts restart and factory reset commands via MQTT.	true / false
config-allowed	boolean	If true, the device accepts configuration changes via MQTT.	true / false

Element	Data type	Description	Example data
qos	number	Selects the "Quality of Service" status for all published messages.	0 = At most once 1 = At least once 2 = Exactly once

Table 11: MQTT configuration

MQTT response:

The resulting response is a JSON object with a "status" field. Status should be "0" if no error occurred and "-1" if there is an error.

In case of an error, the response contains an error array.

The error array contains an error object for each error occurred. The object consists of a field "Element" which names the config element which caused the error, and a field "Message" for the error message.

- ▶ A malformed JSON object produces an error.
- ▶ Not existing parameters produce an error.
- ▶ Parameters with a wrong data type produce an error.

It is not allowed to write all available parameters at once. You may write only one or a limited number of parameters.

Examples:

```
{ "status": -1, "error": [{"Element": "publish-interval", "Message": "Integer
expected"}]}

{ "status": 0}

{ "status": -1, "error": [{"Element": "root", "Message": "Not a JSON
object"}]}
```

For more information see chapter [MQTT topics](#) on page 95.

10.1.2 MQTT topics

MQTT mainly relates to topics. All messages are attached to a topic which adds context to the message itself. Topics may consist of any string and they are allowed to contain slashes (/) as well as wildcard symbols (*, #).

10.1.2.1 Base topic

For LioN-X and the LioN-Xlight variants there is a configurable *Base topic* which is the prefix for all topics. The *Base topic* can be chosen freely by the user. The *Base topic* can also contain selected variables as shown in [Table 12: Base topic variables](#) on page 95.

Variables in the *Base topic* have to be written in brackets ("[]"). The following variables are possible:

Variable	Description
mac	The MAC address of the device
name	The name of the device
order	The ordering number of the device
serial	The serial number of the device

Table 12: Base topic variables

Example:

The *Base topic* "io_[mac]" translates to "io_A3B6F3F0F2F1".

All data is organized in domains. The domain name is the first level in the topic after the *Base topic*. Note the following notation:

Base-Topic/domain/....

There are the following domains:

Domain name	Definition	Example content
identity	All fixed data which is defined by the used hardware and which cannot be changed by configuration or at runtime.	Device name, ordering number, MAC address, port types, port capabilities and more.
config	Configuration data which is commonly loaded once at startup, mostly by a PLC.	IP address, port modes, input logic, failsafe values and more.
status	All (non-process) data which changes quite often in normal operation.	Bus state, diagnostic information, IO-Link device status and data.
process	All process data which is produced and consumed by the device itself or by attached devices.	Digital inputs, digital outputs, cyclic IO-Link data.
iold	IO-Link device parameters according to the IO-Link specification.	Vendor name, product name, serial number, hardware revision, software revision and more.

Table 13: Data domains

There is often one topic used for all gateway related information and topics for each port. All identity topics are published just once at start-up, because this information should never change. All other topics are published either in a fixed interval or just triggered manually, according to the configuration.

Topic	Content examples	Total publish count	Publish interval
[base-topic]/identity/gateway	Name, ordering number, MAC, vendor, I&M etc.	1	Startup
[base-topic]/identity/port/n	Port name, port type	8	Startup
[base-topic]/config/gateway	Configuration parameters, ip address etc.	1	Interval
[base-topic]/config/port/n	Port mode, data storage, mapping, direction	8	Interval
[base-topic]/status/gateway	Bus state, device diagnosis, master events	1	Interval
[base-topic]/status/port/n	Port or channel diagnosis, IO-Link state, IO-Link device events	8	Interval
[base-topic]/process/gateway	All Digital IN/OUT	1	Interval
[base-topic]/process/port/n	Digital IN/OUT per port, IOL-data, pdValid	8	Interval
[base-topic]/iold/port/n	IO-Link device parameter	8	Interval

Table 14: Data model

An MQTT client which wants to subscribe to one or more of these topics can also use wildcards.

Full topic	Description
[base-topic]/identity/gateway	Receive only identity objects for the gateway
[base-topic]/identity/#	Receive all data related to the identity domain
[base-topic]/status/port/5	Receive only status information for port number 5
[base-topic]/+/port/2	Receive information of all domains for port number 2
[base-topic]/process/port/#	Receive only process data for all ports
[base-topic]/config/#	Receive config data for the gateway and all ports.

Table 15: Topic use case examples

10.1.2.2 Publish topic

Overview of all publish JSON data for the defined topics:

Key	Data type
tbd	json_string
ordering_number	json_string
device_type	json_string
serial_number	json_string
mac_address	json_string
production_date	json_string
fw_name	json_string
fw_date	json_string
fw_version	json_string
hw_version	json_string
vendor_name	json_string
vendor_address	json_string
vendor_phone	json_string
vendor_email	json_string
vendor_techn_support	json_string
vendor_url	json_string
vendor_id	json_integer
device_id	json_integer

Table 16: Identity/gateway

Key	Data type	Range	Default value	Remarks
fieldbus_protocol	json_string	profinet, ethernet/ip, ethercat		
network_configuration	json_string	PNS: dcp EIS: stored_value, bootp, dhcp		
rotary_switches	json_integer	0..999		
ip_address	json_string		192.168.1.1	
subnet_mask	json_string		255.255.255.0	
report_alarms	json_boolean		0.0.0.0	
report_ul_alarm	json_boolean	true / false	true	
report_do_fault_without_ul	json_boolean	true / false	false	
force_mode_lock	json_boolean	true / false	false	
web_interface_lock	json_boolean	true / false	false	
do_auto_restart	json_boolean	true / false	true	
fast_startup	json_boolean	true / false	false	PROFINET and EIP only

Table 17: Config/gateway

Key	Data type	Range	Default value	Remarks
protocol	json_string	wait_for_io_system wait_for_io_Connection failsafe connected error		
ethernet_port1	json_string	100_mbit/s_full 100_mbit/s 10_mbit/s_full 100_mbit/s		
ethernet_port2	json_string	100_mbit/s_full 100_mbit/s 10_mbit/s_full 100_mbit/s		
module_restarts	json_integer	0..4294967295		
channel_diagnosis	json_boolean	true / false		
failsafe_active	json_boolean	true / false		
system_voltage_fault	json_boolean	true / false		
actuator_voltage_fault	json_boolean	true / false		
internal_module_error	json_boolean	true / false		
forcemode_enabled	json_boolean	true / false		

Table 18: Status/gateway

Key	Data type	Range	Default value	Remarks
Input_data	json_integer[]			
output_data	json_integer[]			

Table 19: Process/gateway

Key	Data type	Range	Default value	Remarks
port	json_integer	1..8		
type	json_string	digital_universal digital_input digital_Output io_link		
max_output_power_cha	json_string	2.0_mA 0.5_mA		
max_output_power_chb	json_string	2.0_mA 0.5_mA		
channel_cha	json_string	input/output input output io_link aux		
channel_chb	json_string	input/output input output io_link aux		

Table 20: Identity/port/1 ... 8

Key	Data type	Range	Default value	Remarks
port	json_integer	1..8		
direction_cha	json_string	input/output input output		
direction_chb	json_string	input/output input output		
failsafe_cha	json_string	set_low set_high hold_last	set_low	
failsafe_chb	json_string	set_low set_high hold_last	set_low	
surveillance_timeout_cha	json_integer	0..255	80	
surveillance_timeout_chb	json_integer	0..255	80	

Table 21: Config/port/1 ... 8

Key	Data type	Range	Default value	Remarks
port	json_integer	1..8		
physical_state_cha	json_integer	0..1		
physical_state_chb	json_integer	0..1		
actuator_short_circuit_cha	json_boolean	true / false		
actuator_short_circuit_chb	json_boolean	true / false		
sensor_short_circuit	json_boolean	true / false		

Table 22: Status/port/1 ... 8

10.1.3 MQTT configuration - Quick start guide



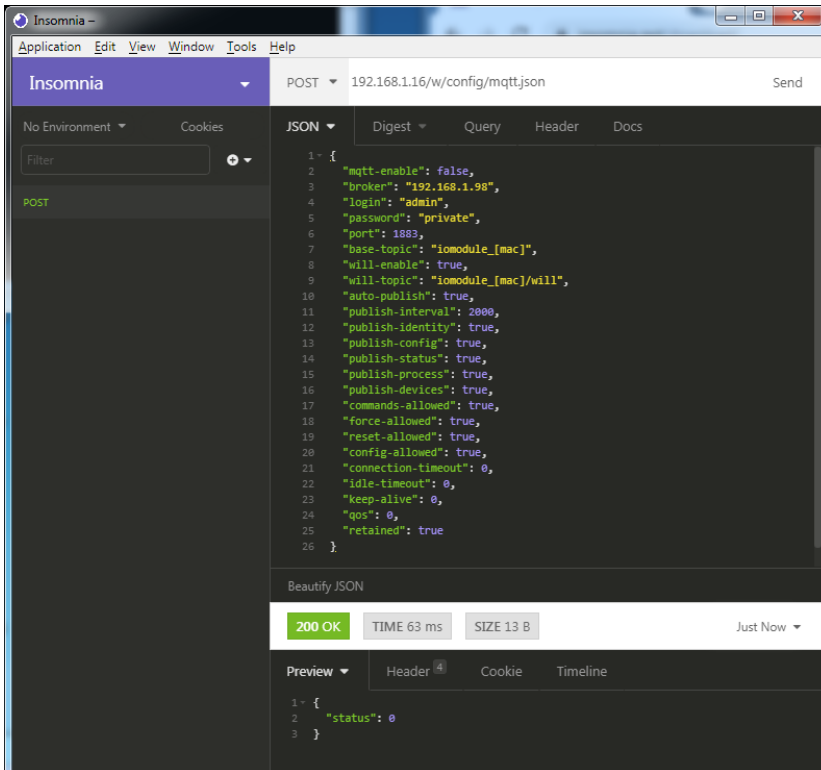
Attention: Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

10.1.3.1 MQTT configuration via JSON

1. Depending on your application case, download and install *Insomnia* or a comparable application: <https://insomnia.rest/download/>

