

# Manual

## EtherNet/IP

### **LioN-X IO-Link Master Multiprotocol:**

**0980 XSL 3912-121-007D-00F (8 x IO-Link Class A)**

**0980 XSL 3912-121-007D-01F (8 x IO-Link Class A)**

**0980 XSL 3912-121-027D-01F (8 x IO-Link Class A)**

**0980 XSL 3913-121-007D-01F (8 x IO-Link Class A/B Mixmodule)**

**0980 XSL 3913-121-027D-01F (8 x IO-Link Class A/B Mixmodule)**

### **LioN-Xlight IO-Link Master EtherNet/IP:**

**0980 LSL 3111-121-0006-002 (8 x IO-Link Class A)**

**0980 LSL 3110-121-0006-002 (4 x IO-Link Class A + 8 x DI)**



# Contents

<b>1 About this manual</b>	<b>10</b>
1.1 General information	10
1.2 Explanation of symbols	11
1.2.1 Use of danger information	11
1.2.2 Use of general information	11
1.3 Version information	12
<b>2 Safety instructions</b>	<b>14</b>
2.1 Intended use	14
2.2 Qualified personnel	15
<b>3 Designations and synonyms</b>	<b>16</b>
<b>4 System description</b>	<b>20</b>
4.1 About LioN-X	20
4.2 Device variants	21
4.3 I/O port overview	23
<b>5 Overview of product features</b>	<b>26</b>
5.1 EtherNet/IP product features	26
5.2 I/O port features	28
5.3 Integrated Web server	29
5.4 Security features	30
5.5 Other features	31

<b>6 Assembly and wiring</b>	<b>32</b>
6.1 General information	32
6.2 Outer dimensions	33
6.2.1 LioN-X multiprotocol variants	33
6.2.2 LioN-Xlight variants with EtherNet/IP	38
6.2.3 Notifications	40
6.3 Port assignments	41
6.3.1 Ethernet ports, M12 socket, 4-pin, D-coded	41
6.3.2 Power supply with M12 power L-coded	42
6.3.2.1 IO-Link Master with Class A ports	43
6.3.2.2 IO-Link Master with Class A/B ports	44
6.3.3 I/O ports as M12 sockets	44
6.3.3.1 IO-Link ports (Class A and Class B)	45
<b>7 Starting operation</b>	<b>47</b>
7.1 EDS file	47
7.2 MAC addresses	47
7.3 State on delivery	48
7.4 Setting network parameters	49
7.4.1 IP address for LioN-X variants	49
7.4.2 IP address for LioN-Xlight variants	49
7.5 Setting the rotary encoding switches	50
7.5.1 EtherNet/IP selection and IP configuration via rotary encoding switches	53
7.5.2 Factory reset	54
<b>8 Configuration EtherNet/IP</b>	<b>55</b>
8.1 Assembly types	55
8.2 Connections	56
8.2.1 IO-Link parameters (Exclusive Owner)	57

8.2.2 IO-Link parameters (Listen Only)	57
--	----

## **9 Configuration parameters** **59**

9.1 General settings	60
9.1.1 Force mode lock	61
9.1.2 Web interface lock	61
9.1.3 Digital Output Control	61
9.1.4 Report $U_L/U_{AUX}$ supply voltage fault	61
9.1.5 Report DO Fault without $U_L/U_{Aux}$	61
9.1.6 CIP object configuration lock	61
9.1.7 External configuration lock	62
9.1.8 IO Mapping Mode	62
9.2 Channel settings	64
9.2.1 IO Mapping (Ch1 .. 16)	66
9.2.2 DO Surveillance Timeout (Ch1 .. 16)	66
9.2.3 DO Failsafe (Ch1 .. 16)	66
9.2.4 DO Restart Mode (Ch1 .. 16)	67
9.2.5 DO Switch Mode (Ch1 .. 16)	67
9.2.6 DI Logic (Ch1 .. 16)	68
9.2.7 DI Filter (Ch1 .. 16)	68
9.2.8 Channel Mode (Ch1 .. 16)	68
9.3 IO-Link diagnosis settings	71
9.3.1 IO-Link Master Diagnosis	71
9.3.2 IO-Link Device Error	71
9.3.3 IO-Link Device Warning	71
9.3.4 IO-Link Device Notification	72
9.3.5 IO-Link Device Diagnosis Port 1 .. 8	72
9.4 IO-Link Port 1 .. 8 settings	73
9.4.1 Output Data Size	75
9.4.2 Input Data Size	76
9.4.3 Input Data Extension	76
9.4.4 Output Data Swapping Mode	76
9.4.5 Output Data Swapping Offset	77
9.4.6 Input Data Swapping Mode	77

9.4.7 Input Data Swapping Offset	77
9.4.8 IOL Failsafe	77
9.4.9 Port Mode	78
9.4.10 Validation and Backup	79
9.4.11 Vendor ID	82
9.4.12 Device ID	82
9.4.13 Cycle Time	82
<b>10 Process data assignment</b>	<b>83</b>
10.1 Consuming data image (output)	83
10.1.1 Digital output channel control	83
10.1.2 IO-Link output data	84
10.2 Producing data image (input)	84
10.2.1 Digital input channel status	85
10.2.2 General diagnostics	85
10.2.3 Sensor diagnostics	86
10.2.4 Actuator/ $U_L/U_{Aux}$ diagnostics	86
10.2.5 IO-Link diagnostics	87
10.2.6 IO-Link input data	88
10.3 Sample applications	91
10.3.1 Process data images – default configuration	91
10.3.2 Process data images with modified data sizes	93
<b>11 Configuration and operation with Rockwell Automation Studio 5000®</b>	<b>97</b>
11.1 Basic commissioning	97
11.2 Add-On Instruction (AOI)	102
<b>12 CIP object classes</b>	<b>111</b>
12.1 EtherNet/IP object classes	111

12.1.1 Identity Object (0x01)	112
12.1.2 Assembly Object (0x04)	115
12.1.3 Discrete Input Point Object (0x08)	116
12.1.4 DLR Object (0x47)	117
12.1.5 QoS Object (0x48)	119
12.1.6 TCP/IP Object (0xF5)	121
12.1.7 Ethernet Link Object (0xF6)	123
12.1.8 LLDP Management Object (0x109)	126
12.2 Vendor specific object classes	128
12.2.1 General Settings Object (0xA0)	128
12.2.2 Channel Settings Object (0xA1)	130
12.2.3 IO-Link Diagnosis Settings Object (0xA2)	132
12.2.4 IO-Link Port Settings Object (0xA3)	133
12.2.5 IO-Link Failsafe Parameter Object (0xA4)	136
12.2.6 IO-Link Device Parameter Object (0xA5)	137
12.3 Message configuration in Rockwell Automation Studio 5000®	141
12.3.1 Get/Set attribute single	142
12.3.2 Get/Set ISDU data	143

## **13 Diagnostics processing 145**

13.1 Error of the system/sensor power supply	145
13.2 Error of the auxiliary/actuator power supply	146
13.3 Overload/short-circuit of the I/O port sensor supply outputs	147
13.4 Overload/short circuit of the digital outputs	148
13.5 IO-Link COM error	150
13.6 IO-Link validation error	151
13.7 IO-Link device diagnostics	152

## **14 IIoT functionality 153**

14.1 MQTT	154
14.1.1 MQTT configuration	154
14.1.2 MQTT topics	157

14.1.2.1 Base topic	157
14.1.2.2 Publish topic	160
14.1.2.3 Command topic (MQTT Subscribe)	165
14.1.3 MQTT configuration - Quick start guide	169
14.1.3.1 MQTT configuration via JSON	169
14.2 OPC UA	171
14.2.1 OPC UA configuration	172
14.2.2 OPC UA address space	174
14.2.3 OPC UA configuration - Quick start guide	175
14.2.3.1 OPC UA configuration via JSON	175
14.3 REST API	177
14.3.1 Standard device information	179
14.3.2 Structure	180
14.3.3 Configuration and forcing	186
14.3.4 Reading and writing ISDU parameters	188
14.3.4.1 Reading ISDU	188
14.3.4.2 Writing ISDU	190
14.3.5 Example: Reading ISDU	192
14.3.6 Example: Writing ISDU	192
14.4 CoAP server	193
14.4.1 CoAP configuration	193
14.4.2 REST API access via CoAP	194
14.4.3 CoAP configuration - Quick start guide	197
14.4.3.1 CoAP configuration via JSON	197
14.5 Syslog	199
14.5.1 Syslog configuration	199
14.5.2 Syslog configuration - Quick start guide	202
14.5.2.1 Syslog configuration via JSON	202
14.6 Network Time Protocol (NTP)	204
14.6.1 NTP configuration	204
14.6.2 NTP configuration - Quick start guide	206
14.6.2.1 NTP configuration via JSON	206

## 15 The integrated Web server

208

15.1 LioN-X 0980 XSL... variants	209
15.1.1 The Status page	209
15.1.2 The Ports page	210
15.1.3 The System page	211
15.1.3.1 HTTPS	213
15.1.3.2 HTTPS certificate manager	213
15.1.4 The User page	213
15.1.5 Certificate creation – example	214
15.2 LioN-Xlight 0980 LSL... variants	217
15.2.1 The System page	217

## **16 IODD** **219**

16.1 IO-Link Device parameters and ISDU requests	219
16.2 Web GUI functionality	220
16.2.1 Port Details page	220
16.2.2 Parameters page	222
16.2.3 IODD Management page	223

## **17 Technical data** **224**

17.1 General	225
17.2 EtherNet/IP protocol	226
17.3 Power supply of the module electronics/sensors	227
17.4 Power supply of the actuators	228
17.4.1 IO-Link Class A devices ( $U_L$ )	228
17.4.2 IO-Link Class A/B devices ( $U_{AUX}$ )	228
17.5 I/O ports Channel A (Pin 4)	229
17.5.1 Configured as digital input, Ch. A (Pin 4)	229
17.5.2 Configured as digital output, Ch. A (Pin 4)	230
17.5.3 Configured as IO-Link port in COM mode, Ch. A	231
17.6 I/O ports Channel B (Pin 2)	232
17.6.1 Configured as a digital input, Ch. B (Pin 2)	232
17.6.2 Configured as a digital output, Ch. B (Pin 2)	233



17.7 LEDs	236
17.8 Data transfer times	239

## **18 Accessories** **241**

# 1 About this manual

## 1.1 General information

Read the assembly and operating instructions on the following pages carefully before starting up the modules. Keep this information where it is accessible to all users.

The texts, figures, diagrams, and examples used in this document are exclusively used 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](mailto:lumberg-automation-support.belden.com)  
[www.lumberg-automation.com](http://www.lumberg-automation.com)  
[catalog.belden.com](http://catalog.belden.com)

Belden Deutschland GmbH – Lumberg Automation™ – reserves the right to make technical changes or changes to this document 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.

## 1.3 Version information

Version	Created	Changes
1.0	03/2021	
1.1	04/2021	
1.2	05/2021	
1.3	11/2021	New chapters: 10.3 / 10.3.1 / 10.3.2 / 11.1 / 11.2 Ch. 4.3 Ch. 9.4: Default values Ch. 10.x: Byte range Ch. 12.2.x: Instances
2.0	03/2022	New chapters: Ch. 12.1.8 ("LLDP") Ch. 14.6 ("NTP") Ch. 16 ("IODE")  New device variants: 0980 XSL 3912-121-007D-01F 0980 XSL 3913-121-007D-01F
2.1	06/2022	Temporarily excluded device variant information for 0980 XSL 3913-121-007D-01F (shipping in 2023)
2.2	10/2022	Device variant information for 0980 XSL 3913-121-007D-01F included. Ch. 7.4: LED description
2.3	12/2022	Ch. 9.1 ("External configuration lock")
2.4	04/2023	Ch. 12.2.6: Description "Instance 1 .. n*", new content Ch. 12.3: Reworked, new sub chapters 12.3.1 + 12.3.2
2.5	07/2023	Warning in ch. <a href="#">Setting the rotary encoding switches</a> on page 50

Version	Created	Changes
2.6	10/2023	Added new feature HTTPS (several chapters updated). New device variants: 0980 XSL 3912-121-027D-01F 0980 XSL 3913-121-027D-01F

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 (2014/30/EU) and the low voltage guideline (2014/35/EU).

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 Belden Deutschland GmbH – 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 all provided device documentation 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 document, can result in serious personal injury or damage to equipment.



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

## 3 Designations and synonyms

AOI	Add-On Instruction
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)
CC	CC-Link IE Field
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
CSP+	Control & Communication System Profile Plus
DAT	Device Acknowledgement Time
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



### 3 Designations and synonyms

---

ERP	Enterprise Resource Planning system
ETH	ETHERNET
FE	Functional Earth
FME	Force Mode Enabled (EIP diagnostics)
FS	Functional Safety
FSU	Fast Start-Up
GSDML	General Station Description Markup Language
High-B	High-Byte
HTTPS	Hyper Text Transfer Protocol Secure
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/O	Input / Output
I/O port	X1 .. X8
I/O port pin 2	Channel B of I/O ports
I/O port pin 4 (C/Q)	Channel A of I/O ports
IODD	I/O Device Description
IOL or IO-L	IO-Link
I/Q	I/O port pin 2 mode, Digital Input/switching signal
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
MP	Multiprotocol: PROFINET + EtherNet/IP + EtherCAT® + Modbus TCP (+ CC-Link IE Field Basic)
MQTT	Message Queuing Telemetry Transport (open networking protocol)
MSB	Most Significant Bit
M12	Metric thread according to DIN 13-1 with 12 mm diameter
NTP	Network Time Protocol
OFDT	One Fault Delay Time
OLE	Output process data Length Error (EIP diagnostics)
OPC UA	Open Platform Communications Unified Architecture (platform independent, service-oriented architecture)
PFH	Probability of dangerous Failure per Hour [h <sup>-1</sup> ]
PD	Process Data
PDCT	Port and Device Configuration Tool
PLC	Programmable Logic Controller
PN	PROFINET
PWR	Power
Qualifier	Validity on a process value. Valid = "1"
REST	REpresentational State Transfer
RFC	Request for Comments
RPI	Requested Packet Interval
RWr	Word data input as seen from the master station (CC-Link)
RWw	Word data output as seen from the master station (CC-Link)
RX	Bit data input as seen from the master station (CC-Link)
RY	Bit data output as seen from the master station (CC-Link)

### 3 Designations and synonyms

SCA	Short Circuit Actuator/ $U_L$ / $U_{AUX}$ (EIP diagnostics)
SCS	Short Circuit Sensor (EIP diagnostics)
SFRT	Safety Function Response Time
SIO mode	Standard Input Output mode
SLMP	Seamless Message Protocol
SNMP	Simple Network Management Protocol
SP	Single Protocol (PROFINET, EtherNet/IP, EtherCAT®, Modbus TCP or CC-Link IE Field Basic)
SPE	Startup Parameterization Error (EIP diagnostics)
T-B	Test Channel B
T-A	Test Channel A
$U_{AUX}$	$U_{Auxiliary}$ , supply voltage for the load circuit (Actuator supply on Class B ports of Class A/B IO-Link Master)
UDP	User Datagram Protocol
UDT	User-Defined Data Types
UINT8	Byte in PLC (IB, QB)
UINT16	Unsigned integer with 16 bits or word in PLC (IW, QW)
$U_L$	$U_{Load}$ , supply voltage for the load circuit (Actuator supply on Class A IO-Link Master)
UL	Underwriters Laboratories Inc. (certification company)
UTC	Coordinated Universal Time (Temps Universel Coordonné)
WCDT	Worst Case Delay Time

*Table 2: Designations and synonyms*

## 4 System description

The LioN modules (Lumberg Automation™ Input/Output Network) 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 installation and maintenance. They additionally enable easy and quick extension.

### 4.1 About LioN-X

The LioN-X and LioN-Xlight variants convert standard input, standard output or IO-Link signals from sensors & actuators into an industrial Ethernet protocol (PROFINET, EtherNet/IP, EtherCAT®, Modbus TCP, CC-Link IE Field Basic) 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](http://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 variants 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 (PN, EIP, EC, MB) Security	8 x IO-Link Class A
935700002	0980 XSL 3912-121-007D-01F	LioN-X M12-60 mm, IO-Link Master Multiprotocol (PN, EIP, EC, MB, CC) Security	8 x IO-Link Class A
935710001	0980 XSL 3912-121-027D-01F	LioN-X M12-60 mm, IO-Link Master Multiprotocol (PN, EIP, EC, MB, CC) Security, HTTPS	8 x IO-Link Class A
935703001	0980 XSL 3913-121-007D-01F	LioN-X M12-60 mm, IO-Link Master Multiprotocol (PN, EIP, EC, MB, CC) Security	8 x IO-Link Class A/B Mixmodule
935711001	0980 XSL 3913-121-027D-01F	LioN-X M12-60 mm, IO-Link Master Multiprotocol (PN, EIP, EC, MB, CC) Security, HTTPS	8 x IO-Link Class A/B Mixmodule
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

Article number	Product designation	Description	I/O port functionality
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
935701003	0980 LSL 3211-121-0006-004	LioN-Xlight M12-60 mm, IO-Link Master EtherCAT®	8 x IO-Link Class A
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
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
935701005	0980 LSL 3411-121-0006-010	LioN-Xlight M12-60 mm, IO-Link Master CC-Link IE Field Basic	8 x IO-Link Class A
935702005	0980 LSL 3410-121-0006-010	LioN-Xlight M12-60 mm, IO-Link Master CC-Link IE Field Basic	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 Class A IO-Link ports

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

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

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

## LioN-X Class A/B IO-Link ports

Device variant	Port	Pin 1 U <sub>S</sub>	Pin 4 / Ch. A (C/Q)				Pin 2 / Ch. B (I/Q)		
0980 XSL 3x13...	<b>Info:</b>	—	4 x Class A 4 x Class B	Type 1	Supply by U <sub>S</sub> <sup>1)</sup>	Supply by U <sub>S</sub> <sup>2)</sup>	Type 1	Supply by U <sub>S</sub> <sup>1)</sup>	Supply by U <sub>Aux</sub>
	<b>X8:</b>	Out (4 A)	IOL (Class B)	DI	DO (0.5 A)	DO (2 A)	—	—	DO/Pwr (2 A)
	<b>X7:</b>	Out (4 A)	IOL (Class B)	DI	DO (0.5 A)	DO (2 A)	—	—	DO/Pwr (2 A)
	<b>X6:</b>	Out (4 A)	IOL (Class B)	DI	DO (0.5 A)	DO (2 A)	—	—	DO/Pwr (2 A)
	<b>X5:</b>	Out (4 A)	IOL (Class B)	DI	DO (0.5 A)	DO (2 A)	—	—	DO/Pwr (2 A)
	<b>X4:</b>	Out (4 A)	IOL (Class A)	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)	—
	<b>X3:</b>	Out (4 A)	IOL (Class A)	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)	—
	<b>X2:</b>	Out (4 A)	IOL (Class A)	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)	—
	<b>X1:</b>	Out (4 A)	IOL (Class A)	DI	DO (0.5 A)	DO (2 A)	DI	DO (2 A)	—

Table 5: Port configuration of 0980 XSL 3x13... variants

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

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



### LioN-Xlight Class A IO-Link ports

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

Table 6: Port configuration of 0980 LSL 3x11... variants

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

Table 7: Port configuration of 0980 LSL 3x10... variants

<sup>1)</sup> 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 EtherNet/IP product features

#### Data connection

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

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

#### Data transmission rates

Featuring a transmission rate of up to 10/100 MBit/s, the EtherNet/IP devices can handle both fast transmission of I/O data and transmission of larger volumes of data.

#### EtherNet/IP Adapter Device

The LioN-X and LioN-Xlight IO-Link Master variants support the EtherNet/IP protocol. This allows the transmission of time sensitive process data between network components in real-time communication.

#### ODVA CIP specification V3.27

The LioN-X and LioN-Xlight IO-Link Master variants comply with ODVA CIP specification V3.27.

#### Integrated switch

The integrated Ethernet switch has two EtherNet/IP ports and thus supports the establishment of a line or ring topology for the EtherNet/IP network.

#### DHCP/BOOTP

The supported Dynamic Host Configuration Protocol (DHCP) and the Bootstrap Protocol (BOOTP) provide mechanisms for automatic obtaining of an IP address from a server managing the devices.

**Device Level Ring**

The additionally implemented Device Level Ring (DLR) enables the design of a highly available network infrastructure of up to 50 DLR ring nodes. If a connection is interrupted, the LioN-X devices immediately switch to an alternative ring segment and thus ensure interruption-free operation. These DLR ring nodes are “beacon-based” according to the EtherNet/IP specification.

**SNMP**

The SNMPv1 protocol handles network component monitoring and communication between the master and device.

**Diagnostic data**

The devices support diagnosis flags and extended diagnostic data that can be appended to the I/O data.

**EDS-based configuration and parameterization of the I/O ports**

The EDS offers the option of configuring and parameterizing the I/O ports on the master devices.

## 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 device variant, the device has 4 IO-Link Class A ports, 4 IO-Link Class A ports and 4 IO-Link Class B ports, or 8 IO-Link Class A ports with an additional digital input and optional output (0980 XSL 3x13... variants) on pin 2 of the I/O port. For detailed information see chapter [I/O port overview](#) on page 23.



**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 the device series is the 5-pin M12 connector (Pin 5 not used at 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 EtherNet/IP via vendor specific IO-Link Device parameter object class and Read/Write ISDU services.

### 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 236.

## 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

Reading and writing of IO-Link Device parameters is supported. The system command Store parameters is needed after parameter writing, to take over the changed parameter into the IO-Link Master backup memory when enabled.

### HTTPS

LioN-X supports several security mechanisms (see also [Security features](#) on page 30). One of them is HTTPS (only applicable for device variants 0980 XSL 3912-121-027D-01F and 0980 XSL 3913-121-027D-01F), which allows encryption-based secure communication to access Web pages.

## 5.4 Security features

### Firmware signature

The official firmware update packages contain a signature which helps prevent 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 help protect the Web interface against unauthorized access. You can manage the users by groups with different access levels “Admin” or “Write”.

#### Default user settings:

User: admin

Password: private



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

## 5.5 Other features

### Interface protection

The devices have reverse polarity, short-circuit and overload protection for each interface.

For more details, see section [Port assignments](#) on page 41.

### Failsafe

The devices support a failsafe function. This allows you to define the behavior of every single channel configured as an output in the case of invalid PLC data (e.g. PLC in STOP) or of lost 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 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