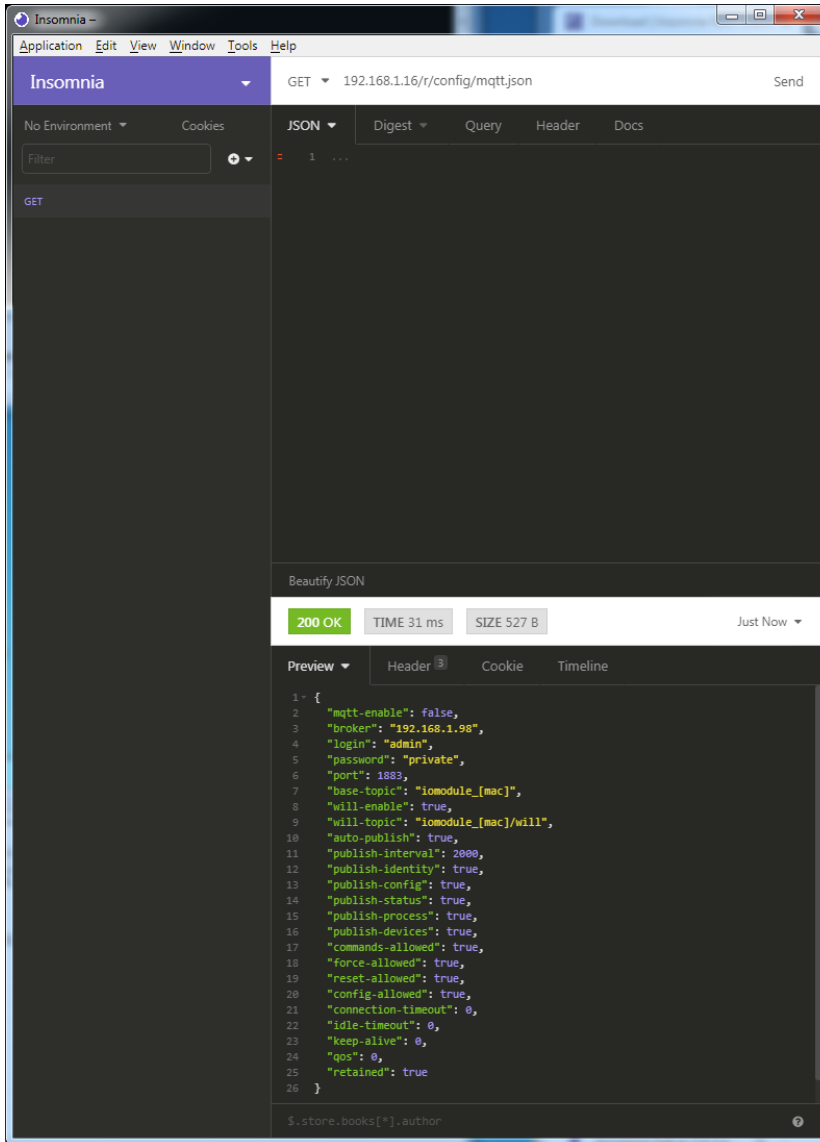
2. Configure MQTT:

POST: [IP-address]/w/config/mqtt.json



3. Read MQTT:

GET: [IP-address]/r/config/mqtt.json



10.2 OPC UA

OPC UA functions are **only** applicable for the following LioN-X variant:

- ▶ 0980 XSL 3912-121-007D-00F

OPC Unified Architecture (OPC UA) is a platform-independent standard with a service-oriented architecture for communication in and with industrial automation systems.

The OPC UA standard is based on the client-server principle and lets machines and devices, regardless of any preferred field bus, communicate horizontally among each other as well as vertically to the ERP system or the cloud. LioN-X provides an OPC UA server on field device level, with which an OPC UA client can connect for information exchange secure in transmission.

For OPC UA, we comply (apart from the exceptions listed [below](#)) with the IO-Link Companion Specification, which can be downloaded from catalog.belden.com or directly from io-link.com.

Feature	Support
Managing IODDs (chapter 6.1.6 in the specification)	Not supported
Mapping IODD information to OPC UA ObjectTypes (chapter 6.3 in the specification)	Not supported
IOLinkIODDDeviceType (chapters 7.2 ff. in the specification)	Not supported
ObjectTypes generated based on IODDs (chapters 7.3 ff. in the specification)	Not supported
Creation of Instances based on ObjectTypes generated out of IODDs (chapter 7.4 in the specification)	Not supported
IODDManagement Object (chapter 8.2 in the specification)	Not supported
RemoveIODD Method (chapter 8.3 in the specification)	Not supported

Table 23: Non-supported OPC UA features according to the IO-Link Companion Specification

10.2.1 OPC UA configuration

In **delivery state**, OPC UA functions are **disabled**. The OPC UA Server can be configured either using the Web interface or directly via a JSON Object sent in an HTTP request. For more information see [OPC UA configuration - Quick start guide](#) on page 109.

The configuration URL is:

```
http://[ip-address]/w/config/opcu.json
```

The configuration can also read back as a JSON file:

```
http://[ip-address]/r/config/opcu.json
```

The configuration is a JSON object. Each JSON member is a configuration element. The object must not contain all elements. Only the provided elements will be changed. All configuration changes apply only after a device restart.

There are the following configuration elements (default values in bold):

Element	Data type	Description	Example data
port	integer	Server port for the OPC UA server.	0, 4840 , 0xFFFF
opcua-enable	boolean	Master switch for the OPC UA server.	true / false
anon-allowed	boolean	If true, anonymous login is allowed.	true / false
commands-allowed	boolean	Master switch for OPC UA commands. If false there will be no writeable OPC UA objects.	true / false
force-allowed	boolean	If true, the device accepts force commands via OPC UA.	true / false
reset-allowed	boolean	If true, the device accepts restart and factory reset commands via OPC UA.	true / false
config-allowed	boolean	If true, the device accepts configuration changes via OPC UA.	true / false

Table 24: OPC UA Configuration

All configuration elements are optional and do not need a specific order. Not every element is required to be sent. This means that only configuration changes will be taken over.

Optional: The configuration parameters of OPC UA can be set directly via the Web interface. It is possible to download the Web interface for sharing with other devices.

Response:

The resulting response is a JSON object with a "status" field. Status should be 0 if no error occurred and -1 if there is an error.

In case of an error, the response contains an error array.

The error array contains an error object for each error occurred. The object consists of a field "Element" which names the config element which caused the error, and a field "Message" for the error message.

Examples:

```
{"status": -1, "error": [{"Element": "upcua-enable", "Message": "Boolean expected"}]}  
  
{"status": 0}  
  
{"status": -1, "error": [{"Element": "root", "Message": "Not a JSON object"}]}
```

10.2.2 OPC UA address space

OPC UA provides different services on the Lion-X devices with which a client can navigate through the hierarchy of the address space and read or write variables. In addition, the client can monitor up to 10 attributes from the address space for value changes.

A connection to an OPC UA server is established via the endpoint URL:

```
opc.tcp://[ip-address]:[port]
```

Various device data such as MAC address, device settings, diagnostics or status information can be read via *Identity objects*, *Config objects*, *Status objects* and *Process objects*.

Command objects can be read and written. This makes it possible, for example, to transfer new network parameters to the device, to use Force Mode or to reset the entire device to its factory settings.

The following figures illustrate the OPC UA address space of the Lion-X devices. The objects and information displayed depend on the device variant used.

10.2.3 OPC UA configuration - Quick start guide

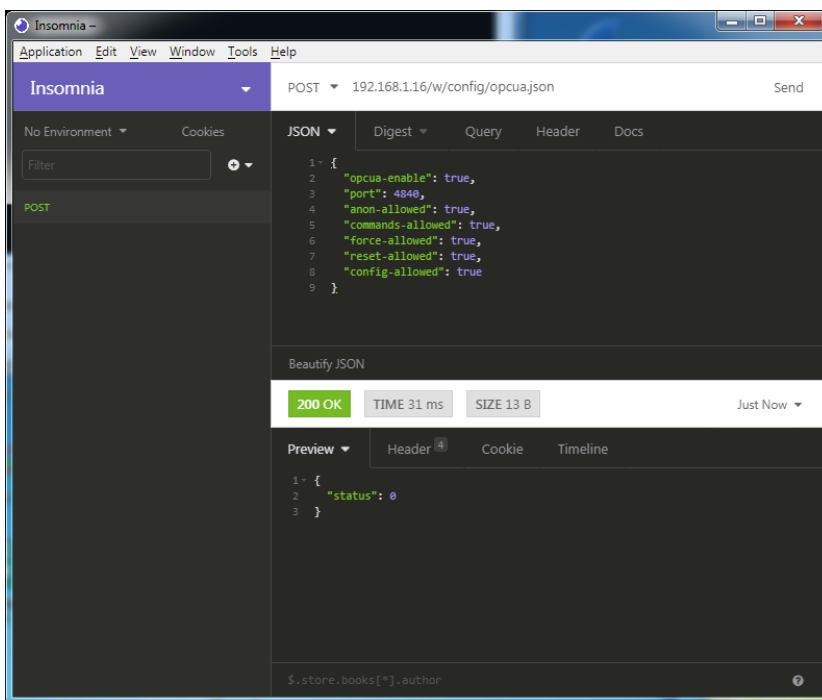
i Attention: Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

10.2.3.1 OPC UA configuration via JSON

1. Depending on your application case, download and install *Insomnia* or a comparable application: <https://insomnia.rest/download/>

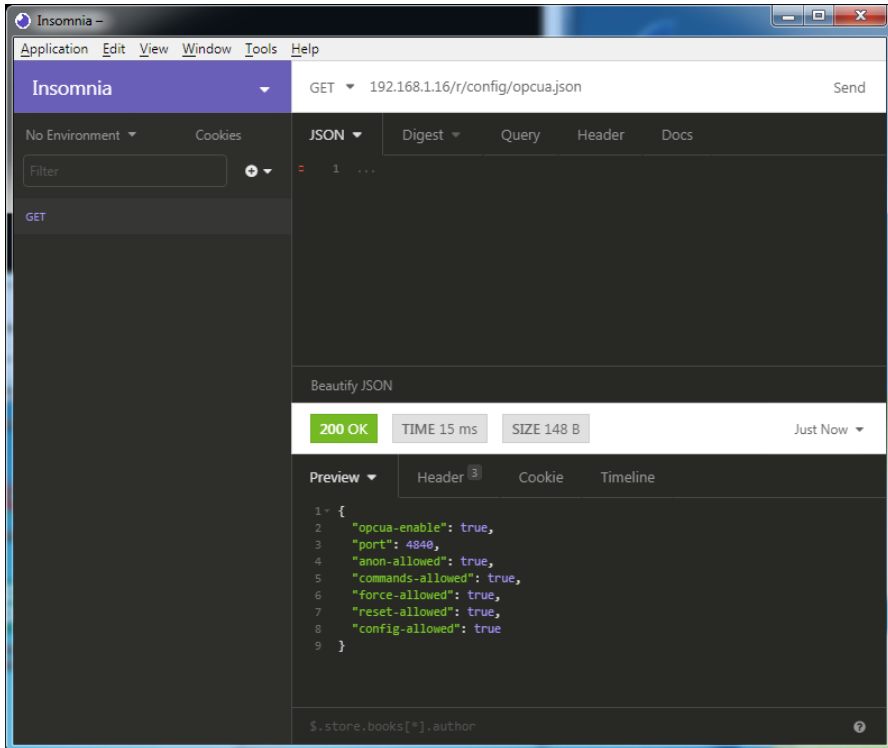
2. Configure OPC UA:

POST: [IP-address] /w/config/opcuajson



3. Read OPC UA:

GET: [IP-address]/r/config/opcuajson



10.3 REST API

The Representational State Transfer – Application Programming Interface (REST API) is a programmable interface which uses HTTP requests to GET and POST data. This enables the access to detailed device information.