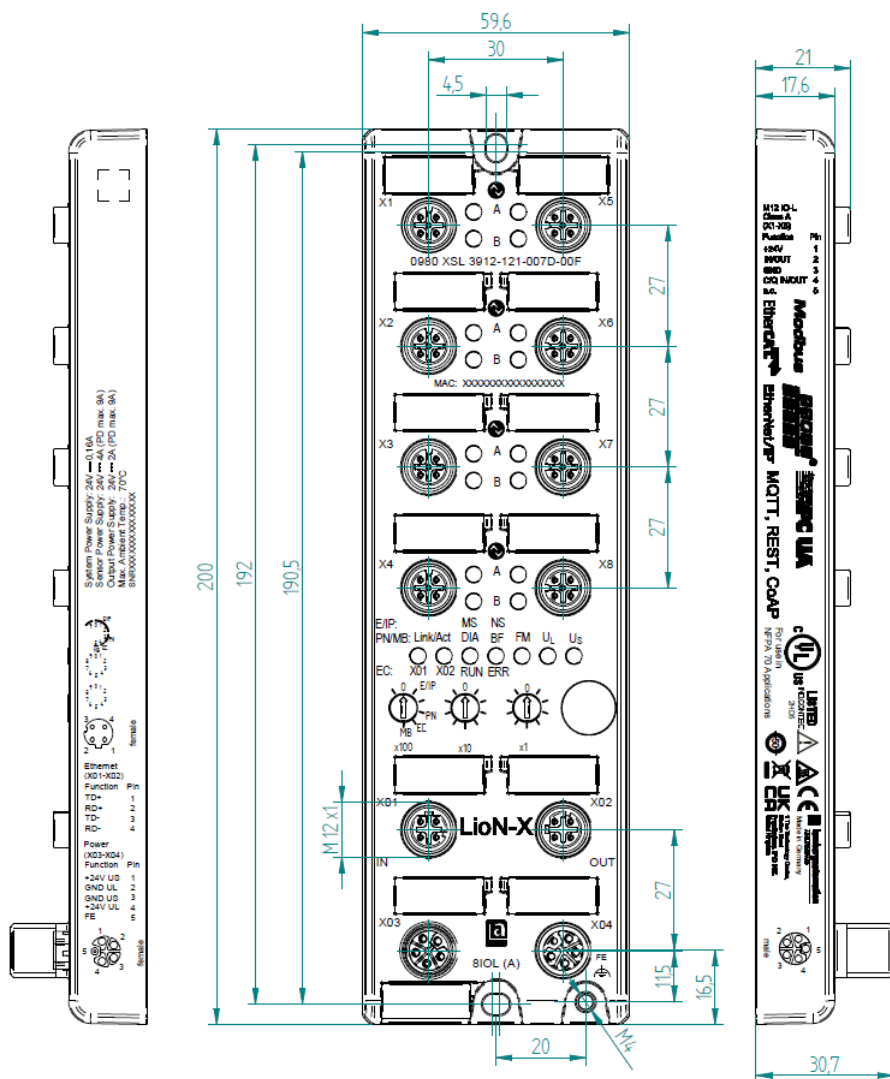


Figure 1: 0980 XSL 3912-121-007D-00F

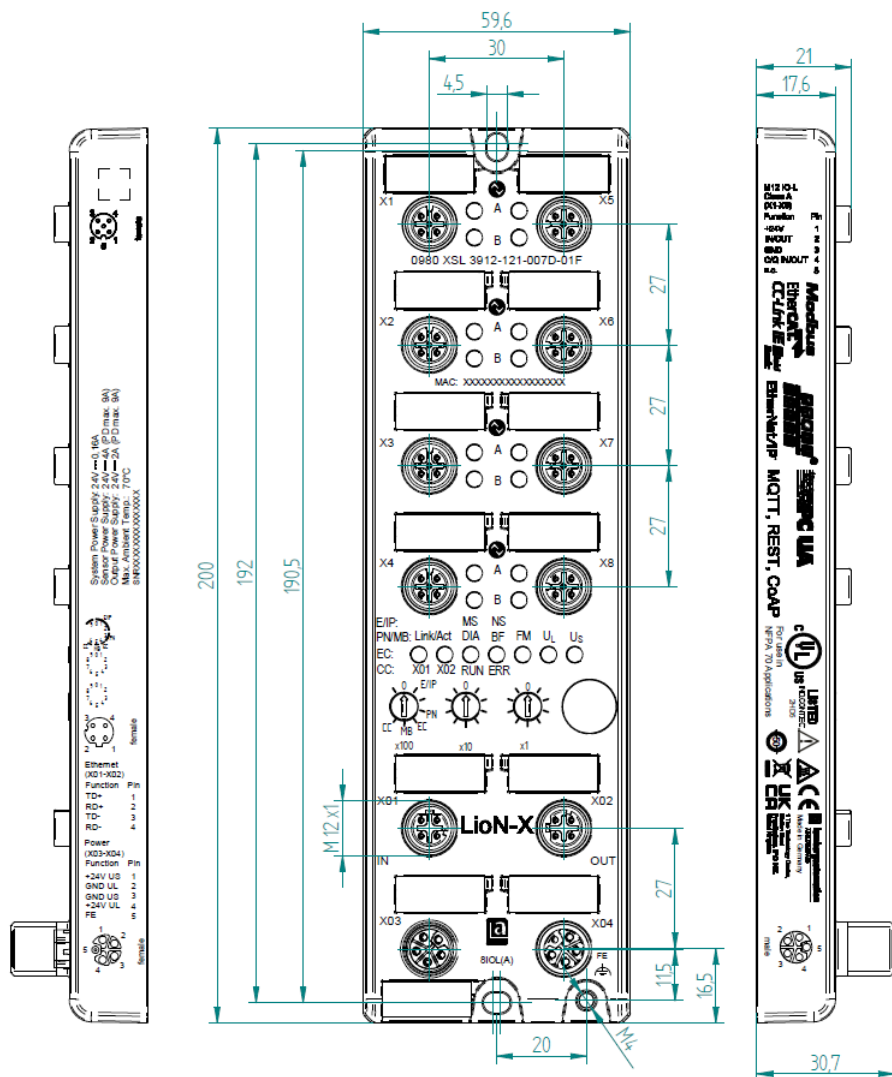


Figure 2: 0980 XSL 3912-121-007D-01F

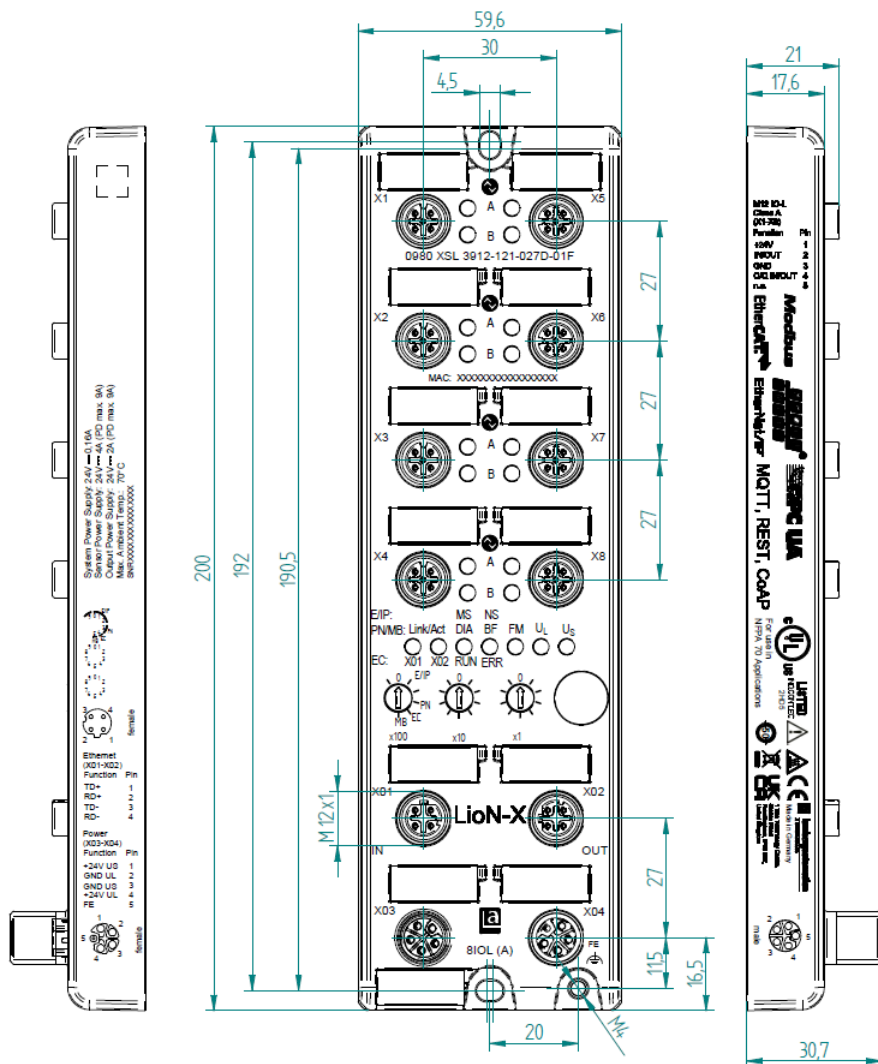


Figure 3: 0980 XSL 3912-121-027D-01F

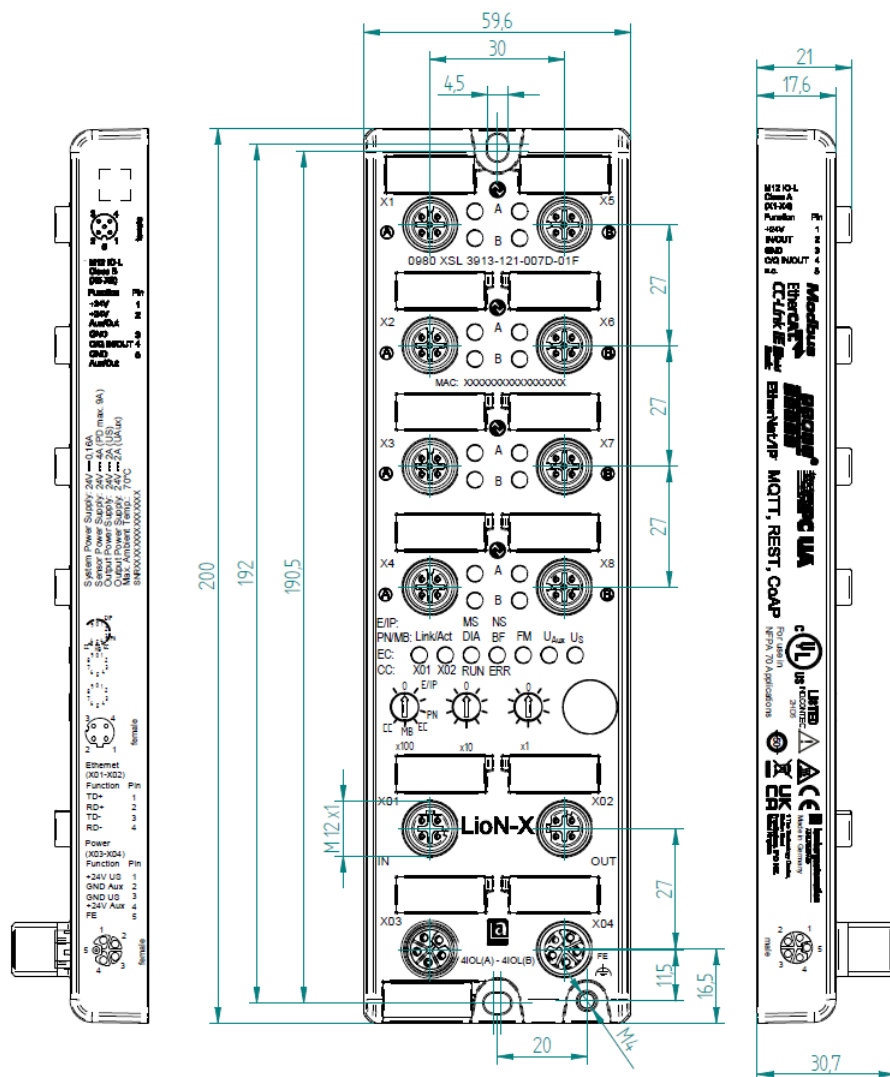


Figure 4: 0980 XSL 3913-121-007D-01F

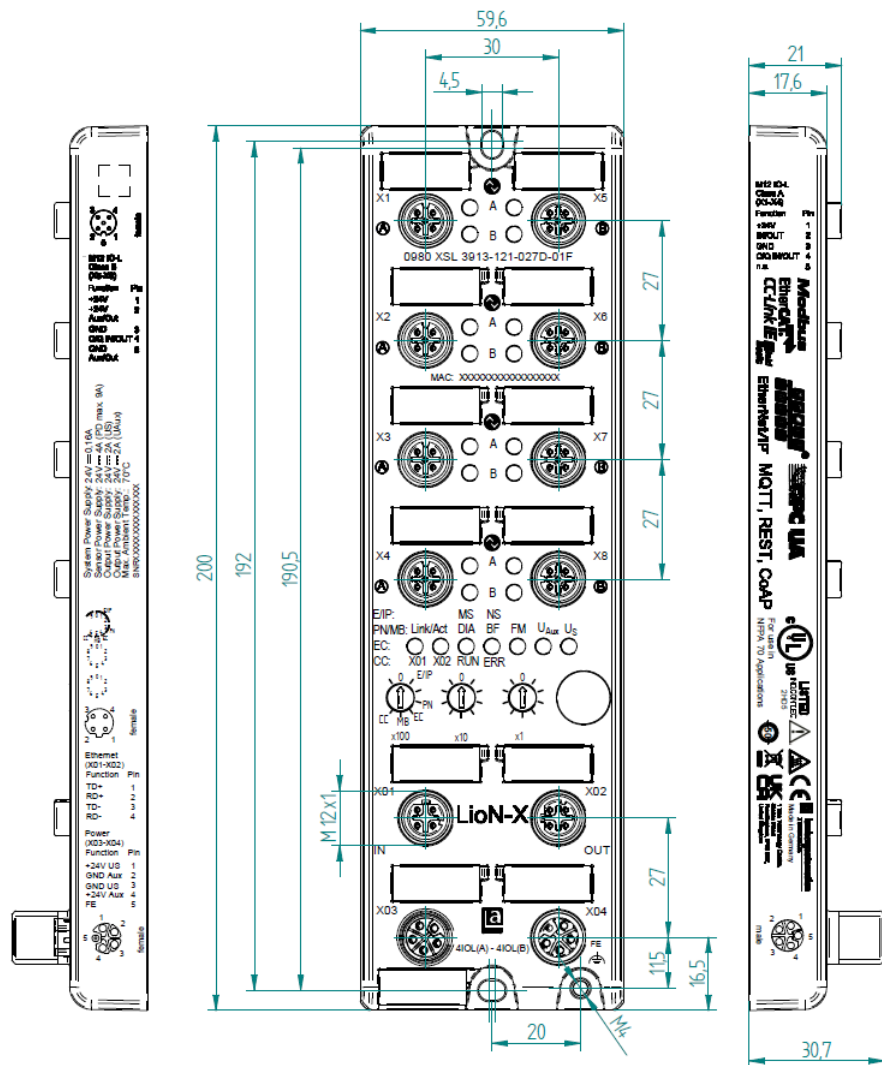


Figure 5: 0980 XSL 3913-121-027D-01F

### 6.2.2 Lion-Xlight variants with EtherNet/IP

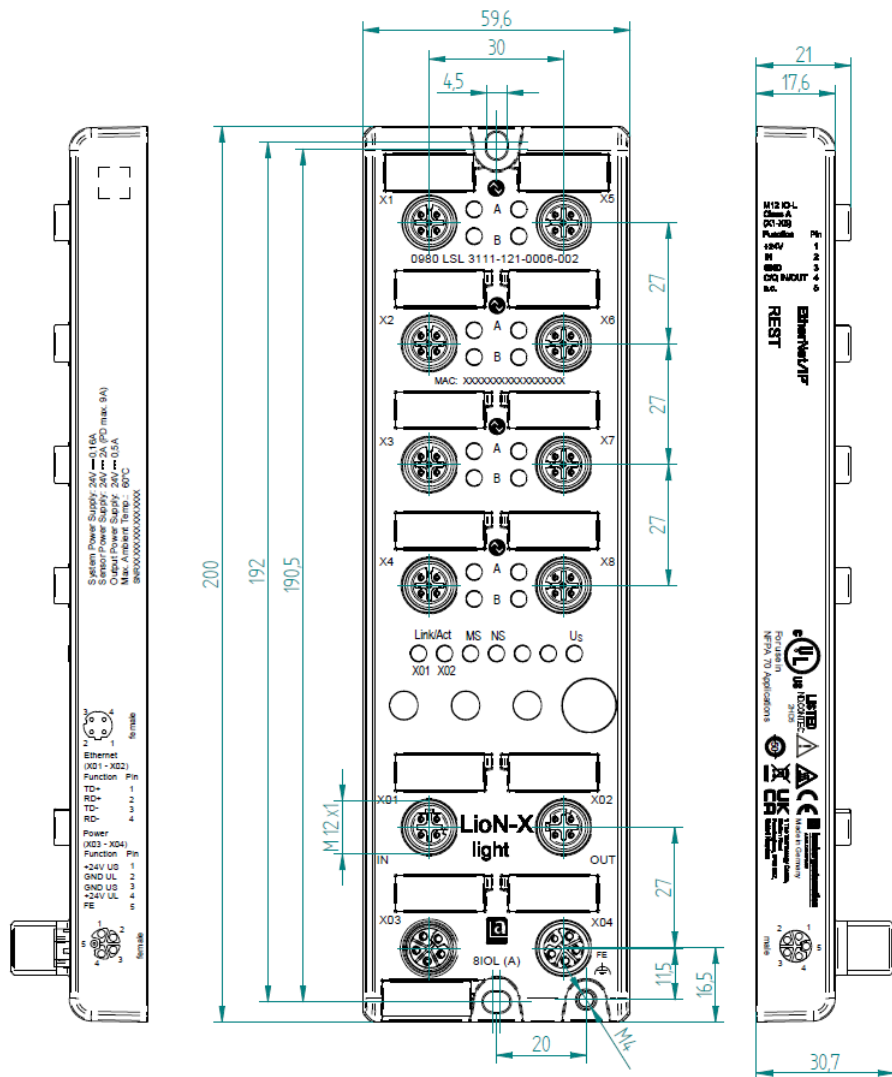


Figure 6: 0980 LSL 3111-121-0006-001

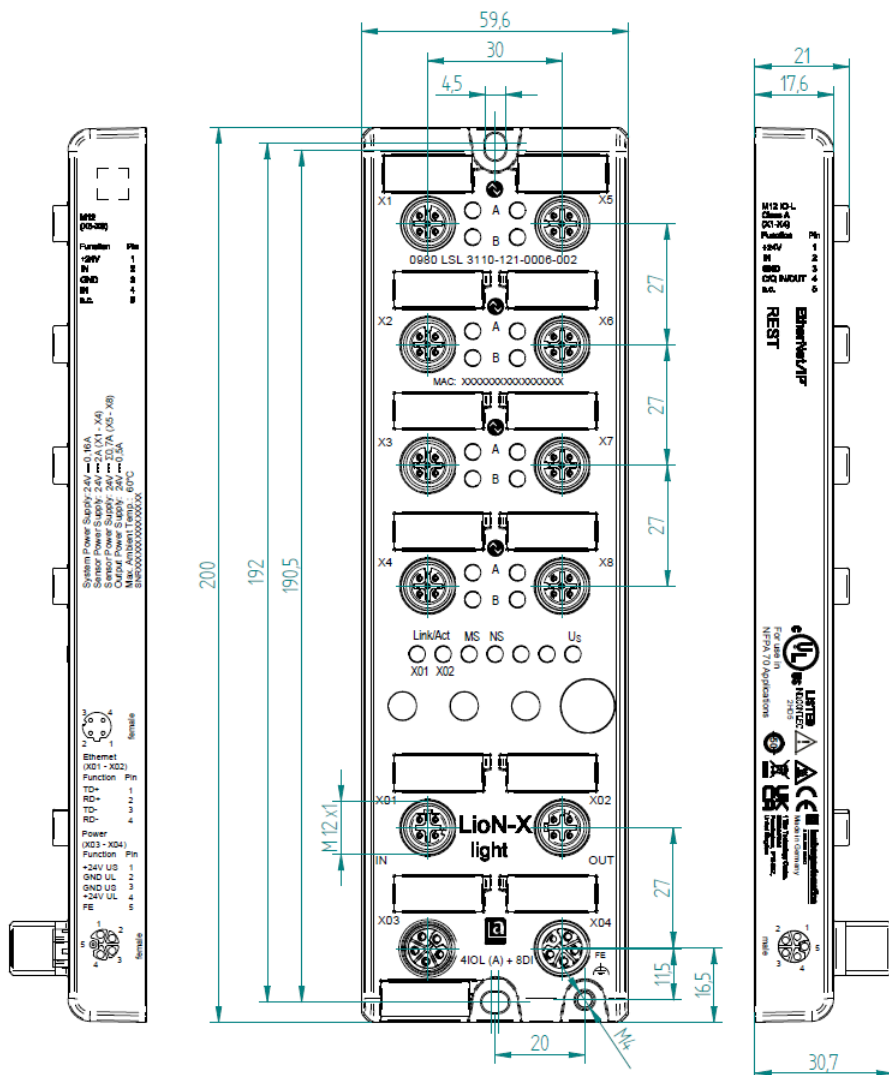


Figure 7: 0980 LSL 3110-121-0006-001

### 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 +125 °C (257 °F) for all LioN-X and LioN-Xlight variants.



**Warning:** Observe the following maximum output power for the sensor supply of Class A devices:

Max. 4.0 A per port; for **UL applications** max. 5 A 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.



**Warning:** Observe the following maximum output power for the sensor supply of Class A/B devices:

Max. 4.0 A per port; for **UL applications** max. 5.0 A from  $U_S$  power supply for every port pair X1/X2, X3/X4, X5/X6, X7/X8 and max. 5.0 A from  $U_{AUX}$  power supply in total for port group 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

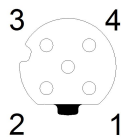


Figure 8: Schematic drawing, ports X01, X02

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 8: 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

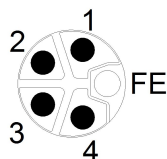


Figure 9: Schematic diagram of the M12 L-coding (connector X03 for Power In)

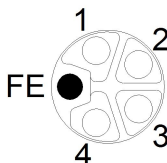


Figure 10: Schematic diagram of the M12 L-coding (socket X04 for Power Out)

### 6.3.2.1 IO-Link Master with Class A ports

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 $U_S$ <sup>1</sup>
	4	$U_L$ (+24 V)	Load supply ( <b>NOT</b> electrically isolated to $U_S$ internally in device)
	5	FE	Functional ground

Table 9: Power supply with M12-Power Class A



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

<sup>1</sup>  $U_L$  and  $U_S$  ground connected in device

6.3.2.2 IO-Link Master with Class A/B ports

Power supply	Pin	Signal	Function
Mixed IO-Link (Class A/B) I/O ports	1	$U_S$ (+24 V)	Sensor/system power supply
	2	$GND\_U_{AUX}$	Ground/reference potential $U_{AUX}$ (electrically <b>isolated</b> to $GND\_U_S$ internally in device)
	3	$GND\_U_S$	Ground/reference potential $U_S$
	4	$U_{AUX}$ (+24 V)	Auxiliary supply (electrically <b>isolated</b> to $U_S$ internally in device)
	5	FE	Functional ground

Table 10: Power supply with M12-Power Class A/B

**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 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

Color coding: black

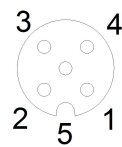


Figure 11: Schematic drawing I/O port as M12 socket IO-Link

**6.3.3.1 IO-Link ports (Class A and Class B)**

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 input or digital output
	3	GND	Ground/reference potential
	4	C/Q	Ch. A: IO-Link data communication, digital input or digital output
	5	n.c.	not connected
0980 XSL 3x13-121...	Pin	Signal	Function
IO-Link Class A, ports X1 ... X4	1	+24 V	power supply +24 V
	2	IN/OUT	Ch. B: Digital input or digital output
	3	GND	Ground/reference potential
	4	C/Q	Ch. A: IO-Link data communication, digital input or digital output
	5	n.c.	not connected
IO-Link Class B, ports X5 ... X8	1	+24 V	power supply +24 V
	2	+24 V AUX/OUT	Ch. B: Auxiliary power supply (electrically isolated with respect to the sensor/ system power supply $U_S$ ) or digital output
	3	GND	Ground/reference potential of +24 V
	4	C/Q	Ch. A: IO-Link data communication, digital input or digital output
	5	GND AUX	Ground/reference potential of +24 V AUX/ OUT
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 input
	3	GND	Ground/reference potential
	4	C/Q	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 input
	3	GND	Ground/reference potential
	4	C/Q	Ch. A: IO-Link data communication, digital input or digital output
	5	n.c.	not connected
Digital Input, ports X5 .. X8	1	+24 V	power supply +24 V
	2	IN	Ch. B: Digital input
	3	GND	Ground/reference potential
	4	IN	Ch. A: Digital input
	5	n.c.	not connected

Table 11: I/O ports as IO-Link Class A and Class B

Used signal names compared to the IO-Link specification conventions:

Pin	LioN-X	IO-Link specification	Comment
1	+24 V	L+	Supplied by $U_S$
2	IN/OUT	I/Q	
	+24 V AUX/OUT	2L	Supplied by $U_{AUX}$
3	GND	L-	
4	C/Q IN/OUT	C/Q	
5	GND AUX	2M	

## 7 Starting operation

### 7.1 EDS file

An EDS file describes the EtherNet/IP device and can be installed in the engineering tool for the configuration of the LioN-X device. Each of the LioN-X variants requires its own EDS file. The file can be downloaded from the product pages on our online catalog: [catalog.belden.com](https://catalog.belden.com)

On request, the EDS file is also sent to you by the support team.

The EDS files are grouped together in an archive file named **EDS-V3.27.1-BeldenDeutschland-LioN-X-yyyyymmdd.eds**.

**yyyyymmdd** stands for the date on which the file was issued.

Download this file and unpack it.

Install the EDS file for the respective device variant by using the hardware or network configuration tool of your controller manufacturer.

In Rockwell Automation Studio 5000®, install the files with the *EDS Hardware Installation Tool*.

The LioN-X and LioN-Xlight variants are then available in the hardware catalog as *Communications Adapter*.

### 7.2 MAC addresses

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

# 7.3 State on delivery

EtherNet/IP parameters in state on delivery or after a factory reset:

Network mode:	DHCP
Static IP address:	192.168.1.XXX (XXX = rotary switch position or last stored data)
Subnet mask:	255.255.255.0
Gateway address	0.0.0.0
Device designations:	0980 XSL 3912-121-007D-00F 0980 XSL 3912-121-007D-01F 0980 XSL 3912-121-027D-01F 0980 XSL 3913-121-007D-01F 0980 XSL 3913-121-027D-01F 0980 LSL 3111-121-0006-002 0980 LSL 3110-121-0006-002
Vendor code:	21
Product type:	12 (Communications Adapter)



## 7.4 Setting network parameters

There are multiple ways to configure the network parameters. By default, DHCP is enabled and the network parameters are requested by DHCP requests to a server. If you want to request the network parameters with BOOTP requests, you must activate the BOOTP function through the Web interface or the TCP/IP interface object (CIP Class ID 0xF5, attribute 3 (0x03)). It is also possible to set static network parameters via this CIP object.

### 7.4.1 IP address for LioN-X variants

The LioN-X multiprotocol variants support IP address configuration via the three rotary encoding switches on the front of the device (see chapter [Setting the rotary encoding switches](#)). The network parameters are also settable via the Web interface, the IIoT protocols or the LioN Management Suite.

### 7.4.2 IP address for LioN-Xlight variants

The LioN-Xlight variants cannot be configured via rotary encoding switches. If your network does not provide a DHCP server, you can set a static IP address via the NetIdent protocol in the LioN Management Suite.

## 7.5 Setting the rotary encoding switches

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 following LioN-X IO-Link Master variants additionally provide the protocol CC-Link IE Field Basic (CC):

- 0980 XSL 3912-121-007D-01F
- 0980 XSL 3912-121-027D-01F
- 0980 XSL 3913-121-007D-01F
- 0980 XSL 3913-121-027D-01F



### **Caution: Risk of device damage due to corrupt device memory**

Any interruption of the power supply to the device during and after protocol selection can lead to a corrupt device memory.

After selecting a protocol followed by a restart of the device, the new protocol is initialized. This can take up to 15 seconds. During this time the device is not usable and the LED indicators are out of function. When the protocol change is complete, the LED indicators return to normal operation and the device can be used again.

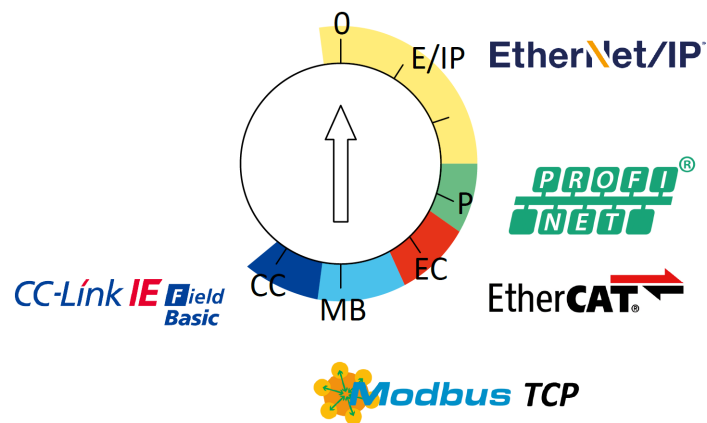
- Make sure that the power supply is maintained during the entire process.

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 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, Modbus TCP or CC-Link IE Field Basic.

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

Table 12: 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 54.

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

### 7.5.1 EtherNet/IP selection and IP configuration via rotary encoding switches

The EtherNet/IP protocol can be selected by the first rotary encoding switch (x100) with a value between 0 – 2.

Use all three rotary encoding switches on the front of the device to set the last octet of the static IP address. The first three octets of the IP address are set by default to 192.168.1.

Each rotary encoding switch in the EtherNet/IP setting is assigned to one decimal digit, so that you can configure a number between 0 – 299. During start-up, the position of the rotary encoding switches is typically read within one time cycle.

For example, the rotary encoding switch setting 2 (x100), 1 (x10) and 0 (x1) is interpreted by default as the IP address 192.168.1.210.

Rotary encoding switch setting	Function
000 (state on delivery, default setting)	On delivery, the DHCP function is enabled. The network parameters are requested by DHCP requests to a server. If you want to request the network parameters with BOOTP requests, you must activate the BOOTP function through the Web server or the TCP/IP interface object (CIP Class ID 0xF5, attribute 3 (0x03)). The network parameters are not saved automatically, but the integrated Web server can be used to save them.
000 (network parameters already saved)	The network parameters last saved are used (IP address, subnet mask, gateway address, DHCP on/off, BOOTP on/off).
001 ... 254	The last 3 digits of the saved or preset IP address are overwritten by the setting of the rotary encoding switches. DHCP or BOOTP are disabled if necessary, and the device will start up with a static IP address.
255 ... 298	The network parameters are requested through DHCP or BOOTP but are not saved.
299	The factory default setting of the IP address (192.168.001.001) is used.
979	The device performs a reset to the factory settings. The network parameters are also reset to the default values. Communication is not possible in this operation mode.

*Table 13: Setting options of the rotary encoding switches for EtherNet/IP*

### 7.5.2 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](#) again to select a new protocol.

For performing a factory reset via software configuration, see chapter [OPC UA configuration](#) on page 172 and the configuration section.

## 8 Configuration EtherNet/IP

The devices support *Implicit Messaging* and *Explicit Messaging* for the EtherNet/IP communication. I/O process data is transferred cyclically via the assembly object connection using *Implicit Messaging*.

Non-critical low priority data, configuration settings and diagnostic data can be exchanged via acyclic messages using *Explicit Messaging*. The exchange is done via EtherNet/IP and vendor specific object classes. For more details on object classes, see chapter [CIP object classes](#) on page 111.

### 8.1 Assembly types

The LioN-X devices support three different assembly types which are defined as follows:

Assembly ID	Assembly Name	Size	Payload
130	Output Connection Point Assembly	0 .. 260 Byte	Consuming Data Image
131	Input Connection Point Assembly	0 .. 446 Byte	Producing Data Image
145	Configuration Assembly	0 or 400 Byte	Module Configuration Data
146	Configuration Assembly	0 oder 300 Byte	Module Configuration (Omron)

The *Consuming Data Image* and the *Producing Data Image* have dynamic sizes which depend on the complete input and output data size of all connected IO-Link devices and on additional input status information. The general input and output process data sizes of each connection can be configured in the engineering tool. Each IO-Link Device process data size can be configured by the *Module Configuration Data*.

The contents of the *Consuming Data Image* and the *Producing Data Image* are specified in chapter [Process data assignment](#) on page 83.

The *Module Configuration Data* is defined in chapter [Configuration parameters](#) on page 59.

## 8.2 Connections

The LioN-X devices support two different connection types which are defined as follows:

Connection name	Connection type	Output connection point assembly	Output data size	Input connection point assembly	Input data size	Configuration assembly	Configuration data size
IO-Link (Exclusive Owner)	Exclusive Owner	130	0..260 Byte	131	0..446 Byte	145	0 or 400 Byte
IO-Link (Listen Only)	Listen Only	192	0	131	0..446 Byte	n/a	0 Byte
IO-Link Omron (Exclusive Owner)	Exclusive Owner	130	0..260 Byte	131	0..446 Byte	146	0 or 300 Byte
IO-Link Omron (Listen Only)	Listen Only	192	0	131	0..446 Byte	n/a	0 Byte

The dynamic data sizes depend on the complete input and output data size of all connected IO-Link devices and additional input status information. The general input and output process data sizes of each connection can be configured in the engineering tool. Each IO-Link device process data size can be configured by the *Module Configuration Data*.

Some engineering tools require the immediate configuration of the connection parameters. For the configuration use the parameters listed in the following chapters.



### 8.2.1 IO-Link parameters (Exclusive Owner)

Connection properties	
Connection name	IO-Link (Exclusive Owner)
Application type	Exclusive Owner
Trigger mode	Cyclic
RPI	min. 1 ms

Connection parameters (O->T)	
Real time transfer format	32 Bit Run/Idle Header
Connection type	POINT2POINT
Assembly ID	130
Data size	0..260 Byte
Data type	INT (2 Byte)

Connection parameters (T->O)	
Real time transfer format	Pure data and modeless
Connection type	MULTICAST, POINT2POINT
Assembly ID	131
Data size	0..446 Byte
Data type	INT (2 Byte)

### 8.2.2 IO-Link parameters (Listen Only)

Connection properties	
Connection name	IO-Link (Listen Only)
Application type	Listen Only
Trigger mode	Cyclic
RPI	min. 1 ms

Connection parameters (O->T)	
Real time transfer format	Heartbeat
Connection type	POINT2POINT
Assembly ID	192
Data size	0 Byte
Data type	INT (2 Byte)

Connection parameters (T->O)	
Real time transfer format	Pure data and modeless
Connection type	MULTICAST
Assembly ID	131
Data size	0..446 Byte
Data type	INT (2 Byte)

# 9 Configuration parameters

Parameters of the LioN-X device can be configured via the configuration assembly, CIP object classes, Web server or IloT protocols. A configuration assembly is sent when an *Exclusive Owner* connection is established. They are optional in this assembly. However, when sending, all existing parameters will be overwritten by this data. Therefore, the content of the configuration assembly has the highest valence.

To avoid parameter overwriting by CIP object classes, Web server or IloT protocols during operation, some lock parameters can be enabled in the PLC configuration respectively configuration assembly.

The following chapters represent different setting groups with its configuration parameters. They are ingredients of the configuration assembly and can be set via *Explicit Messaging* by the specified CIP object classes. The **default values** are highlighted.

# 9.1 General settings

Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA0, Instance 1
Force Mode Lock	1	SINT	<b>0: Disable</b> 1: Enable	Attribute 2
Web Interface Lock	2	SINT	<b>0: Disable</b> 1: Enable	Attribute 3
Digital Output Control	3	SINT	<b>0: DO Channel Control</b> 1: IO-Link Control	Attribute 4
Report $U_L/U_{Aux}$ Supply Voltage Fault	4	SINT	0: Disable <b>1: Enable</b>	Attribute 5
Report DO Fault without $U_L/U_{Aux}$	5	SINT	0: Disable <b>1: Enable</b>	Attribute 6
CIP object configuration lock	24	SINT	<b>0: Disable</b> 1: Enable	Attribute 25
External configuration lock	25	SINT	0: Disable <b>1: Enable</b>	Attribute 26
IO Mapping Mode	31	SINT	<b>0: Default Assignment</b> 1: Byte Swap 2: LSB Ch.A - MSB Ch.B 3: LSB Ch.B - MSB Ch.A 4: Free I/O Mapping	Attribute 32

### 9.1.1 Force mode lock

The input and output process data can be forced via different interfaces (e.g. Web interface, REST, OPC UA, MQTT). The support of interfaces depends on the available software features. If the *Force mode lock* is enabled, it is no longer possible to force input and output process data through these interfaces.



**Danger:** Risk of physical injury or death! Unattended forcing can lead to unexpected signals and uncontrolled machine movements.

### 9.1.2 Web interface lock

The Web interface access can be configured. If *Web interface lock* is enabled, the Web pages are no longer reachable.

### 9.1.3 Digital Output Control

A digital output can only have one control source. With the parameter *Digital Output Control*, you can configure the DO Channel Control (first two bytes of the output data) or the IO-Link Output Data (first byte of each IO-Link device output data) as the control source.

### 9.1.4 Report $U_L/U_{AUX}$ supply voltage fault

During commissioning, it is possible that no power supply is connected to the  $U_L/U_{AUX}$  pins. Therefore it can be helpful to suppress and disable the  *$U_L/U_{AUX}$  supply voltage fault* diagnosis.

### 9.1.5 Report DO Fault without $U_L/U_{Aux}$

With this parameter you suppress the actuator diagnosis message that is sent if no  $U_L/U_{Aux}$  supply is connected while the output data of a digital channel is controlled.

### 9.1.6 CIP object configuration lock

When there is no *Exclusive Owner* connection established, all configuration parameters can be set by vendor specific CIP object classes. To exclude parameter changes the setting function of these objects can be blocked.

When the *CIP object* configuration lock is enabled, the vendor specific configuration parameters cannot be set via the CIP services. This relates also to the *CIP object* configuration lock itself. A reset of this parameter can be done by a configuration assembly when an *Exclusive Owner* connection is established.

### 9.1.7 External configuration lock

Configuration parameters can be set via different alternative interfaces (e.g. Web interface, REST, OPC UA, MQTT). An external configuration can only be done, if no cyclic PLC connection is active. Every new PLC configuration overwrites the external configuration settings.

### 9.1.8 IO Mapping Mode

The LioN-X devices support 5 different I/O mapping modes for the *Digital Output Channel Control* and the *Input Channel Status*. Modes 0 to 3 are pre-defined bit mappings. Mode 4 is a free user defined mapping which can be used in conjunction with the I/O mapping of channel 1 .. 16 in the channel settings.

#### Default Assignment (Mode 0):

DO Ch. Control / DI Ch. Status	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0 (LSB)	X4B	X4A	X3B	X3A	X2B	X2A	X1B	X1A
Byte 1 (MSB)	X8B	X8A	X7B	X7A	X6B	X6A	X5B	X5A

#### Byte Swap (Mode 1):

DO Ch. Control / DI Ch. Status	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0 (LSB)	X8B	X8A	X7B	X7A	X6B	X6A	X5B	X5A
Byte 1 (MSB)	X4B	X4A	X3B	X3A	X2B	X2A	X1B	X1A

**LSB Ch.A - MSB Ch.B (Mode 2):**

DO Ch. Control / DI Ch. Status	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0 (LSB)	X8A	X7A	X6A	X5A	X4A	X3A	X2A	X1A
Byte 1 (MSB)	X8B	X7B	X6B	X5B	X4B	X3B	X2B	X1B

**LSB Ch.B - MSB Ch.A (Mode 3):**

DO Ch. Control / DI Ch. Status	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0 (LSB)	X8B	X7B	X6B	X5B	X4B	X3B	X2B	X1B
Byte 1 (MSB)	X8A	X7A	X6A	X5A	X4A	X3A	X2A	X1A

**Free IO Mapping (Mode 4):**

IO Mapping Channel 1 .. 16 is used (see chapter [Channel settings](#) on page 64).

## 9.2 Channel settings

Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA1, Instance 1 .. 16
IO Mapping (Ch1 .. 16)	32	SINT[16]	<b>0 .. 15:</b> Bit number of 16 channel process data 16: Inactive	Attribute 1
DO Surveillance Timeout (Ch1 .. 16)	48	INT[16]	0 .. 255 ( <b>80</b> )	Attribute 2
DO Failsafe (Ch1 .. 16)	80	SINT[16]	<b>0: Set Low</b> 1: Set High 2: Hold Last	Attribute 3
DO Restart Mode (Ch1 .. 16)	96	SINT[16]	0: Disable <b>1: Enable</b>	Attribute 4
DO Switch Mode (Not available for LioN-Xlight IO-Link Master variants)	112	SINT[16]	0: Push-Pull (U <sub>S</sub> , 0.5 A) 1: High-Side (U <sub>L</sub> , 0.5 A) 2: High-Side (U <sub>L</sub> , 1.0 A) 3: High-Side (U <sub>L</sub> , 1.5 A) 4: High-Side (U <sub>L</sub> , 2.0 A) <b>5: High-Side (U<sub>L</sub>, 2.0 A max)</b>	Attribute 5
DI Logic (Ch1 .. 16)	128	SINT[16]	<b>0: Normally Open</b> 1: Normally Close	Attribute 6
DI Filter (Ch1 .. 16)	144	SINT[16]	0: Disabled 1: 1 ms 2: 2 ms <b>3: 3 ms</b> 4: 6 ms 5: 10 ms 6: 15 ms	Attribute 7



Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA1, Instance 1 .. 16
Channel Mode (Ch1 .. 16)	192	SINT[16]	0: Inactive 1: Digital Output 2: Digital Input 3: IO-Link 4: Auxiliary Power  <b>The supported Channel Mode and the default value depend on the device variant.</b>	Attribute 10

Assignment of channels:

<b>Channel 1</b>	Port X1.ChA	CIP object instance 1
<b>Channel 2</b>	Port X1.ChB	CIP object instance 2
<b>[...]</b>	<b>[...]</b>	<b>[...]</b>
<b>Channel 15</b>	Port X8.ChA	CIP object instance 15
<b>Channel 16</b>	Port X8.ChB	CIP object instance 16

### 9.2.1 IO Mapping (Ch1 .. 16)

These configuration parameters can be used to set a user defined IO mapping. It is valid for the input and output data direction. Duplicated assignment are not allowed. In case of an inconsistent mapping, the complete assembly configuration is rejected with an error code.

To use these parameters, it is required to configure the IO mapping mode of the *General settings* to *Free IO Mapping (Mode 4)*. The default value for each parameter is its own channel number.

### 9.2.2 DO Surveillance Timeout (Ch1 .. 16)

The digital output channels are monitored during runtime. The error states are detected and reported as a diagnosis. To avoid error states during the switching of output channels, the surveillance timeout can be configured as a delay with deactivated monitoring.

The delay time begins with a rising edge of the output control bit. After delay time has elapsed, the output is monitored and error states are reported by diagnosis. When the channel is permanently switched on or off, the typical filter value (not changeable) is 5 ms.

### 9.2.3 DO Failsafe (Ch1 .. 16)

The LioN-X devices support a failsafe function for the channels used as digital outputs. In case of an internal device error, the PLC is in STOP state and cannot provide valid process data. The connection is interrupted or the communication is lost. The outputs are controlled according to the configured failsafe values.

#### Set Low:

If failsafe is active, the physical output pin of the channel is set to low ("0").

#### Set High:

If failsafe is active, the physical output pin of the channel is set to high ("1").

#### Hold Last:

If failsafe is active, the physical output pin of the channel holds the last valid process data state ("0" or "1").

### 9.2.4 DO Restart Mode (Ch1 .. 16)

In case of a short circuit or overload at an output channel, a diagnosis is reported and the output is switched to "off".

If the *DO Restart Mode* for this channel is enabled, the output will automatically be turned on again after a fix time delay for checking if the overload or short circuit condition is still active. When it is active, the channel is switched off again.

If the *DO Restart Mode* is disabled, the output channel is not automatically turned on again. It can be turned on after a logical reset of the process output data of the channel.

### 9.2.5 DO Switch Mode (Ch1 .. 16)

Only applicable for the following device variants:

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

With this parameter you can configure the current limitations for the digital outputs by selecting a DO Switch Mode. You can choose between two different output switch modes:

- ▶ Push-Pull ( $U_S$ , 0.5 A):

If a channel is set to "Push-Pull", the output will be switched active to high or low. In low state, the output can be a current sink. The digital output is supplied by  $U_S$  with a maximum current of 0.5 A. This option is not available for the channel B of any port.

► High-Side ( $U_L$ , 0.5 A..2.0 A max):

If a channel is set to "High-Side", the output will be switched active to high but not to low. In low state, the output has a high impedance. The digital output is supplied by  $U_L$  or  $U_{Aux}$ , depending on the device variant, and has a selectable current limit. This means that the actuator channel error diagnosis is reported when this limit is exceeded. If you set the level to 2.0 A Max., the current limitation is not active and the maximum output current is available.

Refer to chapter [I/O port overview](#) on page 23 to get the available voltage supply for the digital outputs of every LioN-X variant.

### 9.2.6 DI Logic (Ch1 .. 16)

The logical state of an input channel can be configured via these parameters. If a channel is set to "Normally Open", a low signal ("0") is transferred to the process input data (e.g. if a non-damped sensor has an open switching output).

If a channel is set to "Normally Close", a high signal ("0") is transferred to the process input data (e.g. if a non-damped sensor has a closed switching output).

The channel LED shows, independent of these settings, the physical input state of the port pin.

### 9.2.7 DI Filter (Ch1 .. 16)

A filter time for every digital input channel can be configured by these parameters. When there is no need for a filter it can be disabled.

### 9.2.8 Channel Mode (Ch1 .. 16)

The operation mode of every channel can be configured by these parameters. The usability of this setting depends on the hardware variant and can be figured out in the description (e.g. for a 8 IO-Link Class A Master, an IO-Link mode can only be configured for channel A and not for channel B).

#### Inactive:

This mode should be selected when the channel is not in use.



**Attention:** If channel A of a port is set to inactive, the corresponding channel B is also set to inactive regardless of its configuration. In this case, the entire port is therefore deactivated.

### Digital Output:

In this mode, the channel operates as digital output. The channel can be controlled by the *Digital Output Channel Control* (first two bytes of the output data) or by the *IO-Link Output Data* (first byte of each IO-Link Device output data) of the cyclic process data. This depends on the *Digital Output Control* parameter of the *General Settings*.

### Digital Input:

In this mode, the channel operates as digital input. The channel state can be seen in the *Digital Input Channel Status* of the cyclic process data.

### IO-Link:

In this mode, the channel tries to establish a communication with an IO-Link Device. IO-Link process data can be exchanged via a communication link between the IO-Link Master and the IO-Link Device. The size of the IO-Link input and output data as well as the port mode depend on the IO-Link port settings.



**Attention:** Not all channels support this configuration.

**Auxiliary Power:**

IO-Link Master variants with Class B ports provide an auxiliary voltage output on channel B. When *Auxiliary Power* is configured, the voltage output cannot be controlled for this channel and is fed from the  $U_{AUX}$  supply input. IO-Link Class A ports do not support this configuration.



**Attention:** Not all channels support this configuration.

## 9.3 IO-Link diagnosis settings

Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA2, Instance 1
IO-Link Master Diagnosis	208	SINT	0: Disable 1: <b>Enable</b>	Attribute 1
IO-Link Device Error	209	SINT	0: Disable 1: <b>Enable</b>	Attribute 2
IO-Link Device Warning	210	SINT	0: Disable 1: <b>Enable</b>	Attribute 3
IO-Link Device Notification	211	SINT	0: Disable 1: <b>Enable</b>	Attribute 4
IO-Link Device Diagnosis Port 1 .. 8	212 .. 219	SINT[8]	0: Disable 1: <b>Enable</b>	Attribute 5 .. 12

### 9.3.1 IO-Link Master Diagnosis

If this parameter is enabled, the *IO-Link Master Diagnosis* is transferred to the IO-Link diagnoses of the input process data. If configured, additional diagnoses and information are transferred in the *IO-Link Extended Status* and in the *IO-Link events*.

If this parameter is disabled, no *IO-Link Master Diagnosis* is reported.

### 9.3.2 IO-Link Device Error

If this parameter is enabled, the *IO-Link Device Errors* are transferred in the IO-Link diagnoses of the input process data. If configured, additional diagnoses and information are transferred in the *IO-Link Extended Status* and the *IO-Link events*.

If this parameter is disabled, no *IO-Link Device Error* is reported.

### 9.3.3 IO-Link Device Warning

If this parameter is enabled, the *IO-Link Device Warnings* are transferred in the IO-Link diagnoses of the input process data. If configured, additional

diagnoses and information are transferred in the *IO-Link Extended Status* and the *IO-Link events*.

If this parameter is disabled, no *IO-Link Device Warning* is reported.

### 9.3.4 IO-Link Device Notification

If this parameter is enabled, the *IO-Link Device Notifications* are transferred in the IO-Link diagnoses of the input process data. If configured, additional diagnoses and information are transferred in the *IO-Link Extended Status* and the *IO-Link events*.

If this parameter is disabled, no *IO-Link Device Notification* is reported.

### 9.3.5 IO-Link Device Diagnosis Port 1 .. 8

If this parameter is enabled for an IO-Link port, the respective diagnoses are transferred in the IO-Link diagnoses of the input process data. If configured, additional diagnoses and information are transferred in the *IO-Link Extended Status* and the *IO-Link events*.

If this parameter is disabled for an IO-Link port, no respective diagnosis is reported.



## 9.4 IO-Link Port 1 .. 8 settings

Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA3, Instance 1 .. 8
Output Data Size	224, 246, 268, 290, 312, 334, 356, 378	SINT	0: No data 1: 2 Byte 2: 4 Byte 3: 8 Byte 4: 16 Byte <b>5: 32 Byte</b>	Attribute 1
Input Data Size	225, 247, 269, 291, 313, 335, 357, 379	SINT	0: No data 1: 2 Byte 2: 4 Byte 3: 8 Byte 4: 16 Byte <b>5: 32 Byte</b>	Attribute 2
Input Data Extension	226, 248, 270, 292, 314, 336, 358, 380	SINT	0: No Data 1: Extended Status 2: Events <b>3: Extended Status + Events</b>	Attribute 3
Output Data Swapping Mode	227, 249, 271, 293, 315, 337, 359, 381	SINT	<b>0: Raw IO-Link Data</b> 1 .. 16: 1 .. 16 WORD 17 .. 24: 1 .. 8 DWORD	Attribute 4
Output Data Swapping Offset	228, 250, 272, 294, 316, 338, 360, 382	SINT	0 .. 30 Byte (" <b>0</b> ")	Attribute 5
Input Data Swapping Mode	229, 251, 273, 295, 317, 339, 361, 383	SINT	<b>0: Raw IO-Link Data</b> 1 .. 16: 1 .. 16 WORD 17 .. 24: 1 .. 8 DWORD	Attribute 6
Input Data Swapping Offset	230, 252, 274, 296, 318, 340, 362, 384	SINT	0 .. 30 Byte (" <b>0</b> ")	Attribute 7

Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA3, Instance 1 .. 8
IOL Failsafe	231, 253, 275, 297, 319, 341, 363, 385	SINT	<b>0: Set Low</b> 1: Set High 2: Hold Last 3: Replacement Value (transferred via IO-Link Failsafe Parameter Object) 4: IO-Link Master Command	Attribute 8
Port Mode	232, 254, 276, 298, 320, 342, 364, 386	SINT	0: Deactivated 1: Manual (with validation and backup config) <b>2: Autostart (no validation and backup config)</b>	Attribute 9
Validation and Backup	233, 255, 277, 299, 321, 343, 365, 387	SINT	<b>0: No device check and clear (no data storage)</b> 1: Type compatible V1.0 device (no data storage) 2: Type compatible V1.1 device (no data storage) 3: Type compatible V1.1 device with Backup + Restore (download + upload) 4 Type compatible V1.1 device with Restore (download master to device)	Attribute 10
Vendor ID	234, 256, 278, 300, 322, 344, 366, 388	DINT	0 .. 65535 (" <b>0</b> ")	Attribute 11
Device ID	238, 260, 282, 304, 326, 348, 370, 392	DINT	0 .. 16777215 (" <b>0</b> ")	Attribute 12

Configuration parameter	Byte offset config. assembly	Data type	Valid values	CIP object class 0xA3, Instance 1 .. 8
Cycle Time	242, 264, 286, 308, 330, 352, 374, 396	SINT	<b>0: As fast as possible</b> 1: 1.6 ms 2: 3.2 ms 3: 4.8 ms 4: 8.0 ms 5: 20.8 ms 6: 40.0 ms 7: 80.0 ms 8: 120.0 ms	Attribute 13

Assignment of the IO-Link ports:

IO-Link port 1	Port X1.ChA	CIP object instance 1
[...]	[...]	[...]
IO-Link port 8	Port X8.ChA	CIP object instance 8

The number of IO-Link ports depends on the IO-Link Master variant. IO-Link Masters with less than 8 IO-Link ports only provide configuration parameters for their own count. Unused configuration data bytes are sent as "zero bytes" inside the configuration assembly.

Configuration parameters of an IO-Link port are only taken into account by the application when the corresponding Channel Mode of the Channel Settings is set to *IO-Link*.

### 9.4.1 Output Data Size

The *Output Data Size* of the respective IO-Link device can be configured by this parameter. There can be up to 32 Bytes of IO-Link output data per port.

The *Output Data Size* of every IO-Link device has influence on the total *Output Data Size* of the connection. It has to be taken into account that all IO-Link output data fits into the total size.

This parameter is only settable when no connection is active.

### 9.4.2 Input Data Size

The *Input Data Size* of the respective IO-Link device can be configured by this parameter. There can be up to 32 Bytes of IO-Link input data.

The *Input Data Size* of every IO-Link device has influence on the total *Input Data Size* of the connection. It has to be taken into account that all IO-Link input data fits into the total size.

This parameter is only settable when no connection is active.

### 9.4.3 Input Data Extension

The *Input Data Extension* can be selected to extend each IO-Link input data with extended status information and/or IO-Link events.

The *Input Data Extension* of every IO-Link device has influence on the total input data size of the connection. It has to be taken into account that all IO-Link output data including the extension fits into the total size.

This parameter is only settable when no connection is active.

### 9.4.4 Output Data Swapping Mode

The byte order of IO-Link is big endian which is not compatible to EtherNet/IP's little endian format. For setting output data in the correct format, the parameters *Output Data Swapping Mode* and *Output Data Swapping Offset* support the user. There can be selected up to 16 words or up to 8 double words for converting the output data.

#### Raw IO-Link Data:

No byte swap

#### Data type WORD:

Data byte order: Byte 0, Byte 1

Order after Swap: Byte 1, Byte 0

#### Data type DWORD:

Data byte order: Byte 0, Byte 1, Byte 2, Byte 3

Order after Swap: Byte 3, Byte 2, Byte 1, Byte 0

### 9.4.5 Output Data Swapping Offset

The *Output Data Swapping Offset* describes the start point in the process data for using the configured *Output Data Swapping Mode*. Both parameters are dependent on the configured output data size.

### 9.4.6 Input Data Swapping Mode

The byte order of IO-Link is big endian which is not compatible to EtherNet/IP's little endian format. For receiving input data in the correct format, the parameters *Input Data Swapping Mode* and *Input Data Swapping Offset* support the user. There can be selected up to 16 words or up to 8 double words for converting the input data.

#### Raw IO-Link Data:

No byte swap

#### Data type WORD:

Data byte order: Byte 0, Byte 1

Order after Swap: Byte 1, Byte 0

#### Data type DWORD:

Data byte order: Byte 0, Byte 1, Byte 2, Byte 3

Order after Swap: Byte 3, Byte 2, Byte 1, Byte 0

### 9.4.7 Input Data Swapping Offset

The *Input Data Swapping Offset* describes the start point in the process data for using the configured *Input Data Swapping Mode*. Both parameters are dependent on the configured input data size and the optional input data extension.

### 9.4.8 IOL Failsafe

The LioN-X devices support a failsafe function for the output data of the IO-Link channels. In case of an internal device error, the PLC is in STOP state and cannot provide valid process data, the connection is interrupted or the communication is lost: The output data of the IO-Link channels is controlled by the configured failsafe values.

**Set Low:**

If failsafe is active, all bits of the IO-Link output data are set to low ("0").

**Set High:**

If failsafe is active, all bits of the IO-Link output data are set to high ("1").

**Hold Last:**

If failsafe is active, all bits of the IO-Link output data are hold the last valid process data state ("0" or "1").

**Replacement Value:**

A replacement value can be set via the *IO-Link Failsafe* parameter object for every IO-Link device. If failsafe is active, these replacement values are transmitted to the IO-Link device. The current configured IO-Link output data size must be considered. Take into account that in the case of an error the replacement values are sent instead of the output process data so that a configured *Swapping Mode* has influence on the byte order.

**IO-Link Master Command:**

If failsafe is active, an IO-Link-specific mechanism for valid/invalid output process data is used and the IO-Link device determines the behavior itself.

**9.4.9 Port Mode**

The *Port Mode* describes how the IO-Link master handles the presence of an IO-Link device at the port.

**Deactivated:**

The IO-Link port is deactivated but can be configured for later use. No diagnostics are generated if the IO-Link device is not connected.

**IO-Link Autostart:**

The IO-Link port is activated and no explicit port configuration is needed. Configurations such as *Validation and Backup* (Inspection Level), *Vendor ID*, *Device ID* and *Cycle Time* are not required.

**IO-Link Manual:**

The IO-Link port is activated and explicit port configuration can be done for the parameters *Validation and Backup* (Inspection Level), *Vendor ID*, *Device ID* and *Cycle Time*.

**9.4.10 Validation and Backup**

With this parameter, the user can set the behavior of the IO-Link ports regarding the type compatibility and data storage mechanism of the connected IO-Link Device.

The precondition for using *Validation and Backup* is that you configure the *Port Mode* to "IO-Link Manual".

The IO-Link Master has a backup memory which can be used for storing the device parameters and for restoring them on the device. This backup memory can be deleted by the following events:

- ▶ IO-Link Master factory reset
- ▶ *Channel Mode* reconfiguration, e.g. from "Digital-Input" to "IO-Link"
- ▶ *Validation and Backup* reconfiguration, e.g. from "No device check" to "Type compatible V1.1 device with Backup & Restore"

For further information refer to the 'IO-Link Interface and System Specification' version 1.1.3 which can be downloaded from <https://io-link.com/>

**No device check (no data storage):**

No check of connected Vendor ID or Device ID and no "Backup and Restore" support of the IO-Link Master parameter server.

**Type compatible V1.0 device (no data storage):**

Type compatible according IO-Link specification V1.0 which includes validation of Vendor ID and Device ID. The IO-Link specification V1.0 does not support IO-Link Master parameter server.

**Type compatible V1.1 device (no data storage):**

Type compatible according IO-Link specification V1.1 which includes validation of Vendor ID and Device ID. "Backup and Restore" is disabled.

## Type compatible V1.1 device with Backup + Restore

### (upload + download):

Type compatible according IO-Link specification V1.1 which includes validation of Vendor ID and Device ID. "Backup and Restore" is enabled.

Pay attention to the following explanations regarding *Backup and Restore* conditions:

#### ► Backup (Device to Master):

A Backup (upload from IOL-Device to IOL-Master) is performed when an IO-Link Device is connected and the Master does not have any valid parameter data. The read parameter data are permanently stored on the IO-Link Master.

An upload will also be performed, when the IO-Link Device has set the DS\_UPLOAD\_FLAG (Data Storage Upload Flag). This IOL-Device flag can be set in two ways:

- Parameters written to IOL-Device in *Block Parameter* mode: An IO-Link Device sets the DS\_UPLOAD\_FLAG self-dependent, if the parameters were written in block parameter mode to the IO-Link Device with the last system command ParamDownloadStore (e.g. by a third party USB IO-Link Master for commissioning).
- Parameters written to IOL-Device in *Single Parameter* mode: If single parameter data is changed on the IOL-Device during runtime, the stored device parameters on the IOL-Master can be updated using the ParamDownloadStore (index 0x0002, subindex 0x00, value 0x05) command. This command sets the DS\_UPLOAD\_REQ flag on the IOL-Device and thus the IO-Link Master executes an upload procedure from the IO-Link Device.

#### ► Restore (Master to Device):

A Restore (download from IOL-Master to IOL-Device) is performed when an IO-Link Device is connected and the IO-Link Master has valid parameter data stored which are usable for the IOL-Device and not equal compared to the device parameters.

The restore procedure can be blocked by the IO-Link Device via the *Device Access Locks* parameter when supported by the IO-Link Device (Index 0x000C, refer to vendor specific IO-Link Device documentation).



**Type compatible V1.1 device with Restore (download Master to Device):**

Type compatible according IO-Link specification V1.1 which includes validation of Vendor ID and Device ID. Only "Restore" is enabled.

Pay attention to the following explanations regarding *Restore* conditions:

► **Restore (Download / IOL-Master to IOL-Device):**

A Restore (download from IOL-Master to IOL-Device) is performed when an IO-Link Device is connected and the IO-Link Master has valid parameter data stored which are usable for the IOL-Device and not equal compared to the device parameters.

In the *Restore* mode no change of the IOL-Device parameters will be stored permanently on the IOL-Master. When the IOL-Device sets the DS\_UPLOAD\_FLAG in this mode, the device parameters will be restored by the IOL-Master.

The restore procedure can be blocked by the IO-Link Device via the *Device Access Locks* parameter when supported by the IO-Link Device (Index 0x000C, refer to vendor specific IO-Link Device documentation).

### 9.4.11 Vendor ID

The *Vendor ID* is needed for the validation of the IO-Link device and can be configured with this parameter.

Precondition for using the *Vendor ID* is that you configure *Port Mode* to "IO-Link Manual". *Validation and Backup* must be set to a type compatible V1.X device.

### 9.4.12 Device ID

The *Device ID* is needed for the validation of the IO-Link device and can be configured with this parameter.

Precondition for using the *Device ID* is that you configure *Port Mode* to "IO-Link Manual". *Validation and Backup* must be set to a type compatible V1.X device.

### 9.4.13 Cycle Time

The IO-Link cycle time can be configured by this parameter.

The precondition for using *Cycle Time* is that you configure *Port Mode* to "IO-Link Manual".

#### **As fast as possible:**

The IO-Link port uses the max. supported IO-Link Device and Master update cycle time for the cyclic I/O data update between IO-Link Master and IO-Link Device.

#### **1.6 ms, 3.2 ms, 4.8 ms, 8.0 ms, 20.8 ms, 40.0 ms, 80.0 ms, 120.0 ms:**

The cycle time can be set manually to the provided options. This option can be used e.g. for IO-Link devices which are connected over inductive couplers. Inductive couplers are normally the bottleneck in the update cycle time between IO-Link Master and IO-Link Device. In this case, please refer to the data sheet of the inductive coupler.

## 10 Process data assignment

The LioN-X devices in general support process data communication in both directions. The consuming data in this context is defined as the process output data which controls physical outputs and IO-Link output data. The producing data in this context is defined as the process input data which contains the physical inputs, diagnostics and IO-Link input data with optional extended status and event data.

The following chapters describe the data images for the consuming and producing data direction which are assigned to the output and input assemblies.

### 10.1 Consuming data image (output)

Output data frame	Digital output channel control	Reserved (e.g. feature control)	IO-Link output data
Consuming data size	2 Byte, INT	2 Byte, INT	0..256 Byte, INT

The complete *Output data frame* has a variable size of up to 260 Bytes. In general, a 4 Byte Run/Idle Header precedes, resulting in up to 264 Bytes in total.

The following chapters describe the bit assignment.

#### 10.1.1 Digital output channel control

Digital output channel control	Bit	7	6	5	4	3	2	1	0
Channel number (default mapping)	Byte 0	8	7	6	5	4	3	2	1
	Byte 1	16	15	14	13	12	11	10	9

The control values are effective if the respective channels are configured as outputs and the *Digital Output Control* is set to *DO Channel Control*.