For LioN-X and the LioN-Xlight variants, the REST API can be used to read the device status. For the LioN-X multiprotocol variants, the REST API can also be used to write configuration and forcing data.

There are two different REST API standards you can use for the requests:

1. A standardized REST API that has been specified by the IO-Link Community and is described separately:

JSON_Integration_10222_V100_Mar20.pdf

Please download the file from catalog.belden.com or directly from io-link.com.



Attention: Consider the following table to get an overview of the supported features of the IO-Link specification:

Feature		Supported
Gateway	GET /identification	Yes
	GET /capabilities	Yes
	GET /configuration	Yes
	POST /configuration	Yes
	POST /reset	Yes
	POST /reboot	Yes
	GET /events	Yes
Master	GET /masters	Yes
	GET /capabilities	Yes
	GET /identification	Yes
	POST /identification	Yes

Feature		Supported
Port	GET /ports	Yes
	GET /capabilities	Yes
	GET /status	Yes
	GET /configuration	Yes
	POST /configuration	Yes
	GET /datastorage	Not supported
	POST /datastorage	Not supported
Devices	GET /devices	Yes
	GET /capabilities	Yes
	GET /identification	Yes
	POST /identification	Yes
	GET /processdata/value	Yes
	GET /processdata/getdata/value	Yes
	GET /processdata/setdata/value	Yes
	POST /processdata/value	Yes
	GET /parameters	Yes
	GET /parameters/{index}/subindices	Yes
	GET /parameters/{parameterName}/subindices	Not supported
	GET /parameters/{index}/value	Yes
	GET /parameters/{index}/subindices/{subindex}/value	Yes
	GET /parameters/{parameterName}/value	Not supported
	GET /parameters/{parameterName}/subindices/{subParameterName}/value	Not supported
	POST /parameters/{index}/value	Yes
	POST /parameters/{parameterName}/value	Not supported
	POST /parameters/{index}/subindices/{subindex}/value	Yes
	POST /parameters/{parameterName}/subindices/{subParameterName}/value	Not supported
	POST /blockparametrization	Not supported
	GET /events	Yes

Feature		Supported
IODD	GET /iodds	Not supported
	POST /iodds/file	Not supported
	DELETE /iodds	Not supported
	GET /iodds/file	Not supported

Table 25: Support of REST API features according to the IO-Link specification

2. A customized Belden REST API that is described in the following chapters.

10.3.1 Standard device information

Request method:	http GET
Request URL:	<ip>/info.json
Parameters	n.a.
Response format	JSON

The goal of the "Standard device information" request is to get a complete snapshot of the current device status. The format is JSON. For IO-Link devices, all ports with connected IO-Link device information are included.

10.3.2 Structure

Name	Data type	Description	Example
name	string	Device name	"0980 XSL 3912-121-007D-00F"
order-id	string	Ordering number	"935 700 001"
fw-version	string	Firmware version	"V.1.1.0.0 - 01.01.2021"
hw-version	string	Hardware version	"V.1.00"
mac	string	MAC address of the device	"3C B9 A6 F3 F6 05"
bus	number	0 = No connection 1 = Connection with PLC	1
failsafe	number	0 = Normal operation 1 = Outputs are in failsafe	0
ip	string	IP address of the device	
snMask	string	Subnet Mask	
gw	string	Default gateway	
rotarys	array of numbers (3)	Current position of the rotary switches: Array element 0 = x1 Array element 1 = x10 Array element 2 = x100	
ulPresent	boolean	True, if there is a UL voltage supply detected within valid range	
usVoltage_mv	number	US voltage supply in mV	
ulVoltage_mv	number	UL voltage supply in mV (only available for devices with UL supply)	
inputs	array of numbers (2)	Real state of digital inputs. Element 0 = 1 Byte: Port X1 Channel A to Port X4 Channel B Element 0 = 1 Byte: Port X5 Channel A to Port X8 Channel B	\[128,3]
output	array of numbers (2)	Real State of digital outputs. Element 0 =1 Byte: Port X1 Channel A to port X4 Channel B Element 0 = 1 Byte: Port X5 Channel A to port X8 Channel B	\[55,8]

Name	Data type	Description	Example																		
consuming	array of numbers (2)	Cyclic data from PLC to device																			
producing	array of numbers (2)	Cyclic data from device to PLC																			
diag	array of numbers (4)	Diagnostic information <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">Element 0 = 1 Byte:</td> <td>Internal module error (IME)</td> </tr> <tr> <td>Force mode:</td> <td>active</td> </tr> <tr> <td>Actuator:</td> <td>short</td> </tr> <tr> <td>Sensor:</td> <td>short</td> </tr> <tr> <td>U_L:</td> <td>fault</td> </tr> <tr> <td>U_S:</td> <td>fault</td> </tr> <tr> <td>Element 1 = 1 Byte:</td> <td>Sensor short circuit ports X1-X8.</td> </tr> <tr> <td>Element 2 = 1 Byte:</td> <td>Actuator short circuit ports X1 Channel A-X4 Channel B</td> </tr> <tr> <td>Element 3 = 1 Byte:</td> <td>Actuator short circuit ports X5 Channel A-X8 Channel B</td> </tr> </table>	Element 0 = 1 Byte:	Internal module error (IME)	Force mode:	active	Actuator:	short	Sensor:	short	U_L:	fault	U_S:	fault	Element 1 = 1 Byte:	Sensor short circuit ports X1-X8.	Element 2 = 1 Byte:	Actuator short circuit ports X1 Channel A-X4 Channel B	Element 3 = 1 Byte:	Actuator short circuit ports X5 Channel A-X8 Channel B	
Element 0 = 1 Byte:	Internal module error (IME)																				
Force mode:	active																				
Actuator:	short																				
Sensor:	short																				
U_L:	fault																				
U_S:	fault																				
Element 1 = 1 Byte:	Sensor short circuit ports X1-X8.																				
Element 2 = 1 Byte:	Actuator short circuit ports X1 Channel A-X4 Channel B																				
Element 3 = 1 Byte:	Actuator short circuit ports X5 Channel A-X8 Channel B																				
fieldbus	FIELDBUS Object																				
FIELDBUS Object																					
fieldbus_name	string	Currently used fieldbus																			
state	number	Fieldbus state																			
state_text	number	Textual representation of fieldbus state: 0 = Unknown 1 = Bus disconnected 2 = Preop 3 = Connected 4 = Error 5 = Stateless																			
forcing	FORCING Object	Information about the forcing state of the device																			

Name	Data type	Description	Example
channels	Array of CHANNEL (16)	Basic information about all input/output channels	
iol	IOL Object	Contains all IO-Link related information such as events, port states, device parameters.	
iol/diagGateway	array of DIAG	Array of currently active device/gateway related events	
iol/diagMaster	array of DIAG	Array of currently active IOL-Master related events	
iol/ports	array of PORT (8)	Contains one element for each IO-Link port	
CHANNEL Object			
name	string	Name of channel	
type	number	Hardware channel type as number: 0 = DIO 1 = Input 2 = Output 3 = Input/Output 4 = IO-Link 5 = IOL AUX 6 = IOL AUX with DO 7 = IOL AUX with DO. Can be deactivated. 8 = Channel not available	
type_text	string	Textual representation of the channel type	
config	number	Current configuration of the channel: 0 = DIO 1 = Input 2 = Output 3 = IO-Link 4 = Deactivated 5 = IOL AUX	
config_text	string	Textual representation of the current config	
inputState	boolean	Input data (producing data) bit to the PLC	

Name	Data type	Description	Example
outputState	boolean	Output data bit to the physical output pin	
forced	boolean	True, if the output pin of this channel is forced	
simulated	boolean	True, if the input value to the PLC of this channel is simulated	
actuatorDiag	boolean	True, if the output is in short circuit / overload condition	
sensorDiag	boolean	True, if the sensor supply (Pin 1) is in short circuit / overload condition	
maxOutputCurrent_mA	number	Maximum output current of the output in mA	
current_mA	number	Measured current of the output in mA (if current measurement is available)	
voltage_mV	number	Measured voltage of this output in mV (if voltage measurement is available)	
PORT Object			
port_type	string	Textual representation of the IO-Link port type	
iolink_mode	number	Current port mode: 0 = Inactive 1 = Digital output 2 = Digital input 3 = SIO 4 = IO-Link	
iolink_text	string	Textual representation of the current port mode	"Digital Input"
aux_mode	number	Indicates the configured mode for the Pin 2: 0 = No AUX 1 = AUX output (always on) 2 = Digital output (can be controlled by cyclic data) 3 = Digital input	
aux_text	string	Textual representation of the current aux mode	"AUX Output"
cq_mode	number	Port mode according to IOL specification	

Name	Data type	Description	Example
iq_mode	number	Pin2 mode according to IOL specification	
port_status	number	Port status according to IOL specification	
ds_fault	number	Data storage error number	
ds_fault_text	string	Textual data storage error.	
device	DEVICE Object	IO-Link device parameters. → Null if no IO-Link communication active	
diag	array of DIAG (n)	Array of port related events	
DIAG Object			
error	number	Error code	
source	string	Source of the current error.	"device" "master"
eventcode	number	Event code according to IO-Link specification	
eventqualifier	number	Event qualifier according to IO-Link specification	
message	string	Error message	"Supply Voltage fault"
DEVICE Object		Standard parameters of the IOL-Device	
device_id	number		
vendor_id	number		
serial	string		
baudrate	string	Baudrate (COM1,2,3)	
cycle_time	number	Cycle time in microseconds	
input_len	array of numbers (n)	IOL input length in bytes	
output_len	array of numbers (n)	IOL output length in bytes	
input_data	array of numbers (n)	IOL input data	
output_data	array of numbers (n)	IOL output data	
pd_valid	number	"1", if IOL input data is valid	
pdout_valid	number	"1", if IOL output data is valid	
FORCING Object		Forcing information of the device	

Name	Data type	Description	Example
forcingActive	boolean	Force mode is currently active	
forcingPossible	boolean	True, if forcing is possible and force mode can be activated	
ownForcing	boolean	True, if forcing is performed by REST API at the moment	
forcingClient	string	Current forcing client identifier	
digitalOutForced	array of numbers (2)	The force values of all 16 digital output channels.	
digitalOutMask	array of numbers (2)	The forcing mask of all 16 digital output channels.	
digitalInForced	array of numbers (2)	The force values of all 16 digital input channels.	
digitalInMask	array of numbers (2)	The forcing mask of all 16 digital input channels.	