### 10.1.2 IO-Link output data

IO-Link output data	IO-Link port 1 control	IO-Link port 2 control	IO-Link port 3 control	IO-Link port 4 control	IO-Link port 5 control	IO-Link port 6 control	IO-Link port 7 control	IO-Link port 8 control
IO-Link port output size	0 Byte	0 Byte	0 Byte	0 Byte	0 Byte	0 Byte	0 Byte	0 Byte
	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte	2 Byte
	4 Byte	4 Byte	4 Byte	4 Byte	4 Byte	4 Byte	4 Byte	4 Byte
	8 Byte	8 Byte	8 Byte	8 Byte	8 Byte	8 Byte	8 Byte	8 Byte
	16 Byte	16 Byte	16 Byte	16 Byte	16 Byte	16 Byte	16 Byte	16 Byte
	32 Byte	32 Byte	32 Byte	32 Byte	32 Byte	32 Byte	32 Byte	32 Byte

The IO-Link port output size is independent of the configured Channel Mode. It is always considered in the IO-Link output data and therefore the offsets do **not** need to be calculated again by the user in case of a channel mode reconfiguration. Every IO-Link port can be set to its required size. The control data is transferred to the device. However, the content depends on the IO-Link *Output Data Swapping Mode* and *Output Data Swapping Offset*.

If there is no IO-Link port configured, the *Consuming data image* has no IO-Link output data.

## 10.2 Producing data image (input)

Input data frame	Digital input channel status	General diagnostics	Sensor diagnostics	Actuator/ U <sub>Aux</sub> diagnostics	IO-Link diagnostics	IO-Link input data
Producing data size	2 Byte, INT	2 Byte, INT	2 Byte, INT	2 Byte, INT	0 Byte 6 Byte, INT	0..432 Byte, INT

The complete *Input data frame* has a variable size of up to 446 Bytes.

The following chapters describe the bit assignment.

### 10.2.1 Digital input channel status

Digital input channel status	Bit	7	6	5	4	3	2	1	0
Channel number (default mapping)	Byte 0	8	7	6	5	4	3	2	1
	Byte 1	16	15	14	13	12	11	10	9

Each status value is effective if the channel is configured as Input.

### 10.2.2 General diagnostics

General diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	IME	FME	DTO	DTU	SCA	SCS	LVA	LVS
	Byte 1	0	0	0	0	IDN	IDW	IDE	IVE

**LVS**

Low Voltage System/Sensor Supply

**LVA**

Low Voltage Actuator Supply

**SCS**

Short Circuit Sensor

**SCA**

Short Circuit Actuator/ $U_L/U_{Aux}$

**DTU**

Device Temperature Underrun

**DTO**

Device Temperature Overrun

**FME**

Force Mode Enabled

**IME**

Internal Module Error

**IVE**

IO-Link Validation Error (collective error)

**IDE**

IO-Link Device Error (collective error)

<b>IDW</b>	IO-Link Device Warning (collective error)
<b>IDN</b>	IO-Link Device Notification (collective error)
<b>0</b>	Reserved

### 10.2.3 Sensor diagnostics

Sensor diagnostics	Bit	7	6	5	4	3	2	1	0
Port number	Byte 0	X8	X7	X6	X5	X4	X3	X2	X1
	Byte 1	0	0	0	0	0	0	0	0

<b>X1 .. 8</b>	Sensor Short Circuit on Port X1 .. X8
<b>0</b>	Reserved

### 10.2.4 Actuator/U<sub>L</sub>/U<sub>Aux</sub> diagnostics

Actuator/U <sub>L</sub> /U <sub>Aux</sub> diagnostics	Bit	7	6	5	4	3	2	1	0
Channel number (fix)	Byte 0	8	7	6	5	4	3	2	1
	Byte 1	16	15	14	13	12	11	10	9

<b>1 .. 16</b>	Actuator/U <sub>L</sub> /U <sub>Aux</sub> channel error on channel 1 .. 16
----------------	--

### 10.2.5 IO-Link diagnostics

IO-Link diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	ICE8	ICE7	ICE6	ICE5	ICE4	ICE3	ICE2	ICE1
	Byte 1	0	0	0	0	0	0	0	0
	Byte 2	IVE8	IVE7	IVE6	IVE5	IVE4	IVE3	IVE2	IVE1
	Byte 3	IDE8	IDE7	IDE6	IDE5	IDE4	IDE3	IDE2	IDE1
	Byte 4	IDW8	IDW7	IDW6	IDW5	IDW4	IDW3	IDW2	IDW1
	Byte 5	IDN8	IDN7	IDN6	IDN5	IDN4	IDN3	IDN2	IDN1

**ICE1..8**

IO-Link Port COM Error (device missing, broken wire, short circuit)

**IVE1..8**

IO-Link Port Validation Error

**IDE1..8**

IO-Link Port Device Error

**IDW1..8**

IO-Link Port Device Warning

**IDN1..8**

IO-Link Port Device Notification

**0**

Reserved

If there is no IO-Link port configured, the input data image doesn't show IO-Link diagnostics.

10.2.6 IO-Link input data

IO-Link input data	IO-Link port 1				[...]	IO-Link port 8			
	Status	PQI	Extended status	Events		Status	PQI	Extended status	Events
IO-Link port input size	0 Byte 2 Byte 4 Byte 8 Byte 16 Byte 32 Byte	2 Byte	0 Byte 8 Byte	0 Byte 12 Byte	[...]	0 Byte 2 Byte 4 Byte 8 Byte 16 Byte 32 Byte	2 Byte	0 Byte 8 Byte	0 Byte 12 Byte

The IO-Link port input size is independent of the configured Channel Mode. It is always considered in the IO-Link input data and therefore the offsets do **not** need to be calculated again by the user in case of a channel mode reconfiguration. Every IO-Link port can be set to its required size. The device input data is mapped to the **Status** field and the content depends on the IO-Link *Input Data Swapping Mode* and *Input Data Swapping Offset*.

An IO-Link port can be configured via the Channel Mode. The PQI provides some IO-Link information, is always available and is independent of the Status size. The Extended Status and Events can be enabled by the IO-Link port configuration.

Port Qualifier Information (PQI):

PQI (Port Qualifier Information)	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	PQ	DevErr	DevCom	PortActive	SubstDev	NewPar	0	0
	Byte 1	0	0	0	0	0	0	0	0

NewPar

Update of Device parameter detected

SubstDev

Substitute device detected (different SerialNumber)



**PortActive**

Port activated

**DevCom**

Device detected and is in PREOPERATE or OPERATE state

**DevErr**

Error/warning assigned to Device or Port occurred

**PQ**

Valid I/O Process Data from Device

**0**

Reserved

Extended status:

IO-Link Extended status	Bit	7	6	5	4	3	2	1	0
Extended diagnostics	Byte 0	0	0	0	ICT	BUI	SPE	ILE	OLE
	Byte 1	0							
Vendor ID	Byte 2	Vendor ID (LSB)							
	Byte 3	Vendor ID (MSB)							
Device ID	Byte 4	Device ID (LSB)							
	Byte 5	Device ID							
	Byte 6	Device ID (MSB)							
	Byte 7	0							

**OLE**

Output process data length error (device mismatch)

**ILE**

Input process data length error (device mismatch)

**SPE**

Startup parameterization error (direct parameter error)

<b>BUI</b>	Backup inconsistency (parameter storage error)
<b>ICT</b>	Invalid Cycle Time
<b>0</b>	Reserved

## Events:

IO-Link events	Bit	7	6	5	4	3	2	1	0
Event Qualifier 1	Byte 0	Mode		Type		0	0	Instance	
	Byte 1	0	0	0	0	0	0	0	0
Event Code 1	Byte 2	Event Code							
	Byte 3								
Event Qualifier 2	Byte 4	Mode		Type		0	0	Instance	
	Byte 5	0	0	0	0	0	0	0	0
Event Code 2	Byte 6	Event Code							
	Byte 7								
Event Qualifier 3	Byte 8	Mode		Type		0	0	Instance	
	Byte 9	0	0	0	0	0	0	0	0
Event Code 3	Byte 10	Event Code							
	Byte 11								

**Instance**                      Unknown ("0"), Reserved ( Physical Layer PL ("1"), Data Link Layer DL ("2"), Application Layer AL ("3") ), Application ("4")

**Type**                              Notification ("1"), Warning ("2"), Error ("3")

**Mode**                              Event single shot ("1"), Event disappears ("2"), Event appears ("3")

Event Code	Diagnostic code reported by the IO-Link device
0	Reserved

## 10.3 Sample applications

The connection and configuration parameters of the device with its variable data sizes provides you an individual approach for realizing your application. The size of each IO-Link port can be determined which also has influence on the process data offsets.

The following application samples describe the process data assignments for the input and output data including the byte offsets. When there is no need to configure the data sizes, use the first sample to get the default byte offsets for your application. When you decide to reduce the data sizes to set them for example to the required IO-Link data lengths or you when do not need the extended status, have a look at the second sample to understand how the data mapping works.

For Rockwell Automation/Allen Bradley PLC customer, it is recommended to use an Add-On Instruction in Studio 5000® as an interface to the process data as described in chapter [Add-On Instruction \(AOI\)](#) on page 102.

### 10.3.1 Process data images – default configuration

The default configuration of the IO-Link port input and output data sizes is set to the maximum sizes in the EDS files. This means the user gets all data of each IO-Link port. The following tables provide you an overview of the data structures and the byte offsets for input and output data:

#### *Connection parameters*

<b>Output data size</b>	260
<b>Input data size</b>	446

Byte offset	Output data
0	Digital output channel control (2 bytes)
2	Reserved (2 bytes)
4	IO-Link port1 data (control, 32 bytes)
36	IO-Link port2 data (control, 32 bytes)
68	IO-Link port3 data (control, 32 bytes)
100	IO-Link port4 data (control, 32 bytes)
132	IO-Link port5 data (control, 32 bytes)
164	IO-Link port6 data (control, 32 bytes)
196	IO-Link port7 data (control, 32 bytes)
228	IO-Link port8 data (control, 32 bytes)

*Table 14: Default output process data*

Byte offset	Input data
0	Digital input channel status (2 bytes)
2	General diagnostics (2 bytes)
4	Sensor diagnostics (2 bytes)
6	Actuator diagnostics (2 bytes)
8	IO-Link diagnostics (6 bytes)
14	IO-Link port1 data (status, 32 bytes)
46	IO-Link port1 PQI (2 bytes)
48	IO-Link port1 extended status (8 bytes)
56	IO-Link port1 events (12 bytes)
68	IO-Link port2 data (status, 32 bytes)
100	IO-Link port2 PQI (2 bytes)
102	IO-Link port2 extended status (8 bytes)
110	IO-Link port2 events (12 bytes)
122	IO-Link port3 data (status, 32 bytes)
154	IO-Link port3 PQI (2 bytes)
156	IO-Link port3 extended status (8 bytes)

Byte offset	Input data
164	IO-Link port3 events (12 bytes)
176	IO-Link port4 data (status, 32 bytes)
208	IO-Link port4 PQI (2 bytes)
210	IO-Link port4 extended status (8 bytes)
218	IO-Link port4 events (12 bytes)
230	IO-Link port5 data (status, 32 bytes)
262	IO-Link port5 PQI (2 bytes)
264	IO-Link port5 extended status (8 bytes)
272	IO-Link port5 events (12 bytes)
284	IO-Link port6 data (status, 32 bytes)
316	IO-Link port6 PQI (2 bytes)
318	IO-Link port6 extended status (8 bytes)
326	IO-Link port6 events (12 bytes)
338	IO-Link port7 data (status, 32 bytes)
370	IO-Link port7 PQI (2 bytes)
372	IO-Link port7 extended status (8 bytes)
380	IO-Link port7 events (12 bytes)
392	IO-Link port8 data (status, 32 bytes)
424	IO-Link port8 PQI (2 bytes)
426	IO-Link port8 extended status (8 bytes)
434	IO-Link port8 events (12 bytes)

*Table 15: Default input process data*

### 10.3.2 Process data images with modified data sizes

The IO-Link port input and output data sizes and the existence of the extended status can be modified by the configuration assembly. This means the user can decide about which data is mapped to the process data. The following configuration tables provide you a sample and an overview of possible data structures and the byte offsets for input and output data:

*Connection parameters***Output data size** 62**Input data size** 66*IO-Link Port1***Output data size** 2 Byte**Input data size** 2 Byte**Input data extension** No Data*IO-Link Port2***Output data size** 32 Byte**Input data size** 0 Byte**Input data extension** Extended Status*IO-Link Port3***Output data size** 16 Byte**Input data size** 4 Byte**Input data extension** Extended Status + Events*IO-Link Port4***Output data size** 8 Byte**Input data size** 2 Byte**Input data extension** No Data*IO-Link Port5 .. 8***Output data size** 0 Byte**Input data size** 0 Byte**Input data extension** No Data

Byte offset	Output data	Input data
0	Digital output channel control (2 bytes)	Digital input channel status (2 bytes)
2	Reserved (2 bytes)	General diagnostics (2 bytes)
4	IO-Link port1 data (control, 2 bytes)	Sensor diagnostics (2 bytes)
6	IO-Link port2 data (control, 32 bytes)	Actuator diagnostics (2 bytes)
8		IO-Link diagnostics (6 bytes)
10		
12		
14		
16		IO-Link port1 data (status, 2 bytes)
18		IO-Link port1 PQI (2 bytes)
20		IO-Link port2 PQI (2 bytes)
22		IO-Link port2 extended status (8 bytes)
24		
26		
28		
30		IO-Link port3 data (status, 4 bytes)
32		IO-Link port3 PQI (2 bytes)
34		IO-Link port3 extended status (8 bytes)
36		
38		
40		
42	IO-Link port3 data (control, 16 bytes)	IO-Link port3 events (12 bytes)
44		
46		
48		
50		
52		
54		IO-Link port4 data (status, 2 bytes)
56		IO-Link port4 PQI (2 bytes)

Byte offset	Output data	Input data
58		IO-Link port5 PQI (2 bytes)
60		IO-Link port6 PQI (2 bytes)
62		IO-Link port7 PQI (2 bytes)
64		IO-Link port8 PQI (2 bytes)
66		

Table 16: Modified process data



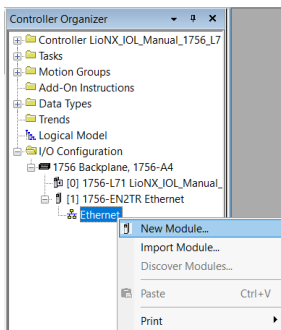
# 11 Configuration and operation with Rockwell Automation Studio 5000®

The configuration and start-up of the LioN-X devices described on the following pages refers to Rockwell Automation Studio 5000®, V30. If you are using an engineering tool from another provider, please consider the related documentation.

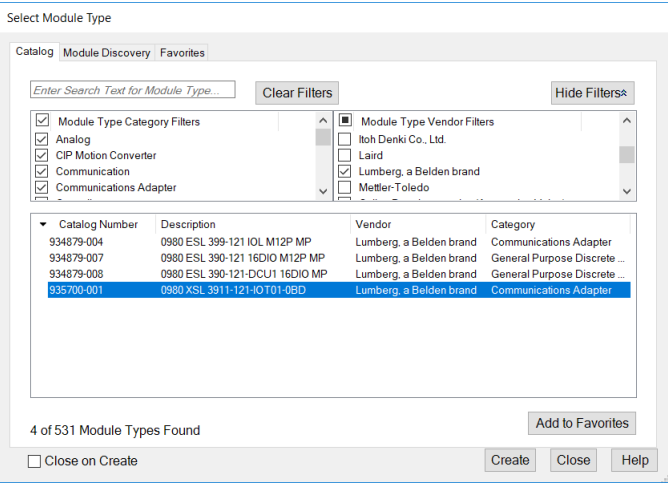
## 11.1 Basic commissioning

Perform the following working steps:

1. Create a new project in Studio 5000®.
2. Select the correct controller.
3. When no integrated EtherNet/IP interface is available, add the proper communication interface to your backplane under **Controller Organizer > I/O-Configuration**.
4. Set a communication path to enable the project download.
5. Install the EDS files of the LioN-X devices in Studio 5000® with the EDS hardware installation tool.
6. Go to **Controller Organizer > I/O-Configuration** and right-click the Ethernet interface.

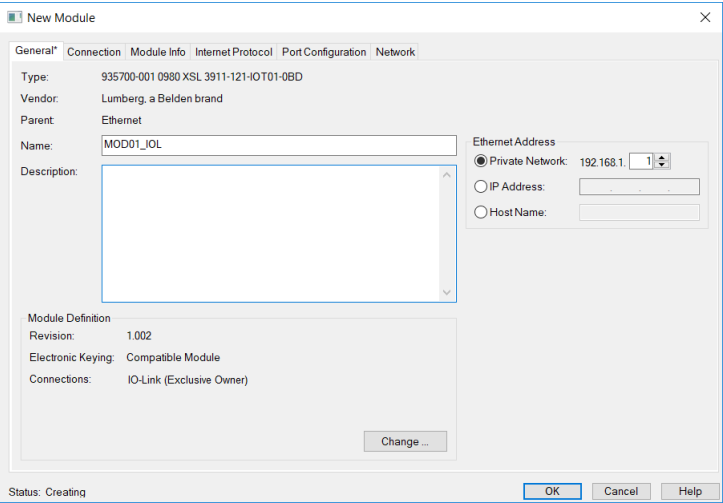


7. Select **New Module** in the menu. The following selection window opens:



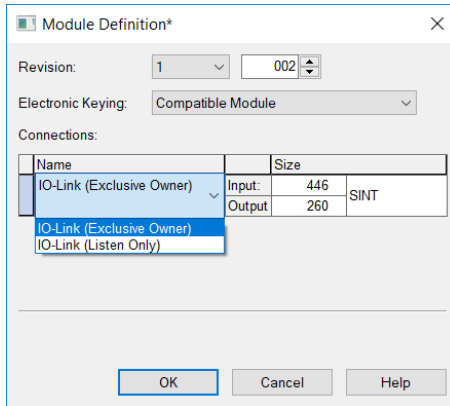
8. Use the **Module Type Vendor Filter** on the right side to display all installed devices of Lumberg Automation™.

9. Select the device you wish to add and click on **Create**.



**10.** Enter a name for the device and set the chosen IP address. In this example, the name is **MOD01\_IOL** and the IP address is **192.168.1.1**.

**11.** Click on **Change** in order to change the settings for the device revision, electronic keying and connection type.



**12.** Select the connection type and configure the total sizes of the input and output process data. The sizes depend on the number of connected IO-Link devices and their data lengths of both directions. Each device input and output data size must also be set later in the IO-Link port configuration. The selection of the data type refers to the type in which Studio 5000® maps the input and output data. The default data type is SINT. The INT type is selectable when each size is a multiple of 2. The DINT type is selectable when each size is a multiple of 4. Click on **OK**.

**13.** In the **Connection** folder of the **Module Properties**, you see the selected connection. This folder also lets you define the **Requested Packet Interval (RPI)** and the EtherNet/IP connection type. A value of 1 ms is the minimum for parameter RPI and the connection types *Unicast* or *Multicast* can be chosen. Apply the settings.

New Module

General\*Connection\*Module Info\*Internet Protocol\*Port Configuration\*Network\*

Name	Requested Packet Interval (RPI) (ms)	Connection over EtherNet/IP	Input Trigger
IO-Link (Exclusive Owner)	20.01.0 - 9999.9	Unicast	Cyclic

☐ Inhibit Module

☐ Major Fault On Controller If Connection Fails While in Run Mode

Module Fault

Status: CreatingOKCancelHelp

14. Move to **Controller-Tags** in **Controller Organizer**. The controller tags for the configuration parameters contain the name of the device, followed by a **":C"**. The configuration parameters can be set under **Value** and are described in chapter [Configuration parameters](#) on page 59.

Scope: LiNX\_Manual Show: All Tags

Name	Value	Force Mask	Style	Data Type
- MOD01_IOLC	{ ... }	{ ... }		_0015.935700_001
MOD01_IOLC.Force_Mode_Lock	0		Decimal	BOOL
MOD01_IOLC.Web_Interface_Lock	0		Decimal	BOOL
MOD01_IOLC.Digital_Output_Control	0		Decimal	BOOL
MOD01_IOLC.Report_UL_UAux_Supply_Voltage_Fault	1		Decimal	BOOL
MOD01_IOLC.Report_DO_Fault_without_UL_UAux	1		Decimal	BOOL
MOD01_IOLC.CIP_object_configuration_lock	0		Decimal	BOOL
MOD01_IOLC.External_configuration_lock	0		Decimal	BOOL
+ MOD01_IOLC.IO_Mapping_Mode	0		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port1_Ch_A	0		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port1_Ch_B	1		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port2_Ch_A	2		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port2_Ch_B	3		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port3_Ch_A	4		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port3_Ch_B	5		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port4_Ch_A	6		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port4_Ch_B	7		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port5_Ch_A	8		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port5_Ch_B	9		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port6_Ch_A	10		Decimal	SINT
+ MOD01_IOLC.IO_Mapping_Port6_Ch_B	11		Decimal	SINT

**15.** The tag of the input process data contain the name of the device, followed by a **"I.Data"**. The output process data has the same name followed by a **"O.Data"**. Both arrays show its configured data sizes. The content of them is described in chapter [Process data assignment](#) on page 83.

Scope:  LibNX_IOL_Manual		Show: All Tags		
Name	Value	Force Mask	Style	Data Type
+ MOD01_IOLC	{ ... }	{ ... }		_0015.935700_
- MOD01_IOLI	{ ... }	{ ... }		_0015.935700_
- MOD01_IOLI.ConnectionFaulted	0		Decimal	BOOL
+ MOD01_IOLI.Data	{ ... }	{ ... }	Decimal	SINT[446]
- MOD01_IOL0	{ ... }	{ ... }		_0015.935700_
- MOD01_IOL0.Data	{ ... }	{ ... }	Decimal	SINT[260]
+ MOD01_IOL0.Data[0]	0		Decimal	SINT
+ MOD01_IOL0.Data[1]	0		Decimal	SINT
+ MOD01_IOL0.Data[2]	0		Decimal	SINT
+ MOD01_IOL0.Data[3]	0		Decimal	SINT
+ MOD01_IOL0.Data[4]	0		Decimal	SINT
+ MOD01_IOL0.Data[5]	0		Decimal	SINT
+ MOD01_IOL0.Data[6]	0		Decimal	SINT
+ MOD01_IOL0.Data[7]	0		Decimal	SINT
+ MOD01_IOL0.Data[8]	0		Decimal	SINT

**16.** When the configuration is completed, the parameters can be downloaded to the EtherNet/IP controller.

## 11.2 Add-On Instruction (AOI)

Rockwell Automation Studio 5000® provides the user a mechanism for the optimization and encapsulation of data and logic via an Add-On Instruction. This AOI can be added to a rung as any other pre-defined instruction in the controller and is useful for the preprocessing of the input and output data of a device.

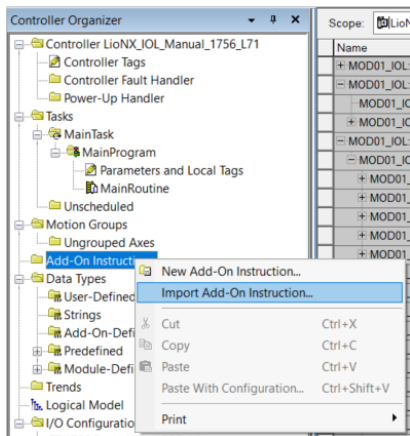
With the help of User-Defined Data Types (UDT), the user gets a comprehensible interface with a clear naming and description for each field of the process data. The advantage is that it is no longer required to calculate byte offsets of the input and output data. Every field of the process data can be directly addressed via a unique name.

Belden provides AOIs for customers which can be downloaded from the product pages on our online catalog: [catalog.belden.com](https://catalog.belden.com).

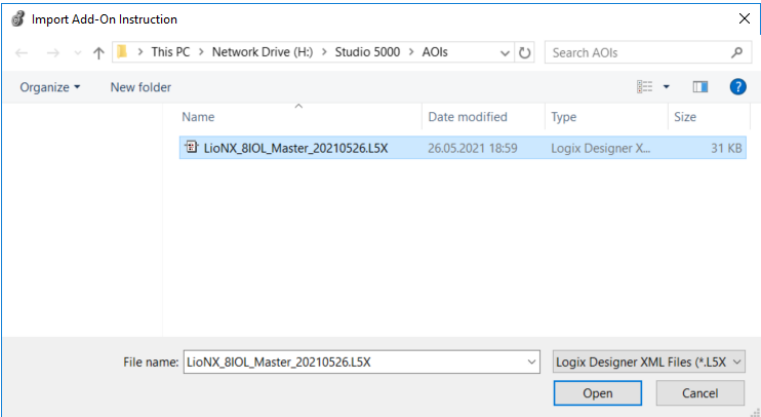
On request, an AOI is also sent to you by the Belden support team.

Perform the following working steps for using an AOI:

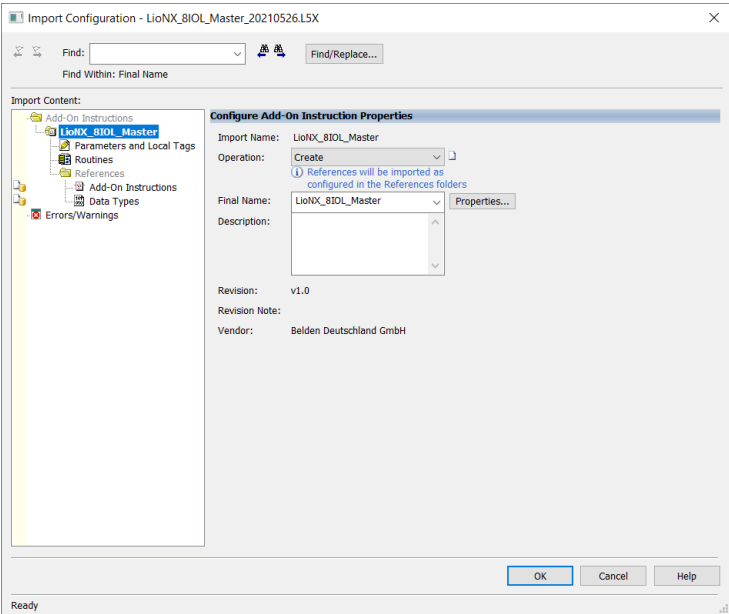
1. In your Studio 5000® project, navigate to **Controller Organizer**, right-click on **Add-On Instructions** and click **Import Add-On Instruction...**:



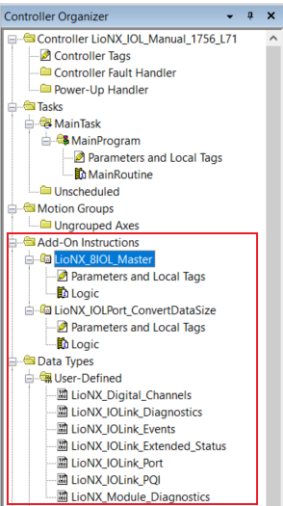
2. Open the \*.L5X file:



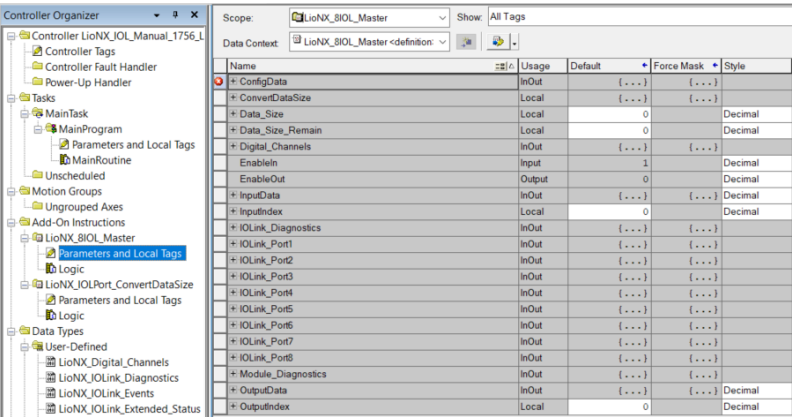
3. Click OK to create the AOI with all necessary User-Defined Data Types:



4. The imported components are now shown in the **Controller Organizer**:



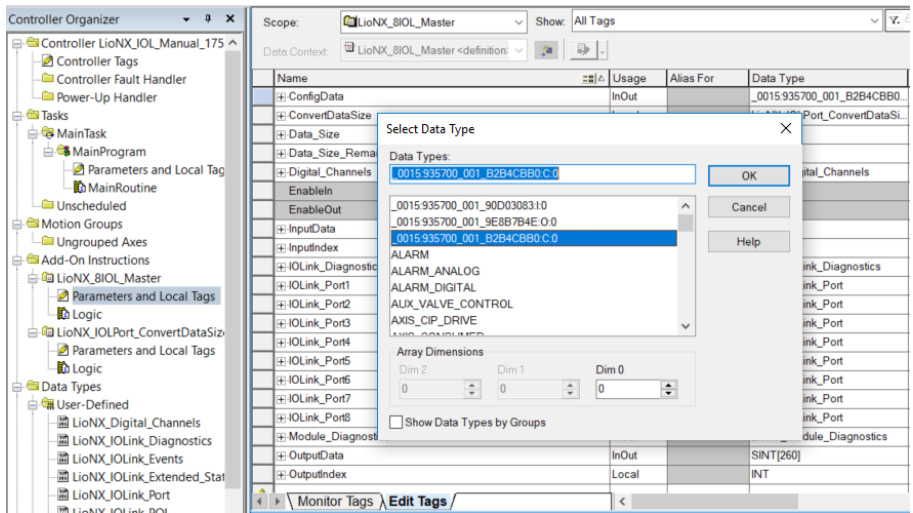
5. Check if an error is shown in the AOI tags (red circle with a white cross). This can happen for the configuration data when you have imported an AOI on your system for the first time:



If no error occurred, continue with step 9.