10.3.3 Configuration and forcing

Method:	POST
URL:	<ip>/w/force.json
Parameters:	None
Post-Body:	JSON Object

Property	Data type	Example values	Description
forcemode	boolean	true / false	Forcing authority on/off
portmode	array (Port mode object)		
digital	array (Digital object)		
iol	array (IOL object)		

Table 26: Root object

Property	Data type	Example values	Remarks
port	integer	0..7	
channel	integer	"a","b"	optional default is "a"
direction	string	"dio","di","do","iol", "off", "aux"	
aux	string	"dio","di","do","iol", "off", "aux"	IOL only, but optional
inlogica	string	"no","nc"	
inlogicb	string	"no","nc"	

Table 27: Port mode object

Property	Data type	Example values	Remarks
port	integer	0..7	
channel	string	"a","b"	
force_dir	string	"phys_out","plc_in","clear"	optional default is "phys_out"
force_value	integer	0,1	

Table 28: Digital object

Property	Data type	Example values	Remarks
port	integer	0..7	
output	array[integer] or null to clear forcing	[55,88,120]	Output forcing
input	array[integer] or null to clear forcing	[20,0,88]	Input simulation to PLC

Table 29: IOL object

10.3.4 Reading and writing ISDU parameters

The *Indexed Service Data Unit* (ISDU) provides a highly flexible message format, which can contain single or multiple commands.

LioN-X IOL-Masters with IloT support reading and writing ISDU parameters from connected IOL-Devices. It is possible to do this as a bulk transfer by reading and writing of multiple ISDU parameters via a single request.

10.3.4.1 Reading ISDU

Method:	POST
URL:	<ip>/r/isdu.json
Parameters:	port (0-7)
Example:	<code>192.168.1.20/r/isdu.json?port=5</code>
Post-Body:	JSON array of read ISDU object

Property	Data type	Example values	Remarks
ix	integer	0-INT16	Index to be read
subix	integer	0-INT8	Subindex to be read

Table 30: Read ISDU object

Property	Data type	Example values	Remarks
status	integer	0, -1	0 = no error, -1= an error occurred
message	string		Error Message if error occurred
data	array (Read ISDU data object)		data, if no error occurred. otherwise null

Table 31: Read ISDU response object

Property	Data type	Example values	Remarks
ix	integer	0-INT16	Index that was read
subix	integer	0-INT8	Subindex that was read
status	integer	0, -1	0 = no error, -1= an error occurred
eventcode	integer		IOL eventcode if status is -1
data	array[integer]		data, if no error occurred. otherwise null

Table 32: Read ISDU data object

10.3.4.2 Writing ISDU

Method:	POST
URL:	<ip>/w/isdu.json
Parameters:	port (0-7)
Post-Body:	JSON array of write ISDU object

Property	Data type	Example values	Remarks
ix	integer	0-INT16	Index to be read
subix	integer	0-INT8	Subindex to be read
data	array[integer]		Data to be written

Table 33: Write ISDU object

Response: Write ISDU response object

Property	Data type	Example values	Remarks
status	integer	0, -1	0 = no error, -1= an error occurred
message	string		Error Message if error occurred
data	array (Write ISDU data object)		data, if no error occurred. otherwise null

Table 34: Write ISDU response object

Property	Data type	Example values	Remarks
ix	integer	0-INT16	Index that was written
subix	integer	0-INT8	Subindex that was written
status	integer	0, -1	0 = no error, -1= an error occurred
eventcode	integer		IOL eventcode if status is -1

Table 35: Write ISDU data object

10.3.5 Example: Reading ISDU

ISDU read request

```
[
  { "ix":5, "subix":0 },
  { "ix":18, "subix":0 },
  { "ix":19, "subix":0 },
  { "ix":20, "subix":0 }
]
```

Response

```
{
  "message": "OK",
  "data":
  [
    { "ix":5, "subix":0, "status":-1, "eventcode":32785 },
    { "ix":18, "subix":0, "data": [79,68,83,49,48,76,49,46,56,47,76,65,54,44,50,
      48,48,45,77,49,50], "status":0 },
    { "ix":19, "subix":0, "data": [53,48,49,50,57,53,51,53], "status":0 },
    { "ix":20, "subix":0, "data": [100,105,115,116,97,110,99,101,32,115,101,110,
      115,111,114], "status":0 }
  ],
  "status":0}
}
```

10.3.6 Example: Writing ISDU

ISDU write request

```
[
  { "ix":24, "subix":0, "data": [97,98,99,100,101,102] },
  { "ix":9, "subix":0, "data": [97,97,97,97,97,98] }
]
```

Response

```
{
  "message": "OK",
  "data": [
    { "ix":24, "subix":0, "status":0 },
    { "ix":9, "subix":0, "eventcode":32785, "status":-1 }
  ],
  "status":0}
}
```

10.4 CoAP server

The CoAP server functions are **only** applicable for the following LioN-X variant:

- ▶ 0980 XSL 3912-121-007D-00F

The **Constrained Application Protocol (CoAP)** is a specialized Internet application protocol for constrained networks such as lossy or low power networks. CoAP is useful especially in M2M (Machine to Machine) communication and can be used to translate simplified HTTP requests of low speed networks.

CoAP is based on the Server-Client principle and a service layer protocol that lets nodes and machines communicate with each other. The LioN-X multiprotocol variants provide CoAP server functionalities via a REST API interface over UDP.

10.4.1 CoAP configuration

In **delivery state**, CoAP functions are **disabled**. The CoAP server can be configured either using the Web interface or directly via a JSON object sent in an HTTP request. For more information see chapter [CoAP configuration - Quick start guide](#) on page 131.

The configuration URL is:

```
http://[ip-address]/w/config/coapd.json
```

The configuration can also read back as a JSON file:

```
http://[ip-address]/r/config/coapd.json
```

The configuration is a JSON object. Each JSON member is a configuration element. The object must not contain all elements. Only the provided elements will be changed. The configuration changes apply only after a device restart.

The following configuration elements are available (default values in bold):

Element	Data type	Description	Example data
enable	boolean	Master switch for the CoAP server	true / false
port	integer (0 to 65535)	Port of the CoAP server	5683

Table 36: CoAP configuration

CoAP response:

The resulting response is a JSON object with a "status" field. Status should be "0" if no error occurred, and "-1" if there is an error.

In case of an error, the response contains an error array.

The error array contains an error object for each error occurred. The object consists of a field "Element" which names the config element that caused the error, and of a field "Message" for the error message.

Examples:

```
{ "status": -1, "error": [{"Element": "upcua-enable", "Message": "Boolean
expected"}]}

{ "status": 0 }

{ "status": -1, "error": [{"Element": "root", "Message": "Not a JSON
object"}]}
```

10.4.2 REST API access via CoAP

A connection to the CoAP server running on the LioN-X multiprotocol variants can be established via the following URL:

```
coap://[ip-address]:[port]/[api]
```

For LioN-X, the following REST API Requests (JSON format) can be accessed via a CoAP endpoint:

Type	API	Note
GET	/r/status.lr	
GET	/r/system.lr	
GET	/info.json"	
GET	/r/config/net.json	
GET	/r/config/mqtt.json	
GET	/r/config/opcu.json	
GET	/r/config/coapd.json	
GET	/r/config/syslog.json	
GET	/contact.json	
GET	/fwup_status	
GET	/iolink/v1/gateway/identification	
GET	/iolink/v1/gateway/capabilities	
GET	/iolink/v1/gateway/configuration	
GET	/iolink/v1/gateway/events	
GET	/iolink/v1/masters	
GET	/iolink/v1/masters/1/capabilities	
GET	/iolink/v1/masters/1/identification	
GET	/iolink/v1/masters/1/ports	
GET	/iolink/v1/masters/1/ports/{port_number}/capabilities	This API is available for all 8 ports. {port_number} should be between "1" and "8".
GET	/iolink/v1/masters/1/ports/{port_number}/status	This API is available for all 8 ports. {port_number} should be between "1" and "8".
GET	/iolink/v1/masters/1/ports/{port_number}/configuration	This API is available for all 8 ports. {port_number} should be between "1" and "8".
GET	/iolink/v1/devices/master1port{port_number}/identification	This API is available for all 8 ports. {port_number} should be between "1" and "8".

Type	API	Note
GET	/iolink/v1/devices/master1port{port_number}/capabilities	This API is available for all 8 ports. {port_number} should be between "1" and "8".
GET	/iolink/v1/devices/master1port{port_number}/processdata/getdata/value	This API is available for all 8 ports. {port_number} should be between "1" and "8".
GET	/iolink/v1/devices/master1port{port_number}/events	This API is available for all 8 ports. {port_number} should be between "1" and "8".

Table 37: REST API access via CoAP

10.4.3 CoAP configuration - Quick start guide



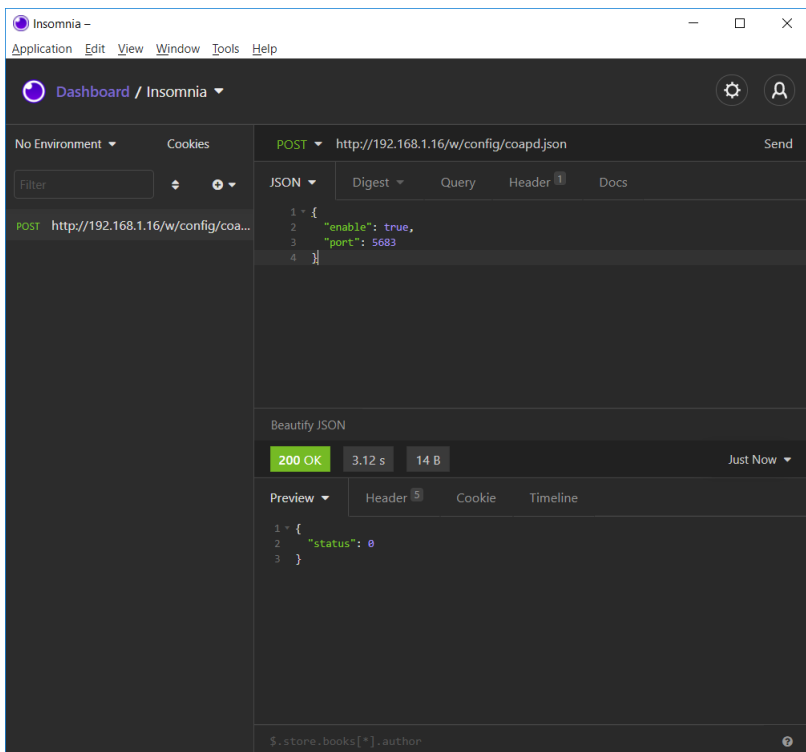
Attention: Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

10.4.3.1 CoAP configuration via JSON

1. Depending on your application case, download and install *Insomnia* or a comparable application: <https://insomnia.rest/download/>

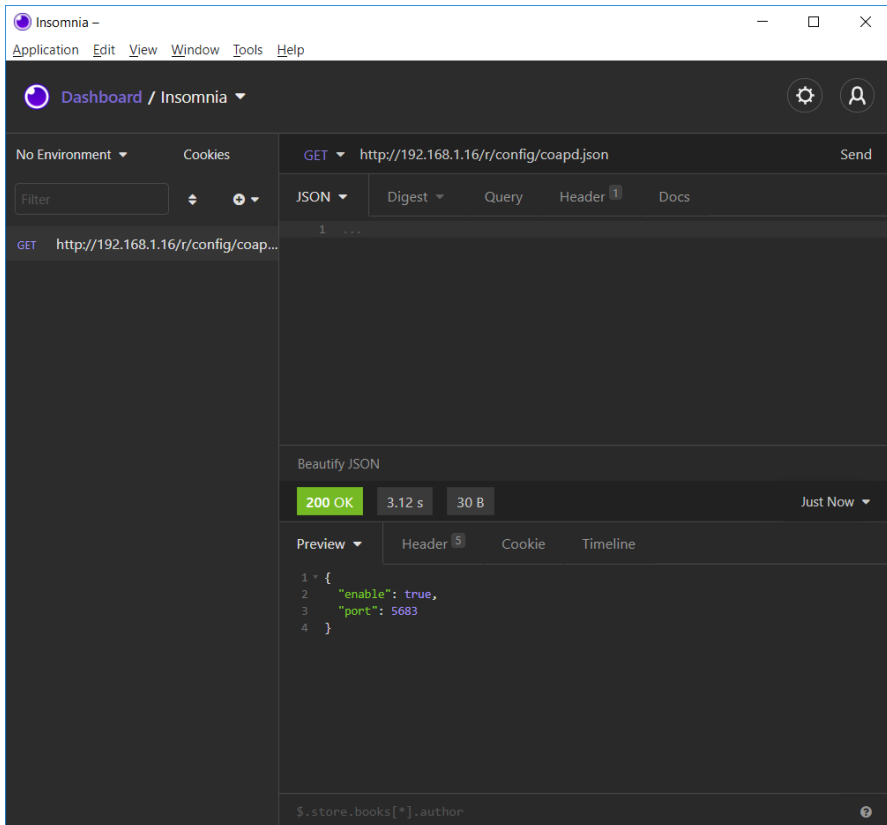
2. Configure CoAP:

POST: [IP-address] /w/config/coapd.json



3. Read CoAP configuration:

GET: [IP-address]/r/config/coapd.json



The screenshot shows the Insomnia REST client interface. The top bar displays the application name 'Insomnia' and standard window controls. Below the menu bar, the 'Dashboard / Insomnia' section is visible. The main workspace is divided into several panels:

- Environment:** Shows 'No Environment' selected.
- Request:** A GET request is defined for the URL 'http://192.168.1.16/r/config/coapd.json'. The response format is set to 'JSON'.
- Response:** The response status is '200 OK', with a response time of '3.12 s' and a body size of '30 B'. The response body is a JSON object:

```
1 * {
2   "enable": true,
3   "port": 5683
4 }
```
- Preview:** A preview of the response body is shown, displaying the same JSON object.

10.5 Syslog

Syslog functions are **only** applicable for the following LioN-X variant:

- ▶ 0980 XSL 3912-121-007D-00F

The LioN-X multiprotocol variants provide a Syslog client which can connect with a configured Syslog server and is able to log messages.

Syslog is a platform-independent standard for logging messages. Each message contains a timestamp as well as information about the severity level and the subsystem. The Syslog protocol RFC5424 is based on the Server-Client principle and lets machines and devices send messages in the network and collect them centrally. (For more details on the used syslog standard, please refer to <https://datatracker.ietf.org/doc/html/rfc5424>.)

LioN-X supports the storage of 256 messages in a ring buffer which are sent to the configured Syslog server. When the ring is full with 256 messages, the oldest message is always replaced by the newly arriving messages. All messages can be saved on the Syslog server. The Syslog client of the IO-Link Master will not store any message permanently.

10.5.1 Syslog configuration

In **delivery state**, Syslog functions are **disabled**. The Syslog client can be configured either using the Web interface or directly via a JSON object sent in an HTTP request. For more information see chapter [Syslog configuration - Quick start guide](#) on page 136.

The configuration URL is:

```
http://[ip-address]/w/config/syslog.json
```

The configuration can also read back as a JSON file:

```
http://[ip-address]/r/config/syslog.json
```

The configuration is a JSON object. Each JSON member is a configuration element. The object must not contain all elements. Only the provided elements will be changed. The configuration changes apply only after a device restart.

The following configuration elements are available (default values in bold):

Element	Data type	Description	Example data
syslog-enable	boolean	Master switch for the Syslog client	true / false
global-severity	integer	<u>Severity level of Syslog client</u> 0 – Emergency 1 – Alert 2 – Critical 3 – Error 4 – Warning 5 – Notice 6 – Info 7 – Debug The client will log all messages of severity according to the setting, including all below levels.	0/1/2/ 3 /4/5/6/7
server-address	string (IP address)	IP address of the Syslog server	192.168.0.51 (Default: null)
server-port	integer (0 to 65535)	Server port of the Syslog server	514
server-severity	integer (0 to 7)	<u>Severity level of Syslog server</u> 0 – Emergency 1 – Alert 2 – Critical 3 – Error 4 – Warning 5 – Notice 6 – Info 7 – Debug	0/1/2/ 3 /4/5/6/7

Table 38: Syslog configuration

Syslog response:

The resulting response is a JSON object with a "status" field. Status should be "0" if no error occurred, and "-1" if there is an error.

In case of an error, the response contains an error array.

The error array contains an error object for each error occurred. The object consists of a field "Element" which names the config element that caused the error, and of a field "Message" for the error message.

Examples:

```
{ "status": -1, "error": [ { "Element": "upcua-enable", "Message": "Boolean
expected" } ] }

{ "status": 0 }

{ "status": -1, "error": [ { "Element": "root", "Message": "Not a JSON
object" } ] }
```

10.5.2 Syslog configuration - Quick start guide

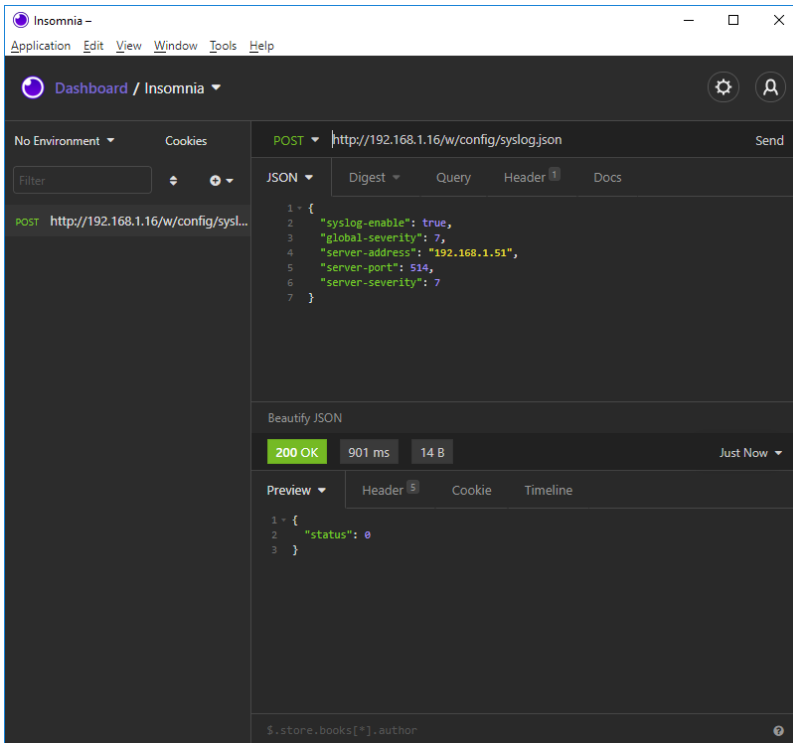
i Attention: Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

10.5.2.1 Syslog configuration via JSON

1. Depending on your application case, download and install *Insomnia* or a comparable application: <https://insomnia.rest/download/>

2. Configure Syslog:

POST: [IP-address]/w/config/syslog.json



The screenshot shows the Insomnia REST client interface. The top bar displays "Insomnia -" and standard window controls. Below the menu bar, the "Dashboard / Insomnia" view is active. The main area shows a REST client configuration for a POST request to "http://192.168.1.16/w/config/syslog.json". The request body is a JSON object:

```

1 {
2   "syslog-enable": true,
3   "global-severity": 7,
4   "server-address": "192.168.1.51",
5   "server-port": 514,
6   "server-severity": 7
7 }