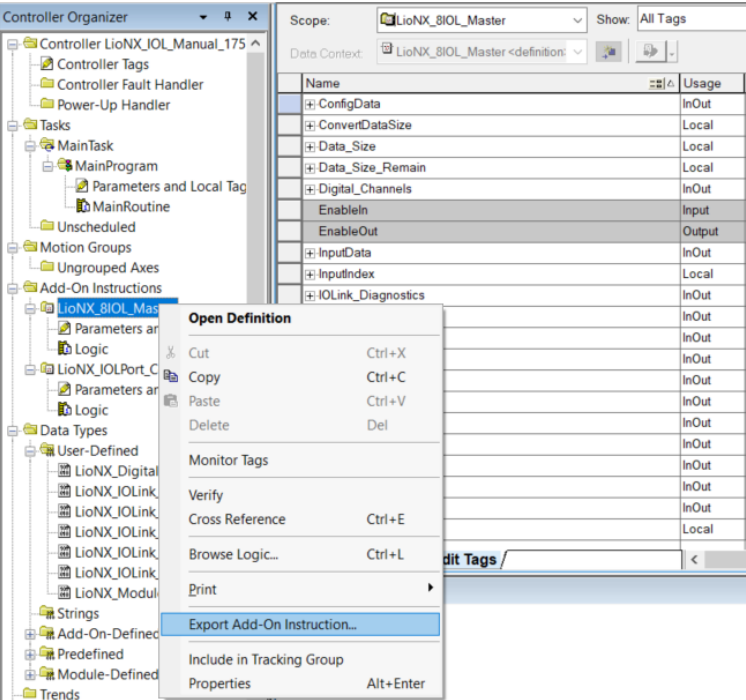


**6. Go to **Edit Tags** and change the data type to the Module-Defined type on your system:**

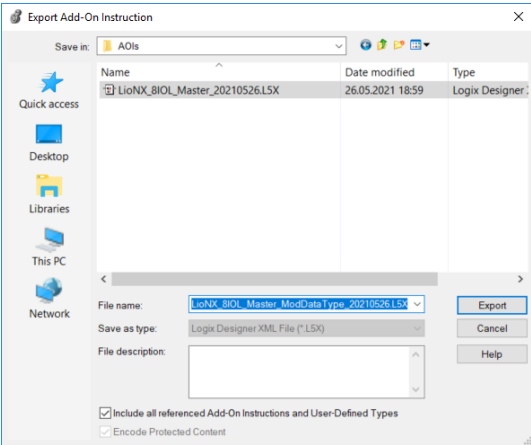


The data type must match the string until the third underscore (\_). The CRC32 before :C:0 is system dependent and does not match the one from the imported AOI. The error is resolved when the red symbol in front of the line is cleared.

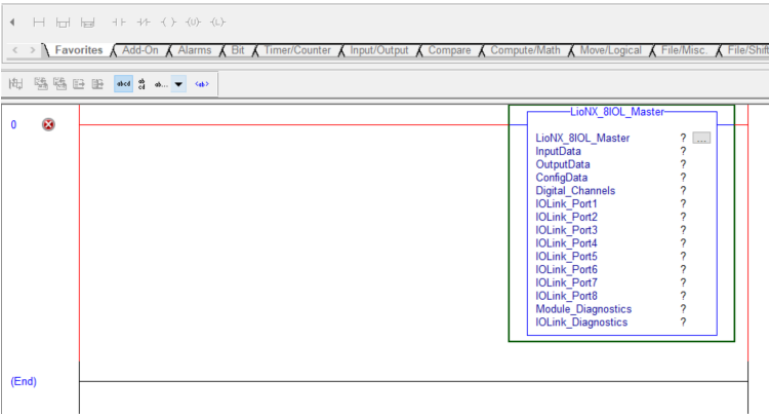
7. When you have changed a data type in the AOI, you should do an export to store this version for future use in other projects on your system. Right-click the AOI and click **Export Add-On Instruction...**:



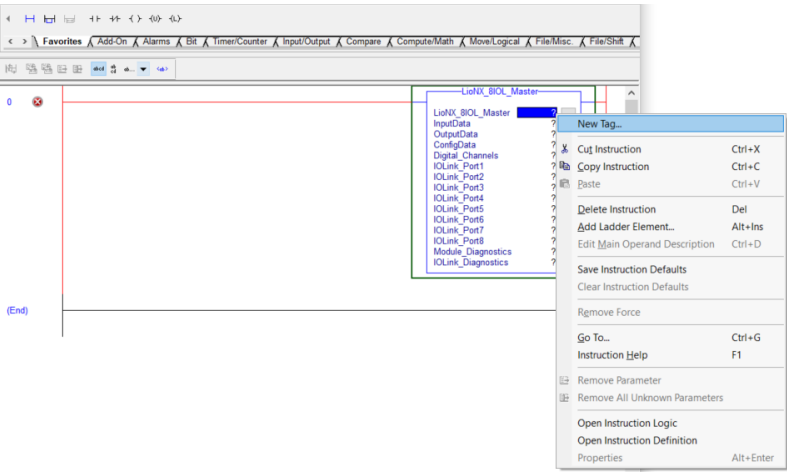
8. Edit the file name and save the AOI:



9. For using the AOI go to a logic, e.g. the *MainRoutine*, and add the IO-Link Master AOI via drag and drop to the rung:



10. Right-Click on the first element of the AOI and click **New Tag...**:



11. Enter a name and create the tag for the AOI:

New Tag

Name: MOD01\_IOL\_AOI Create

Description:

Usage: <controller>

Type: Base Connection...

Alias For:

Data Type: LioNX\_BIOL\_Master

Parameter Connection:

Scope: LioNX\_IOL\_Manual\_1756\_L71

External Access: Read/Write

Style:

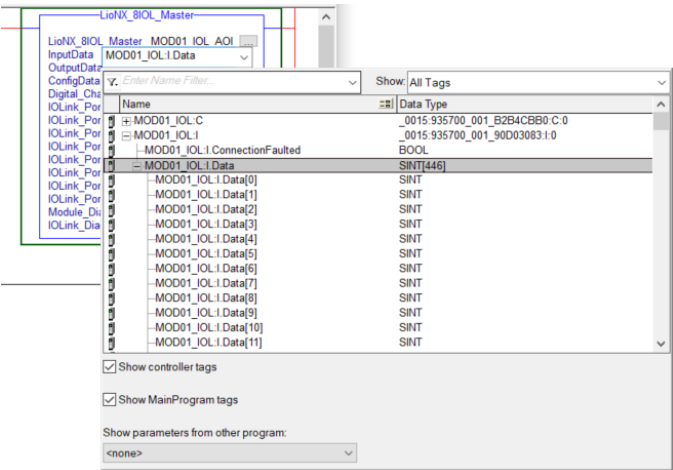
☐ Constant

☐ Sequencing

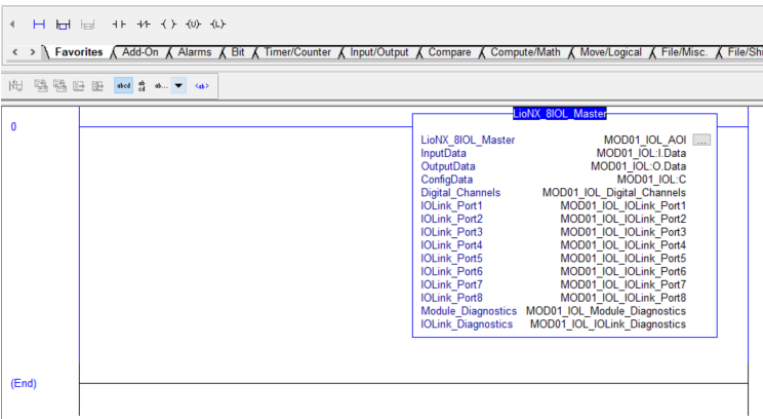
☐ Open Configuration

☐ Open Parameter Connections

12. Assign the input, output and configuration data of the module:



13. Create the tags for the remaining elements as shown in steps 10. and 11.:



**14.** From now on, your logic does not need to synchronously copy the input and output data anymore. It just uses the new data tags as the interface for exchanging process data with the module:

Name	Value	Force Mask	Style	Data Type	Description
LiNX_IOL_Manual					
MOD01_IOL_Digital_Channels	(...)	(...)		LiNX_Digital_Channels	
+ MOD01_IOL_Digital_Channels Control	(...)	(...)	Decimal	SINT[2]	Digital Output Data, default mapping: Bit0=PortK1Ch.A...
+ MOD01_IOL_Digital_Channels Status	(...)	(...)	Decimal	SINT[2]	Digital Input Channel Status, default mapping: Bit0=PortK...
MOD01_IOL_IOLink_Diagnostics	(...)	(...)		LiNX_IOLink_Diagnostics	
+ MOD01_IOL_IOLink_Diagnostics COM_Error	0		Decimal	SINT	Bit0=IOLinkPort1... Bit7=IOLinkPort8
+ MOD01_IOL_IOLink_Diagnostics Reserved	0		Decimal	SINT	not in use
+ MOD01_IOL_IOLink_Diagnostics Validation_Error	0		Decimal	SINT	Bit0=IOLinkPort1... Bit7=IOLinkPort8
+ MOD01_IOL_IOLink_Diagnostics Device_Error	0		Decimal	SINT	Bit0=IOLinkPort1... Bit7=IOLinkPort8
+ MOD01_IOL_IOLink_Diagnostics Device_Warning	0		Decimal	SINT	Bit0=IOLinkPort1... Bit7=IOLinkPort8
+ MOD01_IOL_IOLink_Diagnostics Device_Notification	0		Decimal	SINT	Bit0=IOLinkPort1... Bit7=IOLinkPort8
MOD01_IOL_IOLink_Port1	(...)	(...)		LiNX_IOLink_Port	
+ MOD01_IOL_IOLink_Port1 Control	(...)	(...)	Decimal	SINT[32]	IO-Link Port Output Data
+ MOD01_IOL_IOLink_Port1 Status	(...)	(...)	Decimal	SINT[32]	IO-Link Port Input Data
+ MOD01_IOL_IOLink_Port1 PQI	(...)	(...)		LiNX_IOLink_PQI	IO-Link Port PQI Data
+ MOD01_IOL_IOLink_Port1 PQI PQI_Byte	0		Decimal	SINT	Bit2=NewPar, Bit3=SubstDev, Bit4=PortActive, Bit5=Dev...
+ MOD01_IOL_IOLink_Port1 PQI Reserved	0		Decimal	SINT	not in use
MOD01_IOL_IOLink_Port1 Extended_Status	(...)	(...)		LiNX_IOLink_Extended_Status	IO-Link Port Extended Status
+ MOD01_IOL_IOLink_Port1 Extended_Status Extended_Diag	0		Decimal	INT	Bit0=OutDataLenErr, Bit1=InDataLenErr, Bit2=StartupPar...
+ MOD01_IOL_IOLink_Port1 Extended_Status Vendor_ID	0		Decimal	INT	Vendor ID
+ MOD01_IOL_IOLink_Port1 Extended_Status Device_ID	0		Decimal	DINT	Device ID
MOD01_IOL_IOLink_Port1 Events	(...)	(...)		LiNX_IOLink_Events	IO-Link Port Events
+ MOD01_IOL_IOLink_Port1 Events Event_Qualifier1	0		Decimal	INT	Bit0, 1=Instance, Bit4, 5=Type, Bit6, 7=Mode
+ MOD01_IOL_IOLink_Port1 Events Event_Code1	0		Decimal	INT	Event Code
+ MOD01_IOL_IOLink_Port1 Events Event_Qualifier2	0		Decimal	INT	Bit0, 1=Instance, Bit4, 5=Type, Bit6, 7=Mode
+ MOD01_IOL_IOLink_Port1 Events Event_Code2	0		Decimal	INT	Event Code
+ MOD01_IOL_IOLink_Port1 Events Event_Qualifier3	0		Decimal	INT	Bit0, 1=Instance, Bit4, 5=Type, Bit6, 7=Mode
+ MOD01_IOL_IOLink_Port1 Events Event_Code3	0		Decimal	INT	Event Code
MOD01_IOL_IOLink_Port2	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_IOLink_Port3	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_IOLink_Port4	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_IOLink_Port5	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_IOLink_Port6	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_IOLink_Port7	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_IOLink_Port8	(...)	(...)		LiNX_IOLink_Port	
MOD01_IOL_Module_Diagnostics	(...)	(...)		LiNX_Module_Diagnostics	
+ MOD01_IOL_Module_Diagnostics General	0		Decimal	INT	Bit0=LowVoltsSys, Bit1=LowVoltsAct, Bit2=ShortCircSen, B...
+ MOD01_IOL_Module_Diagnostics Sensor	0		Decimal	INT	Bit0=PortK1... Bit7=PortK8
+ MOD01_IOL_Module_Diagnostics Actuator	0		Decimal	INT	Bit0=PortK1Ch.A... Bit15=PortK8Ch.B

### Note:

If you decide to reduce the input and output data sizes of the connection, you also must set these new sizes to the SINT arrays of *InputData* and *OutputData* within the AOI. In steps **6.** to **8.** it is described how to modify data types of an AOI and how to store the changes.

# 12 CIP object classes

## 12.1 EtherNet/IP object classes

According to the CIP specification, the LioN-X variants support the following standard EtherNet/IP object classes:

Object Class	Object ID	Instances
Identity Object	0x01	0, 1
Message Router Object	0x02	0 (only on class level)
Assembly Object	0x04	0, 130, 131, 145
Connection Manager Object	0x06	0 (only on class level)
Discrete Input Point Object	0x08	0, 1 .. 16
DLR Object	0x47	0, 1
QoS Object	0x48	0, 1
TCP/IP Interface Object	0xF5	0, 1
Ethernet Link Object	0xF6	0, 1 .. 2
LLDP Management Object	0x109	0, 1

All objects with instance attributes are described in the following chapters.

12.1.1 Identity Object (0x01)

Supported services:

Get Attributes All (0x01)

Reset (0x05): 0 = Reset Module (Warmstart), 1 = Reset to Factory Default

Get Attribute Single (0x0E)

Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device
6	Maximum ID Number Class Attributes	Get	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device
7	Maximum ID Number Instance Attributes	Get	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.



**Instance attribute (Instance 1)**

Attribute	Name	Access	Data Type	Description
1	Vendor ID	Get	UINT	Vendor Identification
2	Device Type	Get	UINT	Indication of general type of product
3	Product Code	Get	UINT	Identification of a particular product of an individual vendor
4	Revision	Get	USINT, USINT	Structure with major and minor revision
5	Status	Get	WORD	<p>Summary status of device:</p> <p>b0: Owned</p> <p>b1: Reserved ("0")</p> <p>b2: Configured</p> <p>b3: Reserved ("0")</p> <p>b4 .. 7: Extended Device Status</p> <p>0 = Self-Testing or Unknown</p> <p>1 = Firmware Update in Progress</p> <p>2 = At least one faulted I/O connection</p> <p>3 = No I/O connections established</p> <p>4 = Non-Volatile Configuration bad</p> <p>5 = Major Fault</p> <p>6 = At least one I/O connection in RUN mode</p> <p>7 = At least one I/O connection established, all in IDLE mode</p> <p>8 = Unused (valid only for instances grater than "1")</p> <p>9 = Reserved</p> <p>10 .. 15 = Vendor specific</p> <p>b8: Minor Recoverable Fault</p> <p>b9: Minor Unrecoverable Fault</p> <p>b10: Major Recoverable Fault</p> <p>b11: Major Unrecoverable Fault</p> <p>b12 .. 15: Reserved ("0")</p>
6	Serial Number	Get	UDINT	Serial number of device
7	Product Name	Get	STRING	Human readable identification

Attribute	Name	Access	Data Type	Description
8	State	Get	USINT	Present state of the device: 0 = Nonexistent 1 = Device Self Testing 2 = Standby 3 = Operational 4 = Major Recoverable Fault 5 = Major Unrecoverable Fault 6 .. 254 = Reserved 255 = Default Value
9	Configuration Consistency Value	Get	UINT	Can be a CRC, incrementing count or any other mechanism (vendor specific behavior) to reflect a non-volatile configuration change
19	Protection Mode	Get	WORD	Current protection mode of the device: b0: Implicit Protection enabled b1 .. 2: Reserved b3: Explicit Protection enabled b4 .. 15: Reserved

### 12.1.2 Assembly Object (0x04)

#### Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device
3	Number of Instances	Get	UINT	Number of Instances currently created in this class level of the device
6	Maximum ID Number Class Attributes	Get	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device
7	Maximum ID Number Instance Attributes	Get	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.

#### Instance attribute (Instance <AssemblyID>)

Attribute	Name	Access	Data Type	Description
3	Data	Get, Set	ARRAY	Assembly Data (Set service only available for consuming assemblies that are not part of an active implicit connection)
4	Size	Get	UINT	Number of bytes in Attribute 3

### 12.1.3 Discrete Input Point Object (0x08)

**Supported services:**

Get Attribute Single (0x0E)

**Class attribute (Instance 0)**

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object

**Instance attribute (Instance 1 .. 16)**

Attribute	Name	Access	Data type	Description
3	Value	Get	BOOL	Input Point Value (0 = OFF, 1 = ON)
4	Status	Get	BOOL	Input Point Status (0 = OK, 1 = Alarm)

### 12.1.4 DLR Object (0x47)

#### Supported services:

Get Attributes All (0x01)

Get Attribute Single (0x0E)

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device
6	Maximum ID Number Class Attributes	Get	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device
7	Maximum ID Number Instance Attributes	Get	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.

Instance attribute (Instance 1)

Attribute	Name	Access	Data type	Description
1	Network Topology	Get	BOOL	0 = Linear 1 = Ring
2	Network Status	Get	BOOL	0 = Normal operation 1 = Ring Fault 2 = Unexpected Loop Detected 3 = Partial Network Fault 4 = Rapid Fault/Restore Cycle
10	Active Supervisor Address	Get	ARRAY	Supervisor IP Address, Supervisor MAC Address (0 = not configured)
12	Capability Flags	Get	DWORD	Flag description: b0: Announce-based Ring Node ("0") b1: Beacon-based Ring Node ("1") b2 .. 4: Reserved ("0") b5: Supervisor Capable ("0") b6: Redundant Gateway Capable ("0") b7: Flush_Table frame Capable ("1") b8 .. 15: Reserved ("0")

### 12.1.5 QoS Object (0x48)

#### Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device
6	Maximum ID Number Class Attributes	Get	UINT	The attribute ID number of the last class attribute of the class definition implemented in the device
7	Maximum ID Number Instance Attributes	Get	UINT	The attribute ID number of the last instance attribute of the class definition implemented in the device.

Instance attribute (Instance 1)

Attribute	Name	Access	Data type	Description
1	802.1Q Tag Enable	Get, Set	USINT	Enables ("1") or disables ("0") sending 802.1Q frames on CIP and IEEE 1588 messages (default value "0")
2	DSCP PTP Event	Get, Set	USINT	DSCP value for PTP Event frames (default value "59")
3	DSCP PTP General	Get, Set	USINT	DSCP value for PTP General frames (default value "47")
4	DSCP Urgent	Get, Set	USINT	CIP transport class 0/1 messages with Urgent priority (default value "55")
5	DSCP Scheduled	Get, Set	USINT	CIP transport class 0/1 messages with Scheduled priority (default value "47")
6	DSCP High	Get, Set	USINT	CIP transport class 0/1 messages with High priority (default value "43")
7	DSCP Low	Get, Set	USINT	CIP transport class 0/1 messages with Low priority (default value "31")
8	DSCP Explicit	Get, Set	USINT	CIP UCMM, CIP transport class 2/3, All other EtherNet/IP encapsulation messages (default value "27")



### 12.1.6 TCP/IP Object (0xF5)

#### Supported services:

Get Attributes All (0x01)

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.

#### Instance attribute (Instance 1)

Attribute	Name	Access	Data type	Description
1	Status	Get	DWORD	Interface Status description: b0 .. 3: Interface Configuration Status 0 = Not configured 1 = Configuration obtained by BOOTP, DHCP or stored value 2 = Configuration obtained by hardware settings (e.g. rotary switches) 3 .. 15 = Reserved b4: Mcast Pending b5: Interface Configuration Pending b6: Acd Status b7: Acd Fault b8 .. 31: Reserved ("0")

Attribute	Name	Access	Data type	Description
2	Configuration Capability	Get	DWORD	Interface Capability Flags: b0: BOOTP Client ("1") b1: DNS Client ("0") b2: DHCP Client ("1") b3: DHCP-DNS Update ("0") b4: Configuration Settable ("1") b5: Hardware Configurable (0 = no rotary switches; 1 = rotary switches available) b6: Interface Configuration Change Requires Reset ("0") b7: Acd Capable ("1") b8 .. 31: Reserved ("0")
3	Configuration Control	Get, Set	DWORD	Interface Control Flags: b0 .. 3: Configuration Method: 0 = Stored Value 1 = BOOTP 2 = DHCP 3 .. 15 = Reserved b4: DNS Enable ("0") b5 .. 31: Reserved ("0")
4	Physical Link Object	Get	STRUCT	Path to physical link object
5	Interface Configuration	Get, Set	STRUCT	TCP/IP network interface configuration
6	Host Name	Get, Set	STRING	Host name of the device (length of 0 = not configured)
10	Select Acd	Get, Set	BOOL	Enables ("1") or disables ("0") the use of ACD (default value "1")
11	Last Conflict Detected	Get, Set	STRUCT	Structure containing information related to the last conflict detected
13	Encapsulation Inactivity Timeout	Get, Set	UINT	Number of seconds of inactivity before TCP connection is closed: 0 = disable 1 .. 3600 = timeout in seconds 120 = default value

### 12.1.7 Ethernet Link Object (0xF6)

**Supported services:**

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

Get and Clear (0x4C)

**Class attribute (Instance 0)**

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.
3	Number of Instances	Get	UINT	Number of object instances currently created at this class level of the device (in this case number of ethernet ports)

Instance attribute (Instance 1 .. 2)

Attribute	Name	Access	Data type	Description
1	Interface Speed	Get	UDINT	Current Interface speed in Mbps
2	Interface Flags	Get	DWORD	Interface Flags: b0: Link Status b1: Half ("0") or Full ("1") Duplex b2 .. 4: Negotiation Status: 0 = Auto-negotiation in progress 1 = Auto-negotiation and speed detection failed (using default 10Mbps and half duplex) 2 = Auto negotiation failed but detected speed (using default half duplex) 3 = Successfully negotiated speed and duplex 4 = Auto-negotiation not attempted (forced speed and duplex) b5: Manual Setting Requires Reset b6: Local Hardware Fault b7 .. 31: Reserved ("0")
3	Physical Address	Get	ARRAY	MAC address
4	Interface Counters	Get	STRUCT	Interface Counters
5	Media Counters	Get	STRUCT	Media-specific counters
6	Interface Control	Get, Set	STRUCT	Configuration for physical interface Control Bits (WORD): b0: Auto-negotiate b1: Forced Duplex Mode (0 = Half Duplex; 1 = Full Duplex, only valid when Auto-negotiate = 0) b2 .. 15: Reserved ("0") <i>Forced Interface Speed in Mbps (UINT)</i>

Attribute	Name	Access	Data type	Description
7	Interface Type	Get	USINT	Type of interface: 0 = Unknown interface type 1 = Internal interface 2 = Twisted-pair 3 = Optical fiber 4 .. 255 = Reserved
8	Interface State	Get	USINT	State of interface: 0 = Unknown 1 = Enabled and ready to send and receive data 2 = Disabled 3 = Testing 4 .. 255 = Reserved
9	Admin State	Get, Set	USINT	Administrative state: 0 = Reserved 1 = Enable interface 2 = Disable interface 3 .. 255 = Reserved
10	Interface Label	Get	STRING	Human readable identification (size max. 64)
11	Interface Capability	Get	STRUCT	Interface Capability Flags (DWORD): b0: Manual Setting Requires Reset ("0") b1: Auto-negotiate ("1") b2: Auto-MDIX ("1") b3: Manual Speed/Duplex ("1") b4 .. 31: Reserved ("0")  Speed/Duplex Array Count of following struct (USINT, 4)  Interface Speed in Mbps (UINT, 10/100)  Interface Duplex Mode (USINT, 0/1): 0 = Half Duplex 1 = Full Duplex 2 .. 255 = Reserved

### 12.1.8 LLDP Management Object (0x109)

**Supported services:**

Get Attributes All (0x01)

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

**Class attribute (Instance 0)**

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.
3	Number of Instances	Get	UINT	Number of object instances currently created at this class level of the device (in this case number of ethernet ports)
6	Maximum ID Number Class Attributes	Get	UINT	Attribute ID number of the last class attribute
7	Maximum ID Number Instance Attributes	Get	UINT	Attribute ID number of the last class attribute

**Instance attribute (Instance 1)**

Attribute	Name	Access	Data type	Description
1	LLDP Enable	Get/Set	STRUCT	<p>LLDP Enable Array Length (UINT): 1 + Class attribute 2 from the Ethernet Link Object (0xF6) = 3</p> <p>LLDP Enable Array (BYTE):</p> <p>b0: Global Enable, LLDP Tx &amp; Rx Enabled (1)</p> <p>b1: LLDP Tx Enabled (Instance 1 of Ethernet Link Object) (1)</p> <p>b2: LLDP Tx Enabled (Instance 2 of Ethernet Link Object) (1)</p>
2	msgTxInterval	Get/Set	UINT	<p>From 802.1AB-2016: Interval in seconds for transmitting LLDP frames from this device</p> <p>0 .. 4 = Reserved</p> <p>5 .. 32768 = Message Transmission Interval for LLDP frames (30)</p> <p>32769 .. 65535 = Reserved</p>
3	msgTxHold	Get/Set	USINT	<p>From 802.1AB-2016: Multiplier of msgTxInterval to determine the value of the TTL TLV sent to neighboring devices</p> <p>0 = Reserved</p> <p>1 .. 100 = Message Transmission Multiplier for LLDP Frames (4)</p> <p>101 .. 255 = Reserved</p>
4	LLDP Datastore	Get	WORD	<p>Indication of the retrieval methods for the LLDP database:</p> <p>b0: LLDP Data Table Object (0)</p> <p>b1: SNMP (1)</p> <p>b2: NETCONF YANG (0)</p> <p>b3: RESTCONF YANG (0)</p> <p>b4 .. b15: Reserved (0)</p>
5	Last Change	Get	UDINT	Counter in seconds from the last time any entry in the local LLDP database changed or power up

12.2 Vendor specific object classes

The LioN-X and LioN-Xlight EtherNet/IP variants support the following vendor specific object classes:

Object Class	Instances
General Settings Object (0xA0)	0, 1
Channel Settings Object (0xA1)	0, 1 .. 16
IO-Link Diagnosis Settings Object (0xA2)	0, 1
IO-Link Port Settings Object (0xA3)	0, 1 .. n*
IO-Link Failsafe Parameter Object (0xA4)	0, 1 .. n*
IO-Link Device Parameter Object (0xA5)	0, 1 .. n*

\*) The available instances depend on the number of IO-Link ports of the device variant. Up to 8 IO-Link ports and instances are supported.

12.2.1 General Settings Object (0xA0)

Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.



**Instance attribute (Instance 1)**

Attribute	Name	Access	Data type	Description
2	Force Mode Lock	Get, Set	BOOL	0: Disable 1: Enable
3	Web Interface Lock	Get, Set	BOOL	0: Disable 1: Enable
4	Digital Output Control	Get, Set	BOOL	0: DO Channel Control 1: IO-Link Control
5	Report UL/UAux Supply Voltage Fault	Get, Set	BOOL	0: Disable 1: Enable
6	Report DO Fault without UL/UAux	Get, Set	BOOL	0: Disable 1: Enable
7..24	Reserved			
25	CIP object configuration lock	Get, Set	BOOL	0: Disable 1: Enable
26	External configuration lock	Get, Set	BOOL	0: Disable 1: Enable
27..31	Reserved			
32	IO Mapping Mode	Get, Set	SINT	0: Default Assignment 1: Byte Swap 2: LSB Ch.A - MSB Ch.B 3: LSB Ch.B - MSB Ch.A 4: Free IO Mapping

## 12.2.2 Channel Settings Object (0xA1)

### Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.

### Instance attribute (Instance 1 .. 16)

Attribute	Name	Access	Data type	Description
1	IO Mapping	Get, Set	SINT	0 .. 15: Bit number of 16 channel process data 16: Inactive
2	DO Surveillance Timeout	Get, Set	INT	0 .. 255
3	DO Failsafe	Get, Set	SINT	0: Set Low 1: Set High 2: Hold Last
4	DO Restart Mode	Get, Set	SINT	0: Disable 1: Enable
5*	DO Switch Mode	Get, Set	SINT	0: Push-Pull ( $U_S$ , 0.5 A) 1: High-Side ( $U_L$ , 0.5 A) 2: High-Side ( $U_L$ , 1.0 A) 3: High-Side ( $U_L$ , 1.5 A) 4: High-Side ( $U_L$ , 2.0 A) 5: High-Side ( $U_L$ , 2.0 A max)
6	DI Logic	Get, Set	SINT	0: Normally Open 1: Normally Close
7	DI Filter	Get, Set	SINT	0: Disabled 1: 1 ms 2: 2 ms 3: 3 ms 4: 6 ms 5: 10 ms 6: 15 ms
8..9	Reserved			
10	Channel Mode	Get, Set	SINT	0: Inactive 1: Digital Output 2: Digital Input 3: IO-Link 4: Auxiliary Power <b>The supported Channel Mode depends on the device variant.</b>

\* Not available for Lion-Xlight IO-Link Master variants

### 12.2.3 IO-Link Diagnosis Settings Object (0xA2)

#### Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.