```

The response is a 200 OK status with a 901 ms response time and 14 B body size. The response body is a JSON object:

```

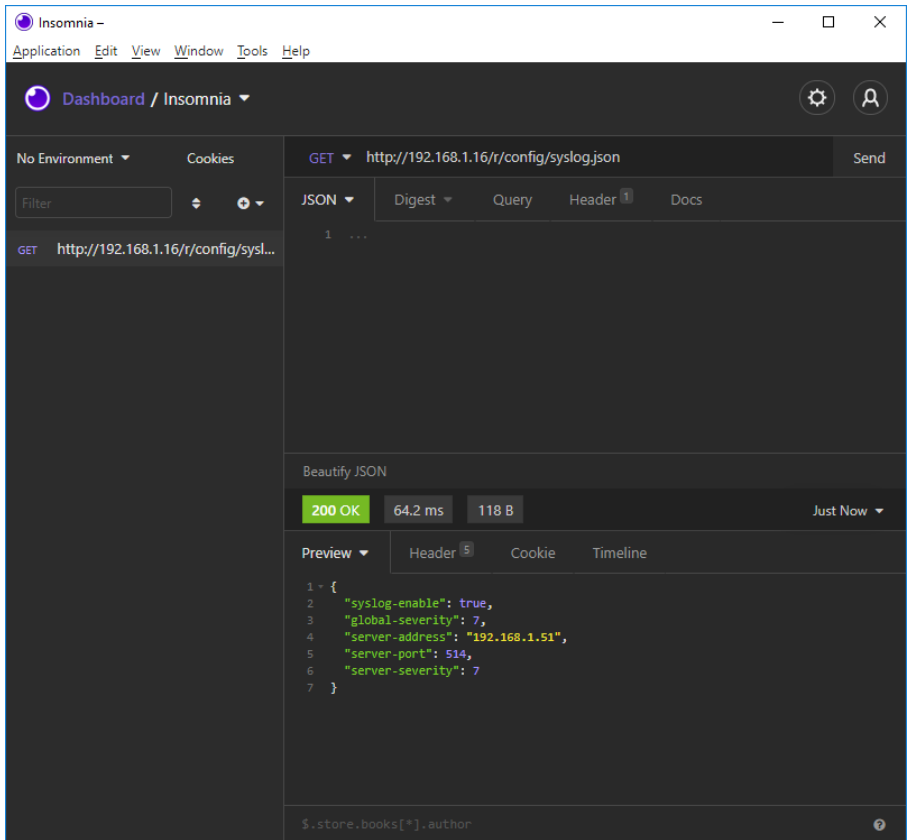
1 {
2   "status": 0
3 }

```

The interface also shows a "Preview" tab with the response body and a "Header" tab with 5 headers. The bottom status bar shows "\$.store.books[*].author".

3. Read Syslog configuration:

GET: [IP-address]/r/config/syslog.json



The screenshot shows the Insomnia REST client interface. The top bar displays "Insomnia -" and standard window controls. Below the menu bar, the "Dashboard / Insomnia" header is visible. The main interface is divided into several sections:

- Environment:** "No Environment" and "Cookies" are shown.
- Request:** A GET request to "http://192.168.1.16/r/config/syslog.json" is configured. The "Send" button is visible.
- Response:** The response is displayed in the "JSON" tab, showing a single line of JSON content.
- Summary:** A "Beautiful JSON" section shows the response status as "200 OK", with a response time of "64.2 ms" and a size of "118 B". The status is updated "Just Now".
- Preview:** The "Preview" tab shows the JSON response content:

```
1 {
2   "syslog-enable": true,
3   "global-severity": 7,
4   "server-address": "192.168.1.51",
5   "server-port": 514,
6   "server-severity": 7
7 }
```
- Footer:** A small footer at the bottom left shows "\$.store.books[*].author" and a help icon.

11 The integrated Web server

LioN-X and the LioN-Xlight variants are equipped with an integrated Web server which makes functions for the device configuration and the display of status and diagnostic information available via a Web interface.

The Web interface provides an overview of the configuration and status of the device. It is also possible to use the Web interface to trigger a reboot, reset to the factory defaults, or perform a firmware update.

Enter `http://` followed by the IP address, such as `http://192.168.1.5`, in your Web browser's address bar. If the status page of the device is not displayed, check your browser and firewall settings.

11.1 LioN-X 0980 XSL... variants

11.1.1 The Status page

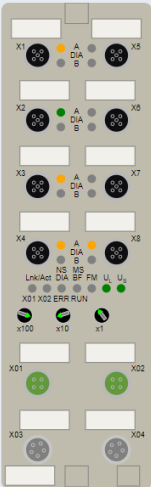


Lion-X Web Interface

Status Ports System User Contact

Status

Device Overview



Device Information

Name LioN-X 8xIO-Link Class A with Multiprotocol

Application Version 10.0.1.26228

Fieldbus Version 1.0.0.0

Bus OPERATE

Device Diagnosis

Forcemode Forcing is locked. Locked

Port Information

Channel	Type	Configuration	State	Dia	Details
X1 A	IO-Link	Digital Input 1 Bit In	On		
X1 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X2 A	IO-Link	IO-Link 4 Bytes In, 4 Bytes Out	Operate		
X2 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X3 A	IO-Link	Digital Output 1 Bit Out	On		
X3 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X4 A	IO-Link	Digital Output 1 Bit Out	On		
X4 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X5 A	IO-Link	Digital Input 1 Bit In	Off		
X5 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X6 A	IO-Link	Digital Input 1 Bit In	Off		
X6 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X7 A	IO-Link	Digital Input 1 Bit In	Off		
X7 B	Digital Input/Output	Digital Input 1 Bit In	Off		
X8 A	IO-Link	Digital Output 1 Bit Out	On		
X8 B	Digital Input/Output	Digital Input 1 Bit In	Off		

The status page provides a quick overview of the current state of the device.

The left side shows a graphical representation of the module with all its LEDs and the positions of the rotary encoding switches.

The right side shows the "Device Information" table with some basic data for the module; for example, the variant, the cyclic communication status and a diagnostic indicator. The indicator shows whether diagnostics for the module exist.

The "Port Information" table shows the configuration and state of the I/O ports.

11.1.2 The Ports page



Lion-X Web Interface

Status Ports System User Contact

Port Details

Show details for port

X1 X2 X3 X4 X5 X6 X7 X8

Port Information		IO-Link	
Forcemode	Forcemode off	Vendor ID	362
Port	X2	Device ID	3674114
Type	IO-Link	Vendor Name	BELDEN Deutschland GmbH
Dia		Vendor Text	www.beldensolutions.com
Port Diagnosis		Product Name	0960 IOL 381-001
• No diagnosis		Product ID	934992002
Pin 4 / Channel A		Product Text	Lion-P IO-Link I/O-Hub, 16DI
Function	IO-Link 4 Bytes In, 4 Bytes Out	Serial No.	123
State	Operate	HW Revision	V1
Pin 2 / Channel B		FW Revision	V3.0.0.0
Function	Digital Input 1 Bit In	Speed	COM3
State	OFF	Cycle time	1000
IO-Link Events		Application Name (Tag)	*** <input type="text"/> <input type="button" value="Set"/>
• No events		Input Data	<input type="text" value="01 00 00 00"/> <input type="button" value="HEX"/>
		Output Data	<input type="text" value="00 00 00 00"/> <input type="button" value="HEX"/>
		Index: <input type="text"/> Subindex: <input type="text" value="0"/>	
		<input checked="" type="radio"/> Dec <input type="radio"/> Hex	
		<input type="button" value="Read"/> <input type="button" value="Write"/> <input type="button" value="System Command"/>	
		Parameter Read/Write	<input type="text"/> <input type="button" value="HEX"/>

The page shows detailed port information. In the field **Port Diagnosis**, incoming and outgoing diagnostics are displayed as clear text. **Pin 2** and **Pin 4** contain information about the configuration and state of the port. For IO-Link ports, additional information relating to the connected sensor and the process data is displayed.

11.1.3 The System page



Lion-X Web Interface

Status Ports System User Contact

System

General Information

Firmware	
Application Version	10.0.1.26228
Fieldbus Version	1.0.0.0
Device	
Name	LioN-X 8xIO-Link Class A with Multiprotocol
Product ID	0980 XSL 3912-121-007D-00F
Ordering Number	935700001
Hardware	1.0
Serial Number	123456
Production Date	2020-12-24T12:00:00Z
Ethernet	
MAC Address	3C:B9:A6:20:05:30
Network	
IP-Address	192.168.0.5
Subnetmask	255.255.255.0
Gateway	192.168.0.5
Source	Manual
Fieldbus	
Name	PROFINET
State	OPERATE

IP Settings

Parameter	Settings
IP-Address	0 . 0 . 0 . 0
Subnet Mask	0 . 0 . 0 . 0
Gateway	0 . 0 . 0 . 0
Startup configuration	<input checked="" type="radio"/> Static <input type="radio"/> DHCP

Submit

MQTT Config	OPC UA Server Config
Mqtt state	Opoua state
Broker	Port
Port	Anonymous login
Base Topic	Listen for Commands
Auto Publish	Process Forcing
Publish Interval (ms)	Change config
Publish Identity	Device Reset
Publish Config	Syslog
Publish Status	Syslog state
Publish Process	Global severity
Publish Devices	Server address
Will State	Server port
Will Topic	Server severity
Listen for Commands	CoAP
Process Forcing	CoAP state
Change Config	Port
Device Reset	
QOS	

Restart device

Confirm to restart the device. All connections will be closed.

Restart

Reset configuration to factory defaults

Restoring factory settings affects all network parameters, including fieldbus specific settings. All network connections will be closed.

Note: If the module has rotary switches, the new IP address is equivalent to the rotary switch position.

Confirm to reset the device. All configuration data will be overwritten by default values!

Factory Reset

Firmware update

FW-Update

The System page shows the basic information for the module like **Firmware** version, **Device** information, **Ethernet**, **Network** and **Fieldbus** information.

Restart Device

The module initializes a software reset.

Reset to Factory Settings

The module restores to the default factory settings.

IP Settings

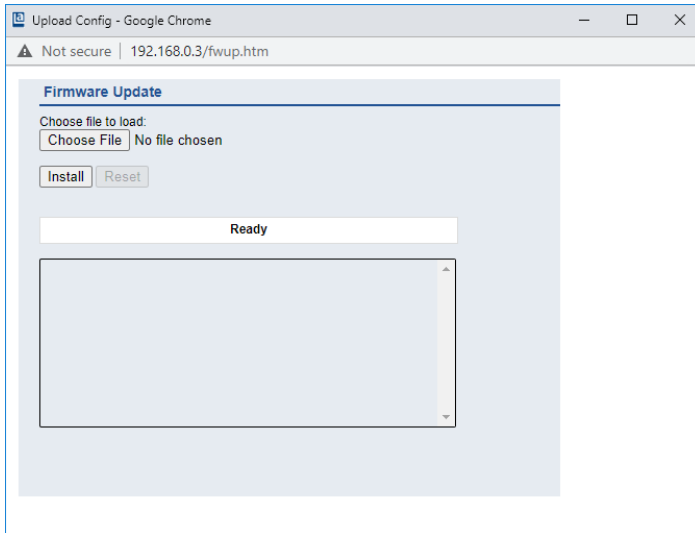
Use this parameter to change the current IP address of the module.