#### Instance attribute (Instance 1)

Attribute	Name	Access	Data type	Description
1	IO-Link Master Diagnosis	Get, Set	BOOL	0: Disable 1: Enable
2	IO-Link Device Error	Get, Set	BOOL	0: Disable 1: Enable
3	IO-Link Device Warning	Get, Set	BOOL	0: Disable 1: Enable
4	IO-Link Device Notification	Get, Set	BOOL	0: Disable 1: Enable
5 .. 12	IO-Link Device Diagnosis Port 1 .. 8	Get, Set	BOOL	0: Disable 1: Enable

### 12.2.4 IO-Link Port Settings Object (0xA3)

#### Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.

#### Instance attribute (Instance 1 .. n\*)

<sup>\*)</sup> n = number of IO-Link ports supported by the device variant

Attribute	Name	Access	Data type	Description
1	Output Data Size	Get, Set	SINT	0: No data 1: 2 Byte 2: 4 Byte 3: 8 Byte 4: 16 Byte 5: 32 Byte Only settable when no connection is established.
2	Input Data Size	Get, Set	SINT	0: No data 1: 2 Byte 2: 4 Byte 3: 8 Byte 4: 16 Byte 5: 32 Byte Only settable when no connection is established.

Attribute	Name	Access	Data type	Description
3	Input Data Extension	Get, Set	SINT	0: No Data 1: Extended Status 2: Events 3: Extended Status + Events Only settable when no connection is established.
4	Output Data Swapping Mode	Get, Set	SINT	0: Raw IO-Link Data 1 .. 16: 1 .. 16 WORD 17 .. 24: 1 .. 8 DWORD Only settable when no connection is established.
5	Output Data Swapping Offset	Get, Set	SINT	0 .. 30 Byte Only settable when no connection is established.
6	Input Data Swapping Mode	Get, Set	SINT	0: Raw IO-Link Data 1 .. 16: 1 .. 16 WORD 17 .. 24: 1 .. 8 DWORD Only settable when no connection is established.
7	Input Data Swapping Offset	Get, Set	SINT	0 .. 30 Byte Only settable when no connection is established.
8	IOL Failsafe	Get, Set	SINT	0: Set Low 1: Set High 2: Hold Last 3: Replacement Value (transferred via IO-Link Failsafe Parameter Object) 4: IO-Link Master Command
9	Port Mode	Get, Set	SINT	0: Deactivated 1: Manual (with validation and backup config) 2: Autostart (no validation and backup config)

Attribute	Name	Access	Data type	Description
10	Validation and Backup	Get, Set	SINT	0: No device check and clear (no data storage) 1: Type compatible V1.0 device (no data storage) 2: Type compatible V1.1 device (no data storage) 3: Type compatible V1.1 device with Backup + Restore (Download + Upload) 4 Type compatible V1.1 device with Restore (Download Master to Device)
11	Vendor ID	Get, Set	DINT	0 .. 65535
12	Device ID	Get, Set	DINT	0 .. 16777215
13	Cycle Time	Get, Set	SINT	0: As fast as possible 1: 1.6 ms 2: 3.2 ms 3: 4.8 ms 4: 8.0 ms 5: 20.8 ms 6: 40.0 ms 7: 80.0 ms 8: 120.0 ms

12.2.5 IO-Link Failsafe Parameter Object (0xA4)

Supported services:

Get Attribute Single (0x0E)

Set Attribute Single (0x10)

Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.

Instance attribute (Instance 1 .. n\*)

\*) n = number of IO-Link ports supported by the device variant

Attribute	Name	Access	Data type	Description
1	Failsafe value of IO-Link port	Get, Set	Array of Bytes	Depends on configured process data lengths, content must consider possible swapping configuration (failsafe value format must match output data format)



### 12.2.6 IO-Link Device Parameter Object (0xA5)

#### Supported services:

##### Instance 0

Get Attribute Single (0x0E)

##### Instance 1 .. n\*

Get ISDU data (0x4B)

Set ISDU data (0x4C)

\*) n = number of IO-Link ports supported by the device variant

#### Class attribute (Instance 0)

Attribute	Name	Access	Data type	Description
1	Revision	Get	UINT	Revision of this object
2	Max. Instance	Get	UINT	Maximum instance number of an object currently created in this class level of the device.

#### Instance attribute (Instance 1 .. n\*)

\*) n = number of IO-Link ports supported by the device variant

Attribute	Name	Access	Data type	Description
1	ISDU data of IO-Link port	Get, Set	Array of Bytes	ISDU data of an IO-Link Device can be read using the Get ISDU data service and written using the Set ISDU data service (see description of these services below).

Get ISDU data

The Index and Subindex shall be set in the source data. The different protocol data formats between EtherNet/IP (little-endian) and IO-Link (big-endian) must be considered. The data length of the response depends on the data type of the IO-Link Device.

Protocol	EtherNet/IP		
Byte	0	1	2
Data type	UINT		USINT
Endianness	LSB	MSB	–
Content	Index		Subindex

*Table 17: Source*

Protocol	IO-Link		
Byte	0	...	n
Data type	Depends on device data type		
Endianness	MSB	...	LSB
Content	Data or Error occurred (max. 232 Bytes)		

*Table 18: Destination*

Set ISDU data

The Index, Subindex and IO-Link data shall be set in the source data. The data length of the request depends on the data type of the IO-Link Device. The different protocol data formats between EtherNet/IP (little-endian) and IO-Link (big-endian) must be considered. There is only data in the response of the IO-Link Device available if an Error occurred.

Protocol	EtherNet/IP			IO-Link		
Byte	0	1	2	3	...	n
Data type	UINT		USINT	Depends on device data type		
Endianness	LSB	MSB	–	MSB	...	LSB
Content	Index		Subindex	Data (max. 232 Bytes)		

*Table 19: Source*

Protocol	IO-Link		
Byte	0	...	n
Data type	Depends on device data type		
Endianness	MSB	...	LSB
Content	Error if occurred (max. 232 Bytes)		

*Table 20: Destination*

If the read or write request is not successful (CIP response status is unequal "0"), the following response format of 4 bytes is available:

Name	Data type	Error code description	Error code
IO-Link Master Error	UINT	Service not available	1
		Port blocked	2
		Timeout	3
		Invalid index	4
		Invalid sub-index	5
		Wrong port	6
		Wrong port function	7
		Invalid length	8
		ISDU not supported	9
IO-Link Device Error	USINT	Refer to IO-Link specification	–
IO-Link Device Additional Error	USINT	Refer to IO-Link specification	–

In [Get/Set ISDU data](#) on page 143, you find an example for Rockwell Automation Studio 5000®.

## 12.3 Message configuration in Rockwell Automation Studio 5000®

Attributes of CIP object classes can be handled in Rockwell Automation Studio 5000® by the *Message instruction*. This requires the selection of the proper message and service type with its respective service code.

The channels as in the *Channel Settings Object* are each assigned in ascending order to an instance ID.

### Assignment of the channels:

Channel 0	Port X1.ChA	CIP object instance 1
Channel 1	Port X1.ChB	CIP object instance 2
[...]	[...]	[...]
Channel 14	Port X8.ChA	CIP object instance 15
Channel 15	Port X8.ChB	CIP object instance 16

The IO-Link ports as in the *IO-Link Port Settings Object*, *IO-Link Failsafe Parameter Object* and *IO-Link Device Parameter Object* are each assigned in ascending order to an instance ID.

### Assignment of the IO-Link ports:

IO-Link port 1	Port X1.ChA	CIP object instance 1
[...]	[...]	[...]
IO-Link port 8	Port X8.ChA	CIP object instance 8

### 12.3.1 Get/Set attribute single

Every attribute with exception of the IO-Link Device Parameter Object Instance 1..n can be accessed using the *Get/Set attribute single* service by the CIP object class ID, the instance ID and attribute ID. The respective data is described in the previous chapters.

The following image shows an example of how to set *Force Mode Lock* (Attribute 2) of the *General Settings Object (0xA0)* with the *Message instruction*:

Message Configuration - MSG\_CIP\_Object\_Attribute

Configuration Communication Tag

Message Type: CIP Generic

Service Type: Set Attribute Single Source: MOD01\_ForceModel

Source Length: 1 (Bytes)

Service Code: 10 (Hex) Class: A0 (Hex) Destination Element: New Tag...

Instance: 1 Attribute: 2 (Hex)

☐ Enable
 ☐ Enable Waiting
 ☐ Start
 ☐ Done
 Done Length: 0

☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path: MOD01\_IOL\_Master Error Text:

OK Abbrechen Übernehmen Hilfe

### 12.3.2 Get/Set ISDU data

The IO-Link Device Parameter Object Instance 1..n can be accessed using the vendor specific *Get/Set ISDU data* service by the CIP object class ID, the instance ID and attribute ID. The Index and Subindex shall be set in the source data. For the *Set ISDU data* service, the IO-Link data must be appended. The different protocol data formats between EtherNet/IP (little-endian) and IO-Link (big-endian) must be considered. The respective data is described in the previous chapters.

The following figure shows an example of how to get an IO-Link Device parameter using the *Get ISDU data (0x4B)* service of the *IO-Link Device Parameter Object (0xA5)* with the *Message instruction*:

Message Configuration - MSG\_MOD02\_IOL\_ISDU\_DATA\_READ

Configuration Communication Tag

Message Type: CIP Generic

Service Type: Custom

Service Code: 4b (Hex) Class: a5 (Hex) Instance: 1 Attribute: 1 (Hex)

Source: MSG\_MOD02\_IOL\_IS Source Length: 3 (Bytes) Destination Element: MSG\_MOD02\_IOL\_IS

New Tag...

☐ Enable ☐ Enable Waiting ☐ Start ☒ Done Done Length: 2

☐ Error Code: Extended Error Code: ☐ Timed Out

Error Path: MOD02\_IOL\_XP Error Text:

OK Abbrechen Übernehmen Hilfe

The Index (0x003C) and Subindex (0x01) of the IO-Link device parameter are set in the little-endian format of EtherNet/IP in the source data:

[-] MSG_MOD02_IOL_ISDU_SRC_READ	{...}	{...}	Hex	SINT[8]
+ MSG_MOD02_IOL_ISDU_SRC_READ[0]	16#3c		Hex	SINT
+ MSG_MOD02_IOL_ISDU_SRC_READ[1]	16#00		Hex	SINT
+ MSG_MOD02_IOL_ISDU_SRC_READ[2]	16#01		Hex	SINT

The response data of the IO-Link device can be found in the destination element. In the following example, the received value is of the type UINT in the big-endian format of IO-Link (0x0546 = 1350):

[-] MSG_MOD02_IOL_ISDU_DST_READ	{...}	{...}	Hex	SINT[8]
+ MSG_MOD02_IOL_ISDU_DST_READ[0]	16#05		Hex	SINT
+ MSG_MOD02_IOL_ISDU_DST_READ[1]	16#46		Hex	SINT



## 13 Diagnostics processing

### 13.1 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 diagnosis is generated. The IO-Link specification requires at least 20 V at the L+ (pin1) output supply of the I/O ports. At least 21 V of  $U_S$  supply voltage for the IO-Link Master are required to minimize the risk of internal voltage drops in the IO-Link Master.

The green  $U_S$  indicator is off.

The error diagnosis 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 21 V DC from the perspective of the system power supply.

The following diagnostics are generated in the producing data image:

General diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	IME	FME	DTO	DTU	SCA	SCS	LVA	<b>LVS</b>
	Byte 1	0	0	0	0	IDN	IDW	IDE	IVE

**LVS**

Low Voltage System/Sensor Supply

## 13.2 Error of the auxiliary/actuator power supply

The voltage value for the incoming auxiliary/actuator power supply is also monitored globally. If *Report  $U_L/U_{AUX}$  Supply Voltage Fault* is enabled, an error message is generated when the voltage drops below approx. 18 V or exceeds approx. 30 V. The  $U_L/U_{AUX}$  indicator shows red.

The following diagnostics are generated in the *producing data image*:

General diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	IME	FME	DTO	DTU	SCA	SCS	LVA	LVS
	Byte 1	0	0	0	IDN	IDW	IDE	IVE	0

### LVA

Low Voltage Actuator Supply

If output channels are set to *High State* and *Report DO Fault without  $U_L/U_{AUX}$* , additional error diagnostics, caused by the voltage failure, are generated on the channels

The following diagnostics are generated in the *producing data image*:

Actuator/ $U_{AUX}$ diagnostics	Bit	7	6	5	4	3	2	1	0
Channel number (fix)	Byte 0	8	7	6	5	4	3	2	1
	Byte 1	16	15	14	13	12	11	10	9

### 1 .. 16

Actuator/ $U_L/U_{AUX}$  channel error on channel 1 .. 16

If *Report  $U_L/U_{AUX}$  Supply Voltage Fault* is disabled, no  $U_L/U_{AUX}$  or channel diagnostics appear.

## 13.3 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 diagnostics in the producing data image are generated:

General diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	IME	FME	DTO	DTU	SCA	<b>SCS</b>	LVA	LVS
	Byte 1	0	0	0	0	IDN	IDW	IDE	IVE

### SCS

Short Circuit Sensor

Sensor diagnostics	Bit	7	6	5	4	3	2	1	0
Port number	Byte 0	<b>X8</b>	<b>X7</b>	<b>X6</b>	<b>X5</b>	<b>X4</b>	<b>X3</b>	<b>X2</b>	<b>X1</b>
	Byte 1	0	0	0	0	0	0	0	0

### X1 .. 8

Sensor Short Circuit on Port X1 .. X8

## 13.4 Overload/short circuit of the digital outputs

In case of an overload or a short circuit of an output channel, the following channel-specific diagnostics are generated in the *producing data image*:

General diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	IME	FME	DTO	DTU	<b>SCA</b>	SCS	LVA	LVS
	Byte 1	0	0	0	0	IDN	IDW	IDE	IVE

### SCA

Short Circuit Actuator/ $U_L/U_{AUX}$

Actuator/ $U_{AUX}$ diagnostics	Bit	7	6	5	4	3	2	1	0
Channel number (fix)	Byte 0	8	7	6	5	4	3	2	1
	Byte 1	16	15	14	13	12	11	10	9

### 1 .. 16

Actuator/ $U_L/U_{AUX}$  channel error on channel 1 .. 16

A channel error is determined by comparing the target value set by a controller and the physical 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 is set by the "Surveillance-Timeout" parameter via the configuration of the device. 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 diagnosis is typically 5 ms.

13.5 IO-Link COM 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, the following diagnostics are generated in the *producing data image*:

IO-Link diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	ICE8	ICE7	ICE6	ICE5	ICE4	ICE3	ICE2	ICE1
	Byte 1	0	0	0	0	0	0	0	0
	Byte 2	IVE8	IVE7	IVE6	IVE5	IVE4	IVE3	IVE2	IVE1
	Byte 3	IDE8	IDE7	IDE6	IDE5	IDE4	IDE3	IDE2	IDE1
	Byte 4	IDW8	IDW7	IDW6	IDW5	IDW4	IDW3	IDW2	IDW1
	Byte 5	IDN8	IDN7	IDN6	IDN5	IDN4	IDN3	IDN2	IDN1

ICE1 .. 8

IO-Link Port COM Error (device missing, broken wire, short circuit)

## 13.6 IO-Link validation error

If an IO-Link Device is exchanged by a new device, the validation is configured. The vendor ID and/or device ID do not match the data of the device and the following diagnostics are generated in the *producing data image*:

IO-Link diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	ICE8	ICE7	ICE6	ICE5	ICE4	ICE3	ICE2	ICE1
	Byte 1	0	0	0	0	0	0	0	0
	Byte 2	<b>IVE8</b>	<b>IVE7</b>	<b>IVE6</b>	<b>IVE5</b>	<b>IVE4</b>	<b>IVE3</b>	<b>IVE2</b>	<b>IVE1</b>
	Byte 3	IDE8	IDE7	IDE6	IDE5	IDE4	IDE3	IDE2	IDE1
	Byte 4	IDW8	IDW7	IDW6	IDW5	IDW4	IDW3	IDW2	IDW1
	Byte 5	IDN8	IDN7	IDN6	IDN5	IDN4	IDN3	IDN2	IDN1

### IVE1 .. 8

#### IO-Link Port Validation Error

If extended status data is enabled by the configuration of an IO-Link port, the vendor ID and device ID are additionally transferred in the *producing data image*.

## 13.7 IO-Link device diagnostics

The diagnostics of an IO-Link Device come in three different levels: Error, Warning or Notification. The following diagnostics are generated in the *producing data image*:

IO-Link diagnostics	Bit	7	6	5	4	3	2	1	0
General Bit	Byte 0	ICE8	ICE7	ICE6	ICE5	ICE4	ICE3	ICE2	ICE1
	Byte 1	0	0	0	0	0	0	0	0
	Byte 2	IVE8	IVE7	IVE6	IVE5	IVE4	IVE3	IVE2	IVE1
	Byte 3	IDE8	IDE7	IDE6	IDE5	IDE4	IDE3	IDE2	IDE1
	Byte 4	IDW8	IDW7	IDW6	IDW5	IDW4	IDW3	IDW2	IDW1
	Byte 5	IDN8	IDN7	IDN6	IDN5	IDN4	IDN3	IDN2	IDN1

**IDE1 .. 8**

IO-Link Port Device Error

**IDW1 .. 8**

IO-Link Port Device Warning

**IDN1 .. 8**

IO-Link Port Device Notification

If IO-Link event data is enabled by the configuration of an IO-Link port the device additionally reports event codes in the *producing data image*. Use the IO-Link Device documentation to interpret the error message.



## 14 IIoT functionality

The LioN-X variants offer a number of new interfaces and functions for the optimal integration into existing or future IIoT (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 IIoT interfaces, which enable new communication channels besides the PLC. The communication is performed via IIoT-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 IIoT protocols. This allows you to manage all modification options for the device settings via personalized user authorizations.

All IIoT 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 IIoT protocols.



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

## 14.1 MQTT

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

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

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.

### 14.1.1 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/HTTPS request. For more information see chapter [MQTT configuration - Quick start guide](#) on page 169.

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 / <b>false</b>
broker	string	IP address of the MQTT Broker	<b>"192.168.1.1"</b>
login	string	Username for MQTT Broker	"admin" (Default: <b>null</b> )
password	string	Password for MQTT Broker	"private" (Default: <b>null</b> )
port	number	Broker port	<b>1883</b>
base-topic	string	Base topic	"iomodule_[mac]" (Default: <b>"lionx"</b> )
will-enable	boolean	If true, the device provides a last will message to the broker	true / <b>false</b>
will-topic	string	The topic for the last will message.	(Default: <b>null</b> )
auto-publish	boolean	If true, all enabled domains will be published automatically in the specified interval.	<b>true</b> / false
publish-interval	number	The publish interval in ms if auto-publish is enabled. Minimum is 250 ms.	<b>2000</b>
publish-identity	boolean	If true, all identity domain data will be published	<b>true</b> / false
publish-config	boolean	If true, all config domain data will be published	<b>true</b> / false
publish-status	boolean	If true, all status domain data will be published	<b>true</b> / false
publish-process	boolean	If true, all process domain data will be published	<b>true</b> / false
publish-devices	boolean	If true, all IO-Link Device domain data will be published	true / <b>false</b>
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 / <b>false</b>
force-allowed	boolean	If true, the device accepts force commands via MQTT.	true / <b>false</b>
reset-allowed	boolean	If true, the device accepts restart and factory reset commands via MQTT.	true / <b>false</b>
config-allowed	boolean	If true, the device accepts configuration changes via MQTT.	true / <b>false</b>

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

Table 21: 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 157.

### 14.1.2 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 ( \* , # ).

#### 14.1.2.1 Base topic

For all LioN-X 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 22: Base topic variables](#) on page 157.

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 22: 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 23: 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 24: 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 25: Use case examples*

14.1.2.2 Publish topic

Overview of all publish JSON data for the defined topics:

Key	Data type
product_name	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 26: Identity/gateway



Key	Data type	Range	Default value	Remarks
fieldbus_protocol	json_string	PROFINET, EtherNet/IP, EtherCAT®		
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 27: 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		
simulation_active_diag	json_boolean	true / false		
us_voltage	json_integer	0 .. 32		in Volts
ul_voltage	json_integer	0 .. 32		in Volts
forcemode_enabled	json_boolean	true / false		

Table 28: Status/gateway

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

Table 29: 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 30: Identity/port/1 .. 8*

Key	Data type	Range	Default value	Remarks
port	json_integer	1 .. 8		
direction_cha	json_string	input/output input output		
restart_mode_cha	json_string	Manual Auto		
restart_mode_chb	json_string	Manual Auto		
input_polarity_cha	json_string	NO NC		
input_polarity_chb	json_string	NO NC		
input_filter_cha	json_integer			ms
input_filter_chb	json_integer			ms
do_auto_restart_cha	json_boolean	true / false		
do_auto_restart_chb	json_boolean	true / false		

Table 31: 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		
current_cha	json_integer			mA
current_chb	json_integer			mA
current_pin1	json_integer			mA

Table 32: Status/port/1 .. 8

### 14.1.2.3 Command topic (MQTT Subscribe)

The main purpose of MQTT is to publish data from the device to a broker. This data can then be received by any subscriber who is interested in this data. But also the other way round is possible. The device can subscribe to a topic on the broker and is then able to receive data. This data can contain configuration or forcing data. This allows the user to fully control a device via MQTT only, without using other ways of communication like Web or REST.

If the configuration allows commands in general, the device subscribes to special Command topics on which it can receive commands from other MQTT clients. The Command topic is based upon the Base topic. It always has the following form:

```
[base-topic]/command
```

After the Command topic, there are fixed topics for different writeable objects. The data format of the MQTT payload is always JSON. It is possible to set only a subset of the possible objects and fields.

#### [...]/forcing

Use the Command topic `[base-topic]/command/forcing` for *Force object* data. The *Force object* can contain any of the following properties:

Property	Data type	Example values	Remarks
forcemode	boolean	true / false	Forcing Authority: on/off
digital	array ( <a href="#">Table 34: Force object: Digital</a> on page 166)		
iol	array ( <a href="#">Table 35: Force object: IOL (IO-Link devices only)</a> on page 166)		

Table 33: Force object properties

For the *Force object* properties `digital` and `iol`, there are several value specifications arrayed:

Property	Data type	Example values	Remarks
port	integer	1, 2, 5	
channel	string	"a", "b"	
force_dir	string	"out", "in", "clear"	
force_value	integer	0, 1	

Table 34: Force object: Digital

Property	Data type	Example values	Remarks
port	integer	0, 1, 5	
output	array[integer]	[55, 88, 120]	
input	array[integer]		Input-Simulation

Table 35: Force object: IOL (IO-Link devices only)

[...]/config

Use the Command topic `[base-topic]/command/config` for *Config object* data. The *Config object* can contain any of the following properties:

Property	Data type	Example values	Remarks
portmode	array (Table 37: Config object: Portmode on page 167)		
ip_address	string	"192.168.1.5"	
subnet_mask	string	"255.255.255.0"	
gateway	string	"192.168.1.100"	

Table 36: Config object properties

For the *Conig object* property `portmode`, there are several value specifications arrayed:

Property	Data type	Example values	Remarks
port	integer	2	
channelA*	string	"dio", "di", "do", "iol", "off"	
channelB*	string	"dio", "di", "do", "iol", "off", "aux"	
inlogicA	string	"no", "nc"	
inlogicB	string	"no", "nc"	
filterA	integer	3	input filter in ms
filterB	integer	3	input filter in ms
autorestartA	boolean		
autorestartB	boolean		
iolValidation	integer	0 = NoCheck 1 = Type 1.0 2 = Type 1.1 3 = Type 1.1 BR 4 = Type 1.1 RES	
iolDeviceID	integer		for validation
iolVendorID	integer		for validation

*Table 37: Config object: Portmode*

\*channelA = Pin 4, channelB = Pin 2

### [...]/reset

Use the Command topic [base-topic]/command/reset for *Reset object* data about restart and factory reset issues. The *Reset object* can contain any of the following properties:

Property	Data type	Example values	Remarks
factory_reset	boolean	true / false	
system_reset	boolean	true / false	

Table 38: Reset object properties

### [...]/publish

Use the Command topic [base-topic]/command/publish for *Publish object* data.

Trigger publish of all topics manually (can be used when auto publish is off or long interval is set).



### 14.1.3 MQTT configuration - Quick start guide



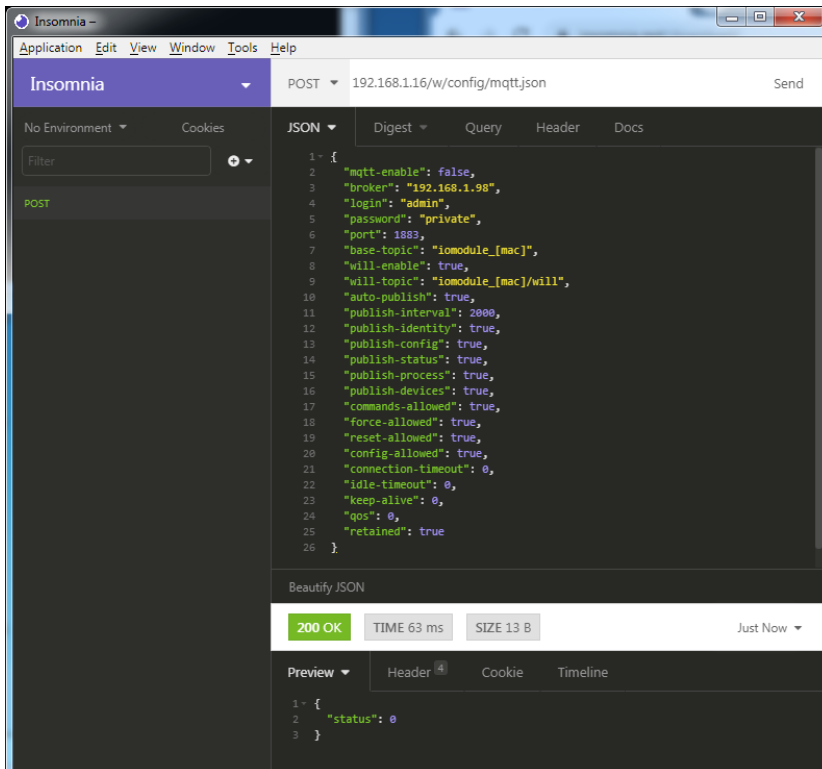
**Attention:** Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

#### 14.1.3.1 MQTT configuration via JSON

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

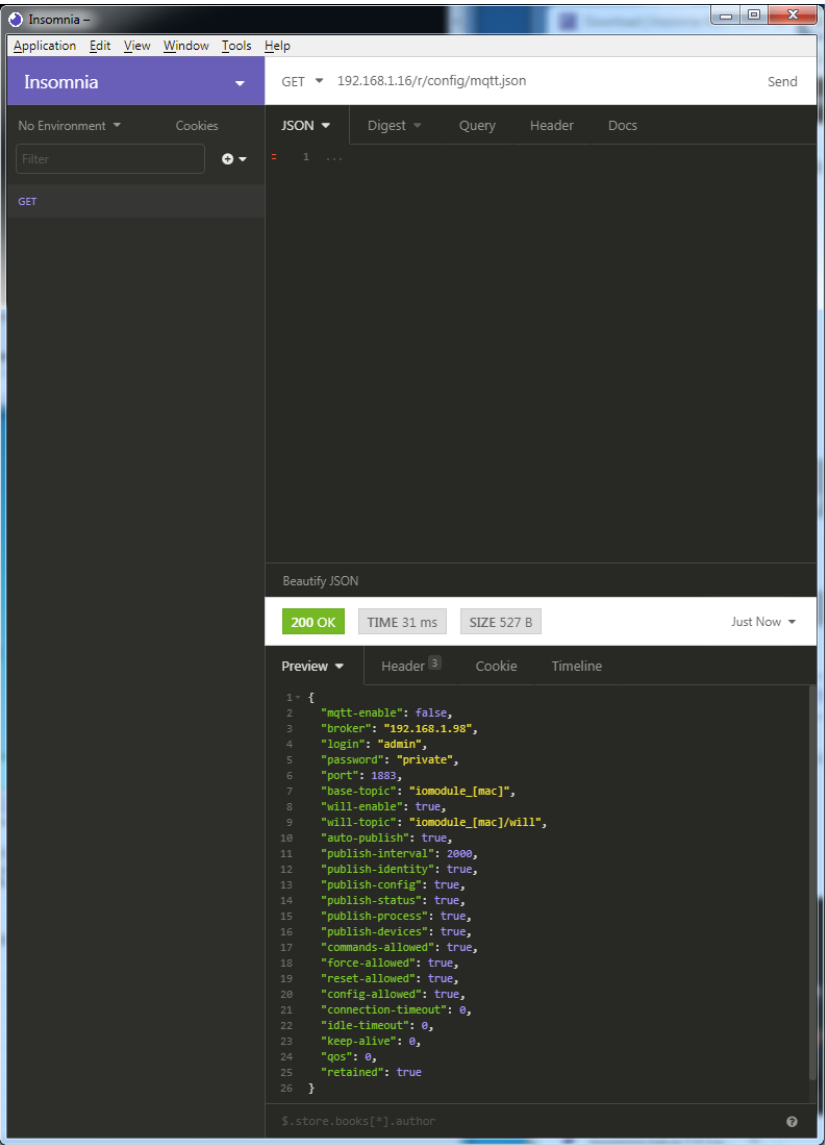
2. Configure MQTT:

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



3. Read MQTT:

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



## 14.2 OPC UA

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

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

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](https://catalog.belden.com) or directly from [io-link.com](https://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 39: Non-supported OPC UA features according to the IO-Link Companion Specification

14.2.1 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/HTTPS request. For more information see [OPC UA configuration - Quick start guide](#) on page 175.