For PROFINET, this is only useful during commissioning. Normally, the PLC sets the IP address at start-up by detecting the PROFINET module via its device name.

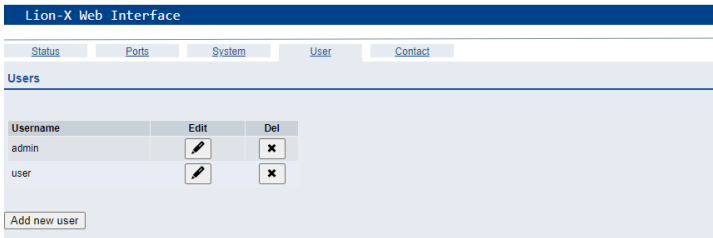
Firmware Update

The module initializes a Firmware update.

For a firmware update choose the *.ZIP container available on our website or ask our support team. Afterwards follow the instructions shown on your screen.



11.1.4 The User page



The User page provides the user management of the Web interface. New users with access rights **Admin** or **Write** can be added here. For security reasons please change the default admin password immediately after configuring the device.

Default user login data:

- ▶ User: admin
- ▶ Password: private

11.2 LioN-Xlight 0980 LSL... variants

11.2.1 The System page



LioN-X Webserver

System | Contact

System

General Information

Firmware	
Version	10.0.0
Device	
Name	LioN-Xlight 8xIO-Link Class A with Profinet
Product ID	0980 LSL 3010-121-0006-001
Ordering Number	935701001
Hardware	1.0
Serial Number	123456
Production Date	2020-12-24T12:00:00Z
Ethernet	
MAC Address	3C:B9:A6:20:05:30
Network	
IP-Address	192.168.0.3
Subnetmask	255.255.255.0
Gateway	192.168.0.3
Fieldbus	
Name	PROFINET
State	OPERATE

IP Settings

Parameter	Settings
IP-Address	192 . 168 . 0 . 3
Subnet Mask	255 . 255 . 255 . 0
Gateway	192 . 168 . 0 . 3
Startup configuration	<input checked="" type="radio"/> Static <input type="radio"/> DHCP

[Restart device](#)

Confirm to restart the device. All connections will be closed.

[Reset configuration to factory defaults](#)

Restoring factory settings affects all network parameters, including fieldbus specific settings. All network connections will be closed.

Note: If the module has rotary switches, the new IP address is equivalent to the rotary switch position.

Confirm to reset the device. All configuration data will be overwritten by default values!

[Firmware update](#)

The System page shows the basic information for the module like **Firmware** version, **Device** information, **Ethernet**, **Network** and **Fieldbus** information.

Restart Device

The module initializes a software reset.

Reset to Factory Settings

The module restores to the default factory settings.

IP Settings

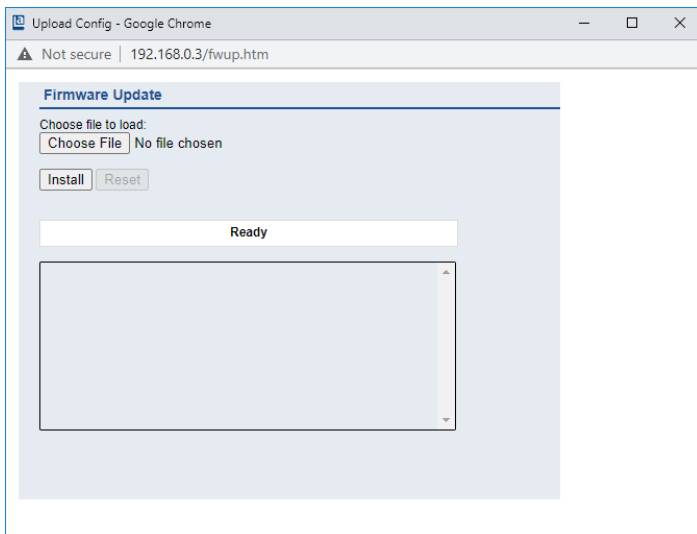
Use this parameter to change the current IP address of the module.

For PROFINET, this is only useful during commissioning. Normally, the PLC sets the IP address at start-up by detecting the PROFINET module via its device name.

Firmware Update

The module initializes a Firmware update.

For a firmware update choose the *.ZIP container available on our website or ask our support team. Afterwards follow the instructions shown on your screen.



12 Firmware update

A firmware update of the device is possible via the integrated Web server using the EoE protocol (Ethernet over EtherCAT®), or the FoE protocol (File over EtherCAT®).



Warning: Risk of data loss, damage to the device and injury due to uncontrolled machine movements.

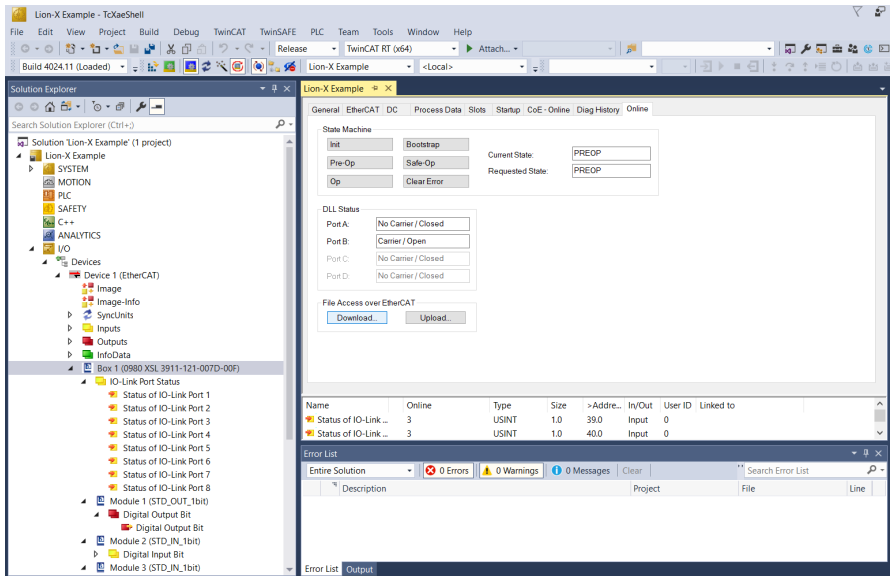
- ▶ Do **NOT** interrupt the power supply to the device during the update.

12.1 Firmware update via FoE

The FoE service must be supported by the IOL-Master and the IOL-Device. The FoE service on the Lion-X variants is supported by default. If the FoE service is configured and the device is in "Pre-Op" state, an update via FoE can be done by using TwinCAT® like in the following example:

1. Rename the file name of the firmware update file provided by Belden to "firmware".

2. In TwinCAT®, select the device you want to update:



3. In the device window on the right, go to the box **File Access over EtherCAT** and press **Download**.

4. In the following window, select the update file provided by Belden:

Edit FoE Name ✕

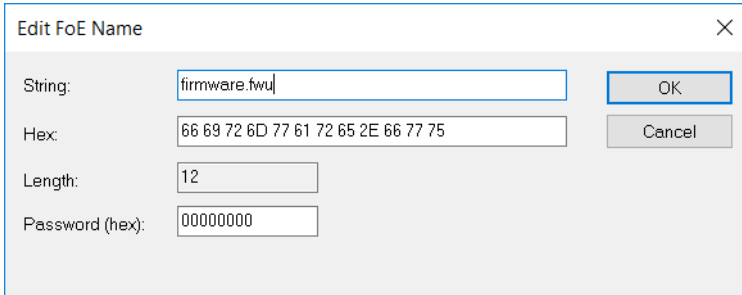
String: OK

Hex: Cancel

Length:

Password (hex):

4. Add the file extension “.fwu” to the file name in field **String**: if not visible:



The screenshot shows a dialog box titled "Edit FoE Name" with a close button (X) in the top right corner. It contains four input fields and two buttons:

- String:** A text box containing "firmware.fwu".
- Hex:** A text box containing "66 69 72 6D 77 61 72 65 2E 66 77 75".
- Length:** A text box containing "12".
- Password (hex):** A text box containing "00000000".
- Buttons:** An "OK" button (highlighted in blue) and a "Cancel" button (greyed out).

5. Press **OK** and wait until the file has been transferred to the device.



Attention: After the file has been transferred, the device will reset automatically. During the restart, older firmware update files will be replaced by the files in the update package.

13 Technical data

The following sections give an overview of the most important functional data needed to operate the device. For further information and detailed technical data, see the respective **Data Sheet** of your required product in the product specific download area on catalog.belden.com.

13.1 General

Protection class (Only applies if the connectors are screwed together or if protective caps are used.) ²	IP65 IP67 IP69K	
Ambient temperature (during operation and storage)	0980 XSL 3x12-121...	-40 °C to +70 °C (-40 °F to +158 °F)
	0980 LSL 3x11-121...	-20 °C to +60 °C (-4 °F to +140 °F)
	0980 LSL 3x10-121...	
Weight	LioN-X 60 mm	ca. 500 gr.
Ambient moisture	Max. 98% RH (For UL applications: Max. 80% RH)	
Housing material	Die-cast zinc	
Surface finish	Frosted nickel	
Flammability class	UL 94 (IEC 61010)	
Vibration resistance (oscillation) DIN EN 60068-2-6 (2008-11)	15 g / 5-500 Hz	
Shock resistance DIN EN 60068-2-27 (2010-02)	50 g / 11 ms +/- X,Y,Z	
Fastening torques	M4 fixing screws	1 Nm
	M4 ground connection	1 Nm
	M12 connector	0.5 Nm
Permitted cables	Ethernet cables according to IEEE 802.3, min. CAT 5 (shielded) Max. length of 100 m, not routed out of facility (= local network)	

Table 39: General information

² Not under UL investigation.

13.2 EtherCAT® protocol

Protocol	EtherCAT® (ETG.1000 V1.2)
ESI file	LumbergAutomation-LioN-X-IO-Link.xml
Transmission rate	100 Mbit/s, full duplex
Type of addressing	Auto-increment addressing, Fixed addressing
Min. cycle time	1 ms
Vendor ID	16A _H
Device ID	0x0400 (same for all LioN-X devices)
Mailbox protocols	CanOpen over EtherCAT® (CoE) File access over EtherCAT® (FoE) Ethernet over EtherCAT® (EoE)
Supported Ethernet protocols	Ping ARP HTTP TCP/IP
Switch functionality	Integrated
EtherCAT® interface Port	2x M12 sockets 4-pin, D-coded (see pin assignments)

Table 40: EtherCAT® protocol

13.3 Power supply of the module electronics/sensors

Nominal voltage U_S	24 V DC (SELV/PELV)		
Voltage range	18-30 V DC		
Power consumption of module electronics	Typically 160 mA (+/-20 % at U_S nominal voltage)		
Voltage level of the sensor power supply	Min. ($U_S - 1.5$ V)		
Voltage ripple U_S	Max. 5 %		
Power supply interruption	Max. 10 ms		
Current consumption sensor system (L+ / Pin 1)	0980 XSL 3912-121...	Port X1 – X8 (Pin 1)	max. 4 A per port (at $T_{\text{ambient}} = 30^\circ \text{C}$)
	0980 LSL 3x11-121...	Port X1 – X8 (Pin 1)	max. 2 A per port (at $T_{\text{ambient}} = 30^\circ \text{C}$)
	0980 LSL 3x10-121...	Port X1 – X4 (L+ / Pin 1)	max. 2 A per port (at $T_{\text{ambient}} = 30^\circ \text{C}$)
		Port X5 – X8 (Pin 1)	max. 0.7 A in total for ports X5 – X8
Short circuit/overload protection of sensor supply	Yes, per port		
Reverse polarity protection	Yes		
Operational indicator (U_S)	LED green:	$18 \text{ V } (+/- 1 \text{ V}) < U_S$	
	LED red:	$U_S < 18 \text{ V } (+/- 1 \text{ V})$	
Port X03, X04	M12-L-coded Power, connector/socket, 5-pole Pin 1 / Pin 3		

Table 41: Information on the power supply of the module electronics/sensors

13.4 Power supply of the actuators

Nominal voltage U_L	24 V DC (SELV/PELV)
Voltage range	18-30 V DC
Voltage ripple U_L	Max. 5 %
Power supply interruption	Max. 10 ms
Reverse polarity protection	Yes
Operational indicator (U_L)	LED green: $18 \text{ V } (+/- 1 \text{ V}) < U_L$ LED red: $U_L < 18 \text{ V } (+/- 1 \text{ V})$ or $U_L > 30 \text{ V } (+/- 1 \text{ V})$ * if "Report U_L supply voltage fault" is enabled.
Port X03, X04	M12_L-coded Power, connector/socket, 5-pole Pin 2 / Pin 4

Table 42: Information on the power supply of the actuators

13.5 IO-Link Master ports Class A, Pin 4

0980 XSL 3912-121...	Port X1 – X8	M12 socket, 5-pin, Pin 4
0980 LSL 3x11-121...		
0980 LSL 3x10-121...	Port X1 – X4	

Table 43: IO-Link Master ports, Class A (Ch. A / C/Q / Pin 4)

13.5.1 Configured as a digital input

Input connection	0980 XSL 3912-121...		Type 1 as per IEC 61131-2
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...	X1 - X4	Type 1 as per IEC 61131-2
		X5 - X8	Type 1 as per IEC 61131-2
Nominal input voltage	24 V DC		
Input current	Typically 3 mA		
Channel type	Normally open, p-switching		
Number of digital inputs	0980 XSL 3912-121...		8
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...		
Status indicator	yellow LED		
Diagnostic indicator	red LED per channel		

Table 44: IO-Link Master Class A ports, pin 4, configured as digital inputs

13.5.2 Configured as a digital output



Attention: For LioN-X variants, the power for the outputs is supplied via the U_L power supply.



Attention: For LioN-Xlight variants, the power for the outputs is supplied via the U_S power supply.

Output type	normally open, p-switching	
Nominal output current per channel		
Signal status "1"	min. ($U_L - 1\text{ V}$)	
Signal status "0"	max. 2 V	
Max. output current per device	0980 XSL 3912-121...	16 A (M12 Power)
	0980 LSL 3x11-121...	4 A
	0980 LSL 3x10-121...	2 A
Max. output current per channel ³	0980 XSL 3912-121...	2 A
	0980 LSL 3x11-121...	0.5 A (power supplied via U_S)
	0980 LSL 3x10-121...	0.25 A for UL applications
Short-circuit/overload protected	yes/yes	
Behavior in case of short circuit or overload	disconnection with automatic power-on	
Number of digital outputs	0980 XSL 3912-121...	8
	0980 LSL 3x11-121...	
	0980 LSL 3x10-121...	4
Status indicator	yellow LED per output	
Diagnostic indicator	red LED per channel	

Table 45: IO-Link Master Ports configured as digital outputs

³ Max. 2.0 A per channel; max. 6.5 A in total (for **UL applications** max. 5.0 A in total) for every port pair (X1/X2, X3/X4, X5/X6, X7/X8); max. 9.0 A in total (with derating) for the whole port group (X1 .. X8).

13.5.3 Configured as an IO-Link port in COM mode

IO-Link Master specification	v1.1.3 ready, IEC 61131-9
Communication rates	4.8 kbaud (COM 1) 38.4 kbaud (COM 2) 230.4 kbaud (COM 3)
Line lengths in the IO-Link Device	max. 20 m
Number of IO-Link ports	8
Min. IO-Link cycle time	400 μ s

Table 46: As an IO-Link port in COM mode

13.6 IO-Link Master ports Class A, Pin 2

0980 XSL 3912-121...	Port X1 – X8	M12 socket, 5-pin, Pin 2
0980 LSL 3x11-121...		
0980 LSL 3x10-121...		

Table 47: IO-Link Master ports, Class A (Ch. B, Pin 2)

13.6.1 Configured as a digital input

Input connection	0980 XSL 3912-121...		Type 1 as per IEC 61131-2
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...	X1 - X4	Type 1 as per IEC 61131-2
		X5 - X8	Type 1 as per IEC 61131-2
Nominal input voltage	24 V DC		
Input current	Typically 3 mA		
Channel type	Normally open, p-switching		
Number of digital inputs	0980 XSL 3912-121...		8
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...		
Status indicator	white LED		
Diagnostic indicator	red LED per channel		

Table 48: IO-Link Master Class A ports, pin 2, configured as digital inputs

13.6.2 Configured as a digital output

i Attention: For LioN-X variants, the power for the outputs is supplied via the U_L power supply.

i Attention: For LioN-Xlight variants, the power for the outputs is supplied via the U_S power supply.

Output type	normally open, p-switching	
Nominal output current per channel Signal status "1" Signal status "0"	min. ($U_L - 1$ V) max. 2 V	
Max. output current per device	0980 XSL 3912-121...	16 A (M12 Power)
	0980 LSL 3x11-121...	4 A
	0980 LSL 3x10-121...	2 A
Max. output current per channel ⁴	0980 XSL 3912-121...	2 A
	0980 LSL 3x11-121...	0 A (no outputs)
	0980 LSL 3x10-121...	
Short-circuit/overload protected	yes/yes	
Behavior in case of short circuit or overload	disconnection with automatic power-on	
Number of digital outputs	0980 XSL 3912-121...	8
	0980 LSL 3x11-121...	–
	0980 LSL 3x10-121...	–
Status indicator	white LED per output	
Diagnostic indicator	red LED per channel	

Table 49: IO-Link Master Ports configured as digital outputs

⁴ Max. 2.0 A per channel; max. 6.5 A in total (for **UL applications** max. 5.0 A in total) for every port pair (X1/X2, X3/X4, X5/X6, X7/X8); max. 9.0 A in total (with derating) for the whole port group (X1 .. X8).

13.7 LEDs

U _L	Green	Auxiliary sensor/actuator voltage OK $18\text{ V (+/- 1 V)} < U_L < 30\text{ V (+/- 1 V)}$
	Red*	Auxiliary sensor/actuator voltage LOW $U_L < 18\text{ V (+/- 1 V)}$ or $U_L > 30\text{ V (+/- 1 V)}$ * if "Report U _L supply voltage fault" is enabled.
	off	None of the above conditions
U _S	Green	System/sensor voltage OK $18\text{ V (+/- 1 V)} < U_S < 30\text{ V (+/- 1 V)}$
	Red	System/sensor voltage LOW $U_S < 18\text{ V (+/- 1 V)}$ or $U_S > 30\text{ V (+/- 1 V)}$
	off	None of the above conditions
X1 ... X8 A	Green	IO-Link COM Mode: IO-Link communication exists
	Flashing green	IO-Link COM Mode: No IO-Link communication.
	Yellow	Standard-IO Mode: Status of digital input or output on C/Q (pin 4) line "on"
	off	None of the above conditions
X1 ... X8 B	White	Status of digital input or digital output on pin 2 line "on"
	Red	Short circuit on C/Q (pin 4) line / All modes: Overload or short circuit on L+ (pin 1) line / communication error
	off	None of the above conditions
P1 Lnk/ Act P2 Lnk/ Act	Green	Ethernet connection to another subscriber exists. Link detected.
	Flashing yellow	Data exchange with another subscriber.
	off	No connection to another subscriber. No link, no data exchange.

BF	Red	Bus fault. No configuration, no or slow physical connection
	Red flashing at 2 Hz	Link exists but no communication link to the EtherCAT® controller
	off	EtherCAT® controller has established an active connection to the device
DIA	Red	EtherCAT® module diagnostic alarm active
	Red flashing at 1 Hz	Watchdog time-out; fail safe mode is active
	Red flashing at 2 Hz, 3 sec	DCP signal service is initiated via the bus
	Red double flash	Firmware update
	off	None of the above conditions

Table 50: Information on the LED colors

14 Accessories

In order to get access to various types of accessories, please visit our Web page:

<http://www.beldensolutions.com>

15 References

1. EtherCAT® Specification - Part 5: Application Layer service definition, ETG.1000.5 V1.0.4, ETG 2017-09.
2. EtherCAT® Specification - Part 6: Application Layer protocol specification, ETG.1000.6 V1.0.4, ETG 2017-09.
3. EtherCAT® Protocol Enhancements, ETG.1020 V1.2.0, ETG 2015-12.
4. EtherCAT® Indicator and Labeling Specification, ETG.1300 V1.1.1, ETG 2015-07.
5. EtherCAT® Slave Information Specification, ETG.2000 V1.0.10, ETG 2018-02.
6. EtherCAT® Modular Device Profile Part 1: General MDP Device Model, ETG.5001.1 V0.9.0, ETG 2016-07.
7. EtherCAT® Modular Device Profile Part 6220: IO-Link Master, ETG.5001.6220 V1.0.5, ETG 2017-04.
8. Protocol API EtherCAT® Slave V4.7.0, Hilscher Gesellschaft für Systemautomation mbH, DOC110909API10EN, Revision 10, 2017-10.
9. IO-Link Interface and System Specification Version 1.1.2, Order No: 10.002, IO-Link Community 2013-07.