The configuration URL is:

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

The configuration can also read back as a JSON file:

```
http://[ip-address]/r/config/opcua.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 changed applies 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, <b>4840</b> , 0xFFFF
opcua-enable	boolean	Master switch for the OPC UA server.	true / <b>false</b>
anon-allowed	boolean	If true, anonymous login is allowed.	<b>true</b> / false
commands-allowed	boolean	Master switch for OPC UA commands. If false there will be no writeable OPC UA objects.	true / <b>false</b>
force-allowed	boolean	If true, the device accepts force commands via OPC UA.	true / <b>false</b>
reset-allowed	boolean	If true, the device accepts restart and factory reset commands via OPC UA.	true / <b>false</b>
config-allowed	boolean	If true, the device accepts configuration changes via OPC UA.	true / <b>false</b>

*Table 40: 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" }] }
```

## 14.2.2 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.

### 14.2.3 OPC UA configuration - Quick start guide

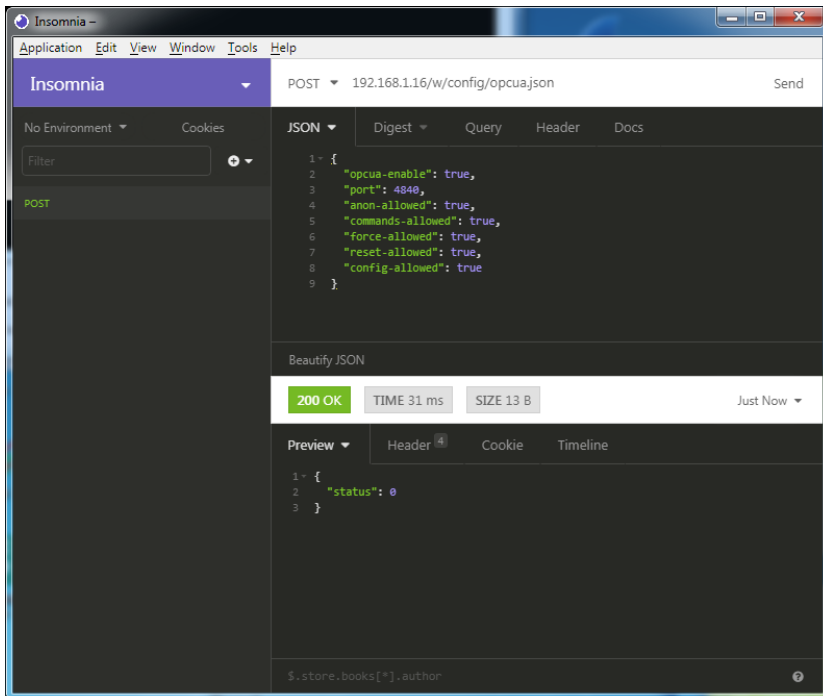
**i** **Attention:** Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

#### 14.2.3.1 OPC UA configuration via JSON

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

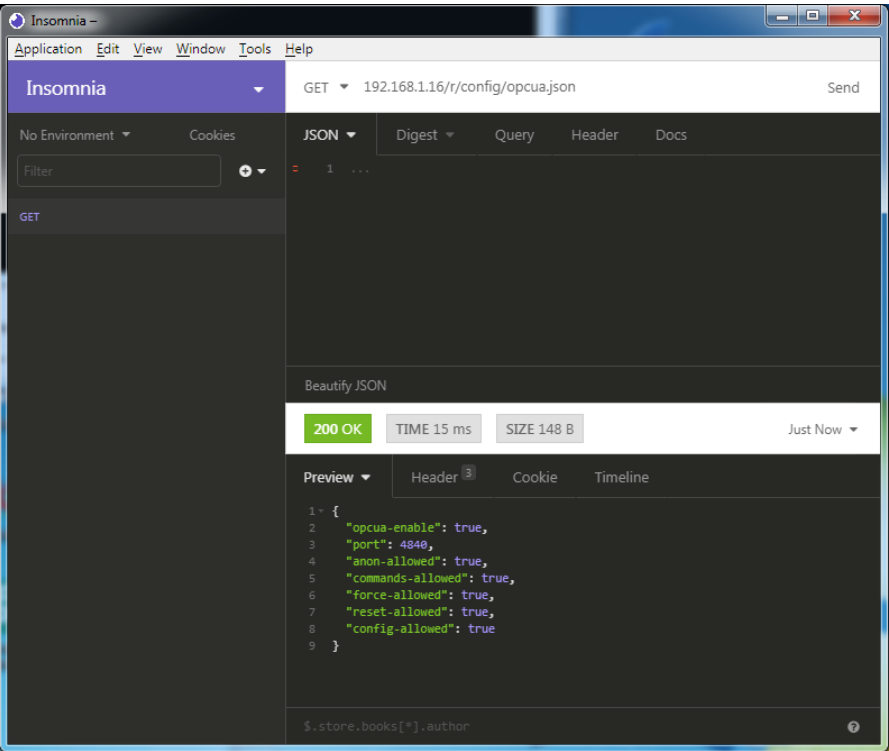
2. Configure OPC UA:

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



3. Read OPC UA:

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





## 14.3 REST API

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

For all LioN-X 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](https://catalog.belden.com) or directly from [io-link.com](https://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	Yes
	POST /datastorage	Yes
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	Not supported
	GET /parameters/{index}/subindices	Not supported
	GET /parameters/{parameterName}/subindices	Not supported
	GET /parameters/{index}/value	Not supported
	GET /parameters/{index}/subindices/{subindex}/value	Not supported
	GET /parameters/{parameterName}/value	Not supported
	GET /parameters/{parameterName}/subindices/{subParameterName}/value	Not supported
	POST /parameters/{index}/value	Not supported
	POST /parameters/{parameterName}/value	Not supported
	POST /parameters/{index}/subindices/{subindex}/value	Not supported
	POST /parameters/{parameterName}/subindices/{subParameterName}/value	Not supported
	POST /blockparametrization	Yes
	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 41: Support of REST API features according to the IO-Link specification*

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

### 14.3.1 Standard device information

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

14.3.2 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. <b>Element 0 = 1 Byte:</b> Port X1 Channel A to Port X4 Channel B <b>Element 0 = 1 Byte:</b> Port X5 Channel A to Port X8 Channel B	[128,3]
output	array of numbers (2)	Real State of digital outputs. <b>Element 0 =1 Byte:</b> Port X1 Channel A to port X4 Channel B <b>Element 0 = 1 Byte:</b> 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)	<div>Diagnostic information</div> <div> <b>Element 0 = 1 Byte:</b>            Bit 7: Internal module error (IME)            Bit 6: Forcemode active            Bit 3: Actuator short            Bit 2: Sensor short            Bit 1: U<sub>L</sub> fault            Bit 0: U<sub>S</sub> fault         </div> <div> <b>Element 1 = 1 Byte:</b>            Sensor short circuit ports X1 .. X8.         </div> <div> <b>Element 2 = 1 Byte:</b>            Actuator short circuit ports X1 Channel A to X4 Channel B         </div> <div> <b>Element 3 = 1 Byte:</b>            Actuator short circuit ports X5 Channel A to X8 Channel B         </div>	
fieldbus	FIELDBUS Object		
<b>FIELDBUS Object</b>			
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	
channels	Array of CHANNEL (16)	Basic information about all input/output channels	

Name	Data type	Description	Example
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	
<b>CHANNEL Object</b>			
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	
outputState	boolean	Output data bit to the physical output pin	

Name	Data type	Description	Example
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)	
<b>PORT Object</b>			
port_type	string	Textual representation of the IO-Link port type	
iolink_mode	number	<b>Current port mode:</b> 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	
iq_mode	number	Pin2 mode according to IOL specification	

Name	Data type	Description	Example
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	
<b>DIAG Object</b>			
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"
<b>DEVICE Object</b>		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	
<b>FORCING Object</b>		Forcing information of the device	
forcingActive	boolean	Force mode is currently active	



Name	Data type	Description	Example
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.	

14.3.3 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 42: 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 43: 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 44: 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 45: IOL object*

14.3.4 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.

14.3.4.1 Reading ISDU

Method:

POST

URL:

<ip>/r/isdu.json

Parameters:

port (0-7)

Example:

192.168.1.20/r/isdu.json?port=5

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 46: 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 47: 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 48: Read ISDU data object*

14.3.4.2 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 49: Write ISDU object

**Response:** Write ISDU response object

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

Table 50: 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 51: Write ISDU data object*



**Attention:** For LioN-X device variants with HTTPS feature, https:// must be used in front of <ip> for every REST API.

### 14.3.5 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
}
```

### 14.3.6 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
}
```



## 14.4 CoAP server

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

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

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/HTTPS 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.

### 14.4.1 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/HTTPS request. For more information see chapter [CoAP configuration - Quick start guide](#) on page 197.

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 / <b>false</b>
port	integer (0 to 65535)	Port of the CoAP server	<b>5683</b>

Table 52: 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" }] }
```

**14.4.2 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/opcua.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 53: REST API access via CoAP

### 14.4.3 CoAP configuration - Quick start guide

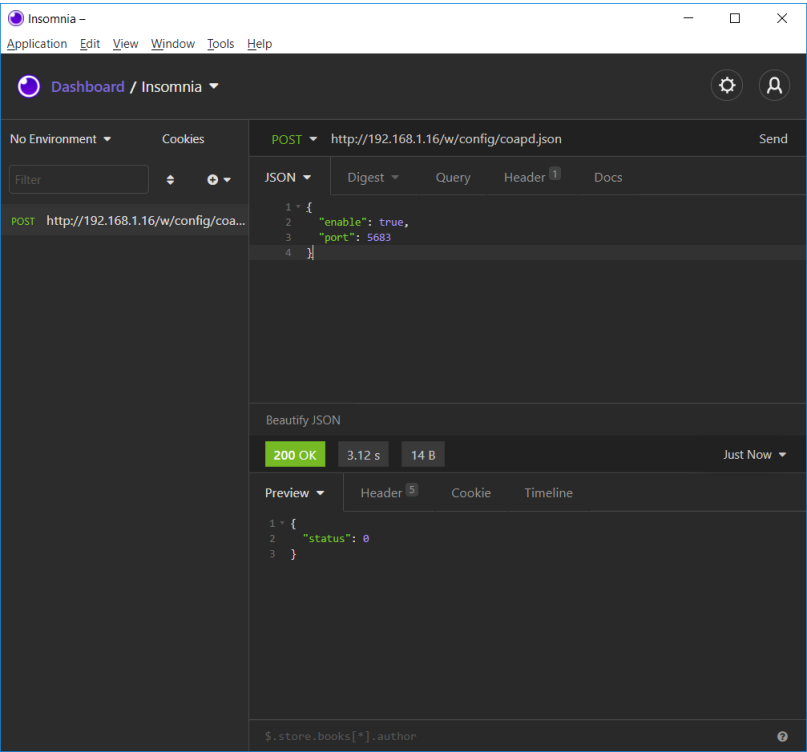
**i** **Attention:** Lumberg Automation™ is not responsible for any content of the referenced Web pages and provides no warranty for any functionality of the named third party software.

#### 14.4.3.1 CoAP configuration via JSON

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

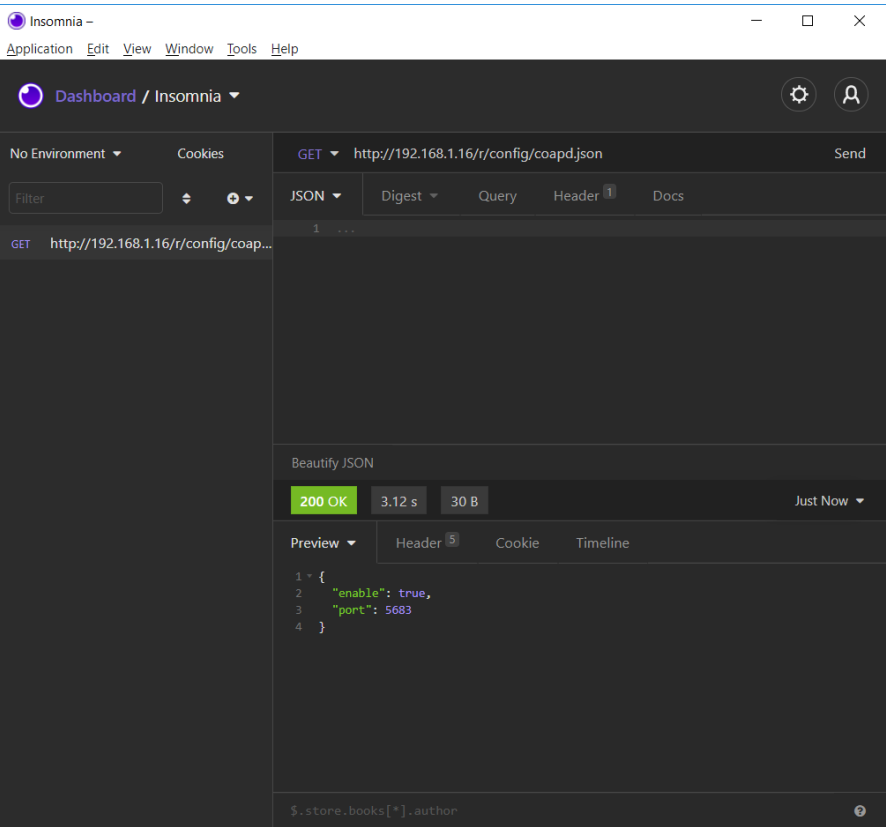
2. Configure CoAP:

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



3. Read CoAP configuration:

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



## 14.5 Syslog

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

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

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.

### 14.5.1 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/HTTPS request. For more information see chapter [Syslog configuration - Quick start guide](#) on page 202.

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 / <b>false</b>
global-severity	integer	<u>Severity level of Syslog client</u> 0 – Emergency 1 – Alert 2 – Critical <b>3 – Error</b> 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/ <b>3</b> /4/5/6/7
server-address	string (IP address)	IP address of the Syslog server	192.168.0.51 (Default: <b>null</b> )
server-port	integer (0 to 65535)	Server port of the Syslog server	<b>514</b>
server-severity	integer (0 to 7)	<u>Severity level of Syslog server</u> 0 – Emergency 1 – Alert 2 – Critical <b>3 – Error</b> 4 – Warning 5 – Notice 6 – Info 7 – Debug	0/1/2/ <b>3</b> /4/5/6/7

Table 54: 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" }] }
```

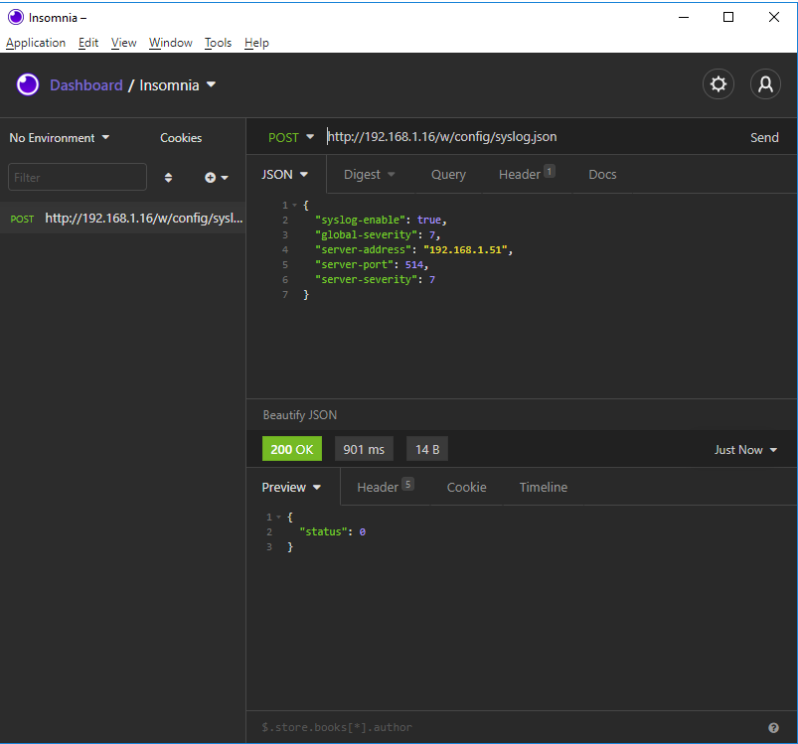
14.5.2 Syslog configuration - Quick start guide



**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.

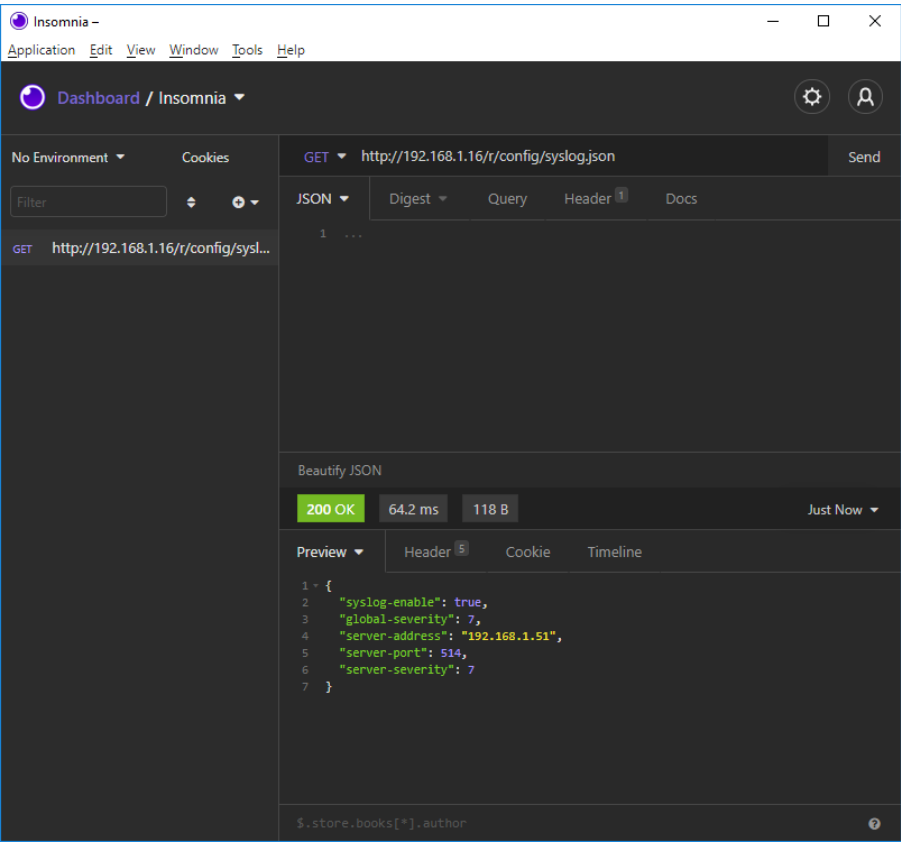
14.5.2.1 Syslog configuration via JSON

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



3. Read Syslog configuration:

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



## 14.6 Network Time Protocol (NTP)

The NTP function is **only** applicable for the following LioN-X variant:

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

The LioN-X multiprotocol variants provide an NTP client (version 3) which can connect with a configured NTP server and is able to synchronize the network time at a configurable interval.

NTP is a network protocol which uses UDP datagrams to send and receive timestamps in order to synchronize with a local clock. The NTP protocol RFC1305 is based on the Server-Client principle and exclusively supplies the synchronization with Coordinated Universal Time (UTC). (For more details on the used NTP standard, please refer to <https://datatracker.ietf.org/doc/html/rfc1305>.)

### 14.6.1 NTP configuration

In **delivery state**, the NTP client is **disabled**. The NTP client can be configured either using the Web interface or directly via a JSON object sent in an HTTP/HTTPS request. For more information see chapter [NTP configuration - Quick start guide](#) on page 206.

The configuration URL is:

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

The configuration can also read back as a JSON file:

```
http://[ip-address]/r/config/ntpc.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
NTP client state	boolean	Master switch for the NTP client	true / <b>false</b>
Server address	string	IP address of the NTP server	192.168.1.50
Server port	integer	Port of the NTP server	<b>123</b>
Update interval	integer	Interval at which the client will connect with the configured NTP server (see table row "Server address"). <b>Note: This value is in seconds.</b>	1/2/10/ <b>60</b>

*Table 55: NTP configuration*

### NTP 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 configuration element that caused the error, and of a field "Message" for the error message.


### Examples:

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

{ "status": 0 }

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

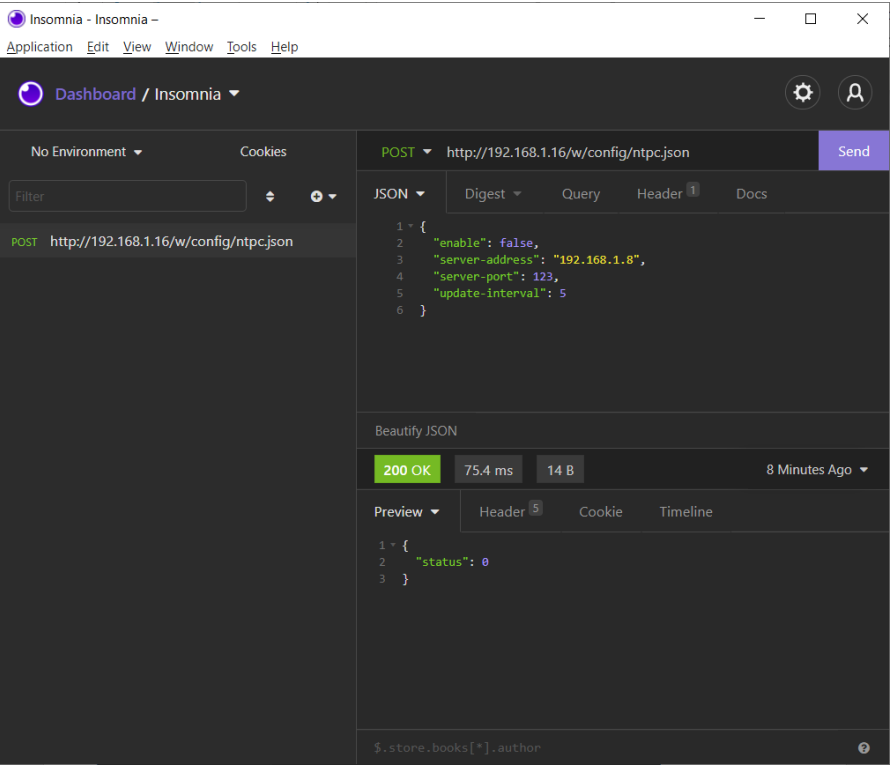
14.6.2 NTP configuration - Quick start guide



**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.

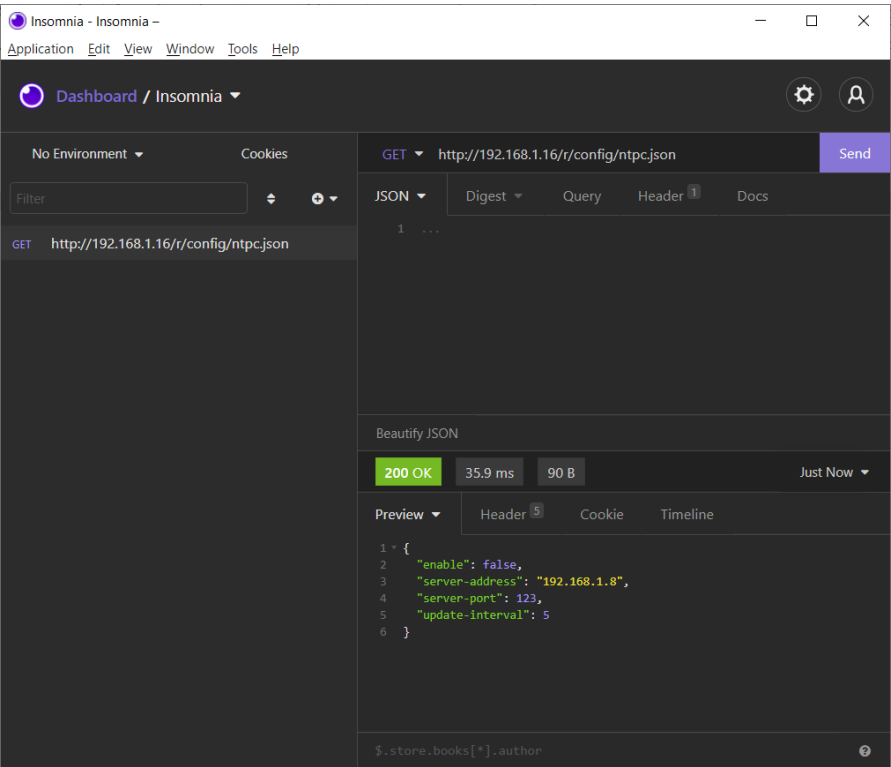
14.6.2.1 NTP configuration via JSON

1. Depending on your application case, download and install *Insomnia* or a comparable application: <https://insomnia.rest/download/>
2. Configure NTP:
- POST:** [IP-address]/w/config/ntpc.json



3. Read NTP configuration:

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



## 15 The integrated Web server

All device 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.

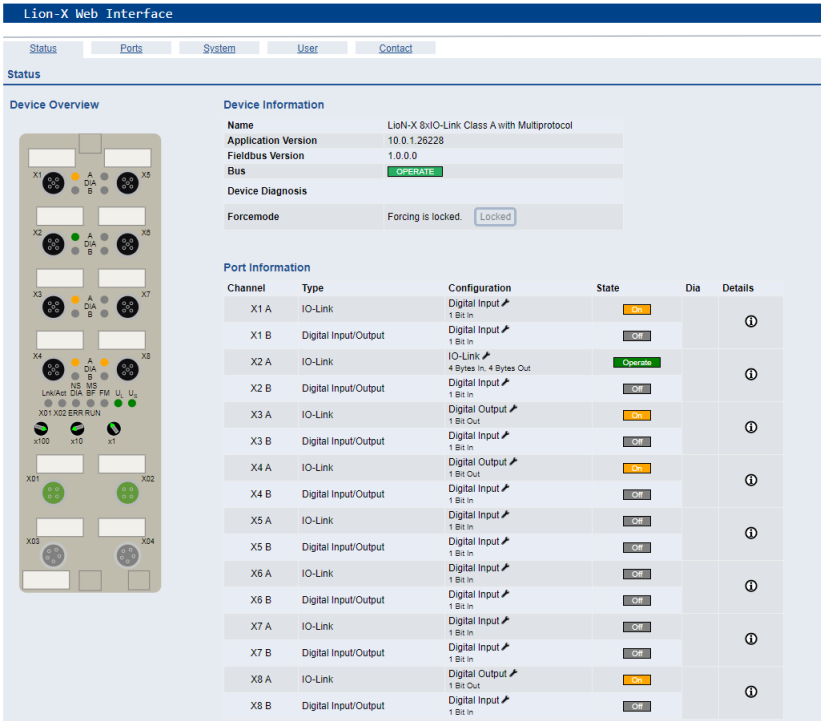
The Web interface provides an overview of the configuration and status of the device. It is also possible to use the Web interface to trigger a reboot, reset to the factory defaults, or perform a firmware update.

Enter "http://" or "https://" 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.



# 15.1 LioN-X 0980 XSL... variants

## 15.1.1 The Status page




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.

15.1.2 The Ports page

 **lumbergautomation**  
A **BELDEN** BRAND

Lion-X Web Interface

StatusPortsSystemUserContact

Port Details

Show details for port

☐ X1☒ X2☐ X3☐ X4☐ X5☐ X6☐ X7☐ X8

Port Information

Forcemode

Forcemode off

PortX2

TypeIO-Link

Dia

Port Diagnosis

No diagnosis

Pin 4 / Channel A

FunctionIO-Link

4 Bytes In, 4 Bytes Out

State

Operate

Pin 2 / Channel B

FunctionDigital Input

1 Bit In

State

Off

IO-Link Events

No events

IO-Link

Vendor ID362

Device ID3674114

Vendor NameBELDEN Deutschland GmbH

Vendor Textwww.beldensolutions.com

Product Name0960 IOL 381-001

Product ID934992002

Product TextLion-P IO-Link I/O-Hub, 16DI

Serial No.123

HW RevisionV1

FW RevisionV3.0.0.0

SpeedCOM3

Cycle time1000

Application Name (Tag)

\*\*\*

Set

Input Data

01 00 00 00

Hex

Output Data

00 00 00 00

Hex

Index: Subindex: 0

☒ Dec☐ Hex

Read

Write

System Command

Parameter Read/Write

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.

## 15.1.3 The System page



**LioN-X Web Interface**

Status Ports System User Contact

**System**

**General Information**

<b>Firmware</b>	
Application Version	11.1.6.28798
Fieldbus Version	1.2.0.0
IO Version	1.0.556.0
<b>Device</b>	
Name	LioN-X 8xIO-Link Class A with Multprotocol & https
Product ID	0980 XSL 3912-121-027D-01F
Ordering Number	935710001
Hardware	1.0
Serial Number	123456
Production Date	2020-12-24T12:00:00Z
<b>Ethernet</b>	
MAC Address	3C B9 A6 20 05 30
<b>Network</b>	
IP-Address	192.168.1.10
Subnetmask	255.255.255.0
Gateway	192.168.1.1
Source	Manual
<b>Fieldbus</b>	
Name	Ethernet/IP
State	ERROR

**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	Disabled
Broker	192.168.1.1
Port	1883
Base Topic	Ionix
Auto Publish	Yes
Publish Interval (ms)	2000
Publish Identity	Yes
Publish Config	Yes
Publish Status	Yes
Publish Process	Yes
Publish Devices	No
Will State	Disabled
Will Topic	
Listen for Commands	No
Process Forcing	No
Change Config	No
Device Reset	No
QoS	At most once
Opus state	Disabled
Port	4840
Anonymous login	Yes
Listen for Commands	No
Process Forcing	No
Change config	No
Device Reset	No
<b>Syslog</b>	
Syslog state	Disabled
Global severity	3
Server address	
Server port	514
Server severity	3
<b>CoAP</b>	
CoAP state	Disabled
Port	5683
<b>NTP</b>	
NTP client state	Disabled
Server address	0.0.0.0
Server port	123
Update interval	60

**HTTPS**

☒ HTTPS enable

Note: enabling or disabling HTTPS require a device restart.

[Apply & Restart](#)

**HTTPS Certificate Manager**

[Certificate Manager](#)

**IOOD**

[Manage IOODs](#)

**Restart device**

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

[Restart](#)

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

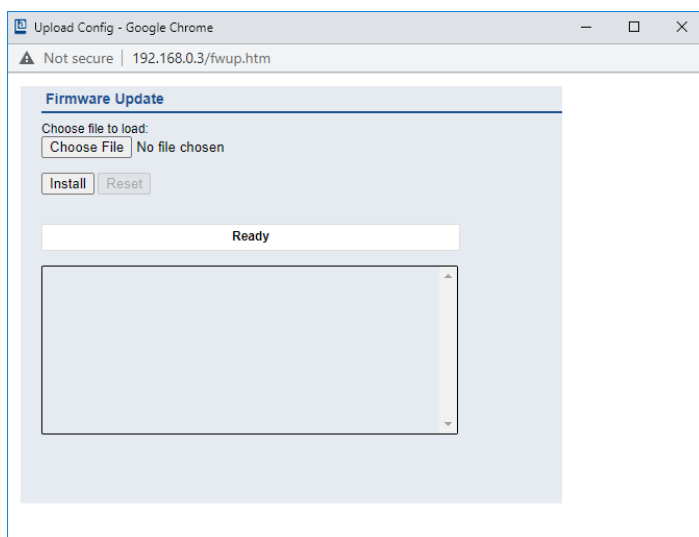
Use this parameter to change the current IP address of the module.

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.



### 15.1.3.1 HTTPS

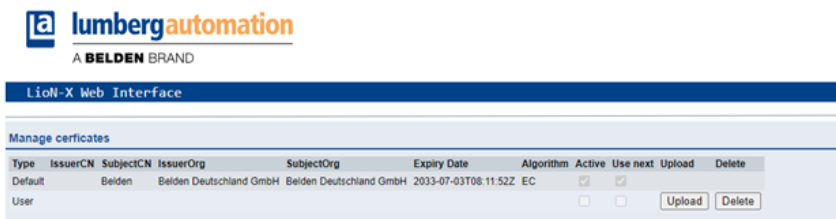
Https-based or http-based communication with the LioN-X Web server. If this option is selected, the communication with the LioN-X Web server will be secure and encryption-based.

### 15.1.3.2 HTTPS certificate manager

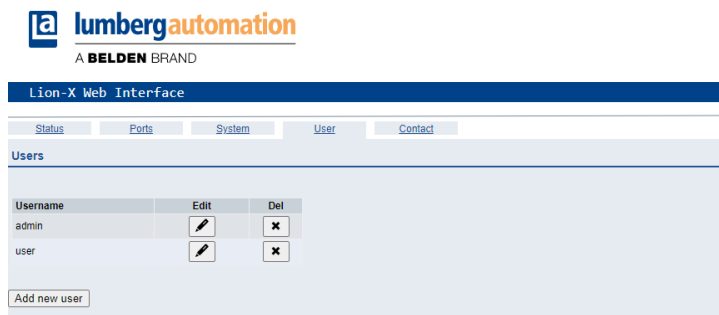
The HTTPS certificate manager shows a default certificate and the currently active certificate for the Web server. You are allowed to delete, upload and select new certificates. For an example of how to create and sign your own certificate with *Mako Server* by Real Time Logic LLC, refer to chapter [Certificate creation – example](#) on page 214.



**Attention:** It is not possible to delete the default certificate.



### 15.1.4 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

## 15.1.5 Certificate creation – example

### 1. Create certificate database:

In *Makro server* by Real Time Logic LLC, navigate to *Create Certificate Database*. Enter *DB Name*, select *Type* as “Elliptic Curve Certificate”, and select *SharkSSL Mode* as shown below.

### 2. Create certificate:

- Key Size: Any value from drop down list can be selected. “secp256r1” is recommended.
- Signature size: “sha256” → The higher the number in the encryption, the higher the security level of communication.

- ▶ Days: Enter the number of days you wish this certificate to be valid (e.g. "10950" for 10 years).
- ▶ Country name: "DE" ("DE" stands for Germany. For other countries, please refer to <https://www.ssl.com/country-codes/>).
- ▶ State or Province: Enter your local province or state (e.g. "Baden-Württemberg").
- ▶ City or Locality: Enter name of city (e.g. "Neckartenzlingen").
- ▶ Organization Name: Enter name of organization (e.g. "Belden Deutschland GmbH").
- ▶ Organization Unit: Enter name of organization unit (e.g. "Belden Deutschland GmbH").
- ▶ Common Name: The common name here belongs to domain name. It must reflect fully or in parts the domain name of where LioN-X device is accessible.
- ▶ Email address: The e-mail address of the certificate's creator.

**Create Certificate Database**

Database	example (SharkSSL Enabled)
Key Size	secp256r1
Signature Size	sha256
Days	10950
Country Name	DE
State or Province	Baden-Wuttemberg
City or Locality	Neckartenzlingen
Organization Name	Belden Deutschland GmbH
Organizational Unit	Belden Deutschland GmbH
Common Name	Lumberg
Email Address	info@belden.com

**Create Key & Certificate**

Certificate Management App (V 5) Real Time Logic © 2018

### 3. Upload the certificate onto the LioN-X device:

In the HTTPS certificate manager (Belden Web interface), click on the button *Upload* and choose the “.pem” and “.key” files generated in the previous step for the upload.

Click on *Upload*.

#### Server certificate upload

Chose server certificate file:  
 certificate.pem

Choose private key file:  
 privkey.pem

Passphrase:

**Upload idle.**  
Uploading file...  
File uploaded succesfully  
Running action...  
**Post upload action finished**



## 15.2 LioN-Xlight 0980 LSL... variants

### 15.2.1 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

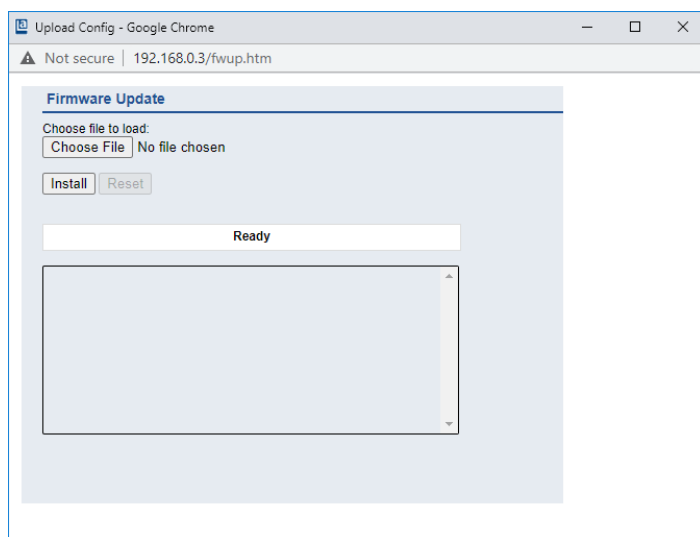
Use this parameter to change the current IP address of the module.

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.



## 16 IODD

IODD functions are **only** applicable for the following device variants:

- ▶ 0980 XSL 3912-121-007D-00F
- ▶ 0980 XSL 3912-121-007D-01F
- ▶ 0980 XSL 3912-121-027D-01F
- ▶ 0980 XSL 3913-121-007D-01F
- ▶ 0980 XSL 3913-121-027D-01F

The **IO Device Description** (IODD) is a set of files formally describing an IO-Link Device. The IODD is created by the vendor and is mandatory for each IO-Link Device.

Belden LioN-X IO-Link Masters with the "IODD on Module" functionality are ready to use IODDs in order to make the IO-Link Device configuration much more easier and the process data human readable in a better way. IODDs can be uploaded via the Web Interface and remanently stored on the IO-Link Master afterwards.

If a corresponding IO-Link Device is connected, the stored IODD is used to provide a user friendly configuration page, where all parameters of the device can be viewed and edited. Additionally, according to the IODD, the process data will also be formatted and displayed to the user.

### 16.1 IO-Link Device parameters and ISDU requests

Every IO-Link Device provides parameters that can be read and written via the special IO-Link service ISDU (Indexed **S**ervice **D**ata **U**nit).


Every parameter is addressed by an index. Sub-indices are possible but optional. Some parameters (most of them read-only) are mandatory for IO-Link devices an can be found always on the same indices (See *Table B.8* in the *IO-Link Interface and System Specification*: [https://io-link.com/share/Downloads/Package-2020/IOL-Interface-Spec\\_10002\\_V113\\_Jun19.pdf](https://io-link.com/share/Downloads/Package-2020/IOL-Interface-Spec_10002_V113_Jun19.pdf)).

A vendor can use additional parameters and therefore more indices for their devices in order to provide additional configuration options. These vendor specific parameters can be described in an IODD. The "IODD on Module" feature of the LioN-X IO-Link Masters can read and parse this information out of an IODD and use it to provide the user viewing and editing options for vendor specific parameters without any additional knowledge about the vendor specific device features.

## 16.2 Web GUI functionality

All of the "IODD on Module" features are accessible via the LioN-X Web interface.

### 16.2.1 Port Details page

 **lumbergautomation**  
A BELDEN BRAND

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**

Forcemode Forcemode off

Port X2

Type IO-Link

Dia

**Port Diagnosis**

No diagnosis

**Pin 4 / Channel A**

Function IO-Link

4 Bytes In, 4 Bytes Out

State Operate

**Pin 2 / Channel B**

Function Inactive

State Inactive

**IO-Link Events**

No events

**IO-Link**

Vendor ID 362

Device ID 36741114

Vendor Name BELDEN Deutschland GmbH

Vendor Text www.beldensolutions.com

Product Name 0960 IOL 381-001

Product ID: 934992002

Product Text LioN-P IO-Link (IO-Hub, 16DI)

Serial No. x42n

HW Revision V1

FW Revision V3.0.0.0

Speed COM3

Cycle time 1000

IODD Upload

Configure device

Application Name (Tag) AppTag7

Set

83 c8 88 88

HEX

Name	Value
Port X1A	false
Port X1B	false
Port X2A	false
Port X2B	false
Port X3A	false
Port X3B	false

The Port Details Page shows all information about the selected port. In the left column, all port and channel specific information is displayed. If the port is configured as IO-Link and there is an IO-Link Device connected, all IO-Link information for the connected device is displayed in the right column.

### IODD buttons

The row called *IODD* provides access to the "IODD on Module" features. The button *UPLOAD* will let the user upload an IODD file into the module, regardless of the original device the IODD has been designed for.

The maximum number of IODDs is limited due to storage space. If there is no more space left for new IODDs, there will be an error message. In this case, navigate to the IODD Management page to delete IODDs which are no longer used.

If there is a matching IODD for the currently connected device already stored in the system, the button *CONFIGURE* is shown in the interface. By clicking this button, the Parameter Page will open to configure the device.

### Process data

For every connected IO-Link Device, raw process data for input and output direction (set of bytes) is on display.

If a matching IODD providing information about process data is already stored in the system, this data will also be displayed in a user-friendly format according to the IODD.

16.2.2 Parameters page

IODD - Device configuration

Diagnosis

Parameter	Value	Unit	Min	Max	Description
Device Status	Device is OK				Indicator for the current device condition and diagnosis state.

Identification

Parameter	Value	Unit	Min	Max	Description
Vendor Name	BELEN Deutschland GmbH				The vendor name that is assigned to a Vendor ID.
Vendor Text	www.beldensolutions.com				Additional information about the vendor.
Product Name	0960 IOL 381-001				Complete product name.
Product ID	934992002				Vendor-specific product or type identification (e.g., item number or model number).
Product Text	LioN-P IO-Link I/O-Hub, 16DI				Additional product information for the device.
Serial Number	x42n				Unique, vendor-specific identifier of the individual device.
Hardware Revision	V1				Unique, vendor-specific identifier of the hardware revision of the individual device.
Firmware Revision	V3.0.0.0				Unique, vendor-specific identifier of the firmware revision of the individual device.
Application-specific Tag	appTag7		0	32	Possibility to mark a device with user- or application-specific information.
Function Tag	functionTag5		0	32	
Location Tag	locationTag5		0	32	

Parameter

Parameter	Value	Unit	Min	Max	Description
User Serial Number	x42n		0	16	
Module Identification ID	1		0	127	

General Device Settings

Parameter	Value	Unit	Min	Max	Description
I/O data mapping	LioN-P				
DIS-PRM-RST	enable parameter reset				

General Diagnostic Settings

Parameter	Value	Unit	Min	Max	Description
Disable peripheral diagnosis	enable diagnosis				

Input Filter

Parameter	Value	Unit	Min	Max	Description
Port X1A	off				
Port X1B	0.5ms				
Port X2A	1ms				
Port X2B	2ms				
Port X2C	2ms				

The parameters page "IODD – Device configuration" shows all parameters which are provided by the IODD of the device. That means the parameter set is variable and depends on the connected IO-Link Device.

The stored IODD reads the parameter meta data, such as names, units, min/ max values, descriptions etc. The values will be obtained directly from the connected device. For that reason it may take several seconds until the page is updated.

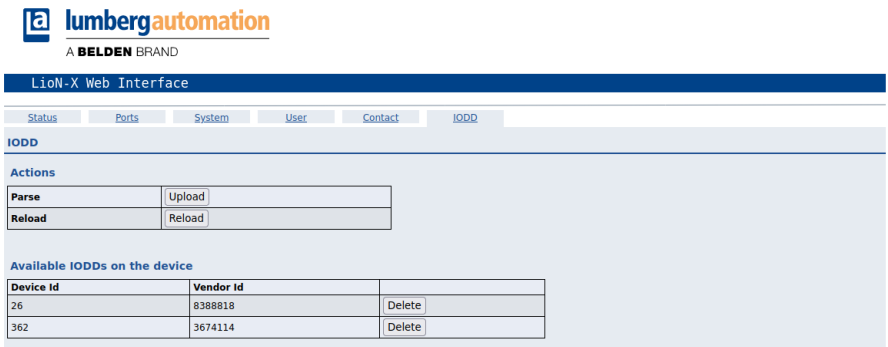
If not already saved into the browser, you will be asked for your credentials to continue. A valid user access with Web Interface group membership is needed in order to edit the device parameters. After the registration, enabled values can be changed. Disabled values cannot be changed and may be

marked as "read-only" in the IODD. All values are directly written back to the device after any change.

## Limitations

- ▶ Editing parameter values will directly change them inside the connected device. No parameter server action is triggered by that.
- ▶ There is a maximum size of the IODD in order to be uploaded into the system. This depends on several values, such as file size, parameter count, nesting levels etc.

### 16.2.3 IODD Management page



**lumbergautomation**  
A BELDEN BRAND

Lion-X Web Interface

Status Ports System User Contact IODD

**IODD**

Actions

Parse	Upload
Reload	Reload

Available IODDs on the device

Device Id	Vendor Id	
26	8388818	Delete
362	3674114	Delete

The IODD Management Page can be accessed via the System page displaying all IODDs that are currently stored in the system. All IODDs matching connected devices are marked. On the IODD Management page, you can manually delete any IODD in the system.

## Standard Definitions File

IODDs are usually referencing to a Standard Definitions File. The latest Standard Definitions File is already pre-installed on the system when the device is shipped. It can also be updated manually by clicking the button "Upload Standard Definitions File".

## 17 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](https://catalog.belden.com).



## 17.1 General

Protection class (Only applies if the connectors are screwed together or if protective caps are used.) <sup>2</sup>	IP65 IP67 IP69K	
Ambient temperature (during operation and storage)	0980 XSL 3x12-121...	-40 °C .. +70 °C (-40 °F .. +158 °F)
	0980 XSL 3x13-121...	
	0980 LSL 3x11-121...	-20 °C .. +60 °C (-4 °F .. +140 °F)
	0980 LSL 3x10-121...	
Weight	LioN-X 60 mm	approx. 500 gr. (17.6 oz)
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 56: General information*

<sup>2</sup> Not under UL investigation.

17.2 EtherNet/IP protocol

Protocol	EtherNet/IP, CIP V3.27
Update cycle	1 ms
EDS file	EDS-V3.27.1-BeldenDeutschland-XXX-yyyymmdd.eds
Transmission rate	10/100 Mbit/s, half/full duplex
Transmission procedure Autonegotiation	10BASE-T/100BASE-TX supported
RPI min.	1 ms
Vendor ID	21
Product type	12 (Communications Adapter)
Product code	41000 (0980 XSL 3912-121-007D-00F, 935700-001) 41001 (0980 LSL 3111-121-0006-002, 935701-002) 41002 (0980 LSL 3110-121-0006-002, 935702-002) 41003 (0980 XSL 3913-121-007D-01F, 935703-001) 41005 ((0980 XSL 3912-121-007D-01F, 935700-002) 41006 (0980 XSL 3912-121-027D-01F ,935710-001) 41007 (0980 XSL 3913-121-027D-01F ,935711-001)
Supported Ethernet protocols	Ping ARP- HTTP / HTTPS TCP/IP DHCP/BOOTP
Switch functionality	Integrated
EtherNet/IP interface Connections Autocrossing	2 M12 sockets, 4-pin, D-coded (see pin assignments) 2 M12 Hybrid male/female, 8-pin supported
Electrically isolated Ethernet ports -> FE	2000 V DC

Table 57: EtherNet/IP protocol

## 17.3 Power supply of the module electronics/sensors

Port X03, X04	M12-L-coded Power, connector/socket, 5-pole Pin 1 / Pin 3		
Nominal voltage $U_S$	24 V DC (SELV/PELV)		
Current $U_S$	Max. 16 A		
Voltage range	21 .. 30 V DC		
Power consumption of module electronics	Typically 160 mA (+/-20 % at $U_S$ nominal voltage)		
Power supply interruption	Max. 10 ms		
Voltage ripple $U_S$	Max. 5 %		
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 XSL 3913-121...		
	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
Voltage level of the sensor power supply	Min. ( $U_S - 1.5 \text{ V}$ )		
Short circuit/overload protection of sensor supply	Yes, per port		
Reverse polarity protection	Yes		
Operational indicator ( $U_S$ )	LED green:	18 V (+/- 1 V) < $U_S$	
	LED red:	$U_S < 18 \text{ V (+/- 1 V)}$	

*Table 58: Information on the power supply of the module electronics/sensors*

## 17.4 Power supply of the actuators

### 17.4.1 IO-Link Class A devices ( $U_L$ )

Nominal voltage $U_L$	24 V DC (SELV/PELV)
Voltage range	18 .. 30 V DC
Current $U_L$	Max. 16 A
Voltage ripple $U_L$	Max. 5 %
Reverse polarity protection	Yes
Operational indicator ( $U_L$ )	LED green: 18 V (+/- 1 V) < $U_L$ LED red: $U_L$ < 18 V (+/- 1 V) or $U_L$ > 30 V (+/- 1 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 59: Information on the power supply of the actuators

### 17.4.2 IO-Link Class A/B devices ( $U_{AUX}$ )

Nominal voltage $U_{AUX}$	24 V DC (SELV/PELV)
Voltage range	18 .. 30 V DC
Current $U_{AUX}$	Max. 16 A
Voltage ripple $U_{AUX}$	Max. 5 %
Reverse polarity protection	Yes
Electric isolation $U_S \leftrightarrow U_{AUX}$	500 V
Operational indicator ( $U_{AUX}$ )	LED green: 18 V (+/- 1 V) < $U_{AUX}$ LED red: $U_{AUX}$ < 18 V (+/- 1 V) or $U_{AUX}$ > 30 V (+/- 1 V) * if "Report $U_{AUX}$ supply voltage fault" is enabled.
Port X03, X04	M12_L-coded Power, connector/socket, 5-pole Pin 2 / Pin 4

Table 60: Information on the power supply of the actuators

## 17.5 I/O ports Channel A (Pin 4)

0980 XSL 3912-121...	Port X1 .. X8	Class A	IOL, DI, DO	M12 socket, 5-pin, Pin 4
0980 LSL 3x11-121...	Port X1 .. X8	Class A	IOL, DI, DO	
0980 LSL 3x10-121...	Port X1 .. X4	Class A	IOL, DI, DO	
	Port X5 .. X8	—	—, DI, —	
0980 XSL 3913-121...	Port X1 .. X4	Class A	IOL, DI, DO	
	Port X5 .. X8	Class B	IOL, DI, DO	

Table 61: IO-Link Master ports: Functional overview for Ch. A (Pin 4)

### 17.5.1 Configured as digital input, Ch. A (Pin 4)

Input connection	0980 XSL 3912-121...		Type 1 as per IEC 61131-2
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...		
	0980 XSL 3913-121...		
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...	X1 .. X8	8
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...		
	0980 XSL 3913-121...		
Status indicator	yellow LED		
Diagnostic indicator	red LED per channel		

Table 62: I/O ports Ch. A (Pin 4) configured as digital inputs

## 17.5.2 Configured as digital output, Ch. A (Pin 4)

**i Attention:** For variants 0980 XSL 3912-121-007D-00F, 0980 XSL 3912-121-007D-01F and 0980 XSL 3912-121-027D-01F, the digital outputs of Channel A are **supplied by the  $U_L$  power** when parameterized to "High-Side Switch" mode.

**i Attention:** For variants 0980 XSL 3913-121-007D-01F and 0980 XSL 3913-121-027D-01F, the digital outputs are supplied as follows:

► "X1 .. X8 / Channel A" are supplied by the  $U_S$  power

**i Attention:** For variants 0980 LSL 3010-121-0006-001 and 0980 LSL 3011-121-0006-001, the digital outputs of Channel A are **supplied by the  $U_S$  power**.

Output type	normally open, p-switching (parameterized to "High-Side Switch" mode)	
Nominal output voltage per channel		
Signal status "1"	min. ( $U_S - 1\text{ V}$ ) <b>or</b> min. ( $U_L - 1\text{ V}$ ) depending on the device variant	
Signal status "0"		
Max. output current per device	0980 XSL 3912-121...	9 A (power supplied via $U_L$ )
	0980 XSL 3913-121...	9 A (power supplied via $U_S$ )
	0980 LSL 3x11-121...	4 A (power supplied via $U_S$ )
	0980 LSL 3x10-121...	2 A (power supplied via $U_S$ )
Max. output current per channel <sup>3</sup>	0980 XSL 3912-121... (X1 .. X8)	2 A (power supplied via $U_S$ )
	0980 XSL 3913-121... (X1 .. X8)	2 A (power supplied via $U_S$ )
	0980 LSL 3x11-121... (X1 .. X8)	0.5 A (power supplied via $U_S$ )
	0980 LSL 3x10-121... (X1 .. X4)	0.25 A for <b>UL applications</b>

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

Short-circuit/overload protected	yes/yes	
Behavior in case of short circuit or overload	deactivation with automatic power-on (parameterized)	
Number of digital outputs	0980 XSL 3912-121... (X1 .. X8)	8
	0980 XSL 3913-121... (X1 .. X8)	
	0980 LSL 3x11-121... (X1 .. X8)	
	0980 LSL 3x10-121... (X1 .. X4)	4
Status indicator	yellow LED per output	
Diagnostic indicator	red LED per channel	

*Table 63: I/O ports Ch. A (Pin 4) configured as digital outputs*

### 17.5.3 Configured as IO-Link port in COM mode, Ch. A

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	0980 XSL 3912-121... (X1 .. X8)	8
	0980 XSL 3913-121... (X1 .. X8)	8
	0980 LSL 3x11-121... (X1 .. X8)	8
	0980 LSL 3x10-121... (X1 .. X4)	4
Min. IO-Link cycle time	400 µs	

*Table 64: Configured as IO-Link port in COM mode*

## 17.6 I/O ports Channel B (Pin 2)

0980 XSL 3912-121...	Port X1 .. X8	Class A	DI, DO	M12 socket, 5-pin, Pin 2
0980 LSL 3x11-121...	Port X1 .. X8	Class A	DI	
0980 LSL 3x10-121...	Port X1 .. X4	Class A	DI	
	Port X5 .. X8	–	DI	
0980 XSL 3913-121...	Port X1 .. X4	Class A	DI, DO	
	Port X5 .. X8	Class B	DO, U <sub>AUX</sub>	

Table 65: IO-Link Master ports: Functional overview for Ch. B (Pin 2)

### 17.6.1 Configured as a digital input, Ch. B (Pin 2)

Input connection	0980 XSL 3912-121...	Type 1 as per IEC 61131-2	
	0980 XSL 3913-121...		
	0980 LSL 3x11-121...		
	0980 LSL 3x10-121...		
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...	X1 .. X8	8
	0980 XSL 3913-121...	X1 .. X4	4
	0980 LSL 3x11-121...	X1 .. X8	8
	0980 LSL 3x10-121...	X1 .. X8	8
Status indicator	white LED		
Diagnostic indicator	red LED per channel		

Table 66: I/O ports Ch. B (Pin 2) configured as digital inputs



### 17.6.2 Configured as a digital output, Ch. B (Pin 2)



**Attention:** For variants 0980 XSL 3912-121-007D-00F, 0980 XSL 3912-121-007D-01F and 0980 XSL 3912-121-027D-01F, the digital outputs of Channel B are **supplied by the  $U_L$  power**.



**Attention:** For variants 0980 XSL 3913-121-007D-01F and 0980 XSL 3913-121-027D-01F, the digital outputs are supplied as follows:

- ▶ "X1 .. X4 / Channel B" are supplied by the  $U_S$  power
- ▶ "X5 .. X8 / Channel B" are supplied by the  $U_{AUX}$  power



**Attention:** For variants 0980 LSL 3010-121-0006-001 and 0980 LSL 3011-121-0006-001, the digital outputs of Channel B are **supplied by the  $U_S$  power**.

Output type	normally open, p-switching	
Nominal output voltage per channel Signal status "1" Signal status "0"	min. ( $U_S - 1\text{ V}$ ) <b>or</b> min. ( $U_L - 1\text{ V}$ ) <b>or</b> min. ( $U_{AUX} - 1\text{ V}$ ) depending on the device variant max. 2 V	
Max. output current per device	0980 XSL 3912-121...	9 A (power supplied via $U_L$ )
	0980 XSL 3913-121...	8 A (power supplied via $U_{AUX}$ )
	0980 LSL 3x11-121...	4 A (power supplied via $U_S$ )
	0980 LSL 3x10-121...	2 A (power supplied via $U_S$ )
Max. output current per channel 4,5	0980 XSL 3912-121...	2 A (power supplied via $U_S$ )
	0980 XSL 3913-121...	X1 .. X4: 2 A (power supplied via $U_S$ )
		X5 .. X8: 2 A (power supplied via $U_{AUX}$ )
	0980 LSL 3x11-121...	0 A (no outputs on Ch. B)
	0980 LSL 3x10-121...	0 A (no outputs on Ch. B)
Short-circuit/overload protected	yes/yes	
Behavior in case of short circuit or overload	deactivation with automatic power-on (parameterized)	
Number of digital outputs	0980 XSL 3912-121...	8
	0980 XSL 3913-121...	8
	0980 LSL 3x11-121...	–
	0980 LSL 3x10-121...	–
Status indicator	white LED per output	

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

<sup>5</sup> For Class A/B devices: Max. 2.0 A per channel; for every port pair X1/X2, X3/X4, X5/X6, X7/X8 max. 6.5 A (for **UL applications** max. 5.0 A); for port group X5/X6/X7/X8 max. 5.0 A from  $U_{AUX}$ ; for the whole port group X1 .. X8 max. 9.0 A in total (with derating).

Diagnostic indicator	red LED per channel
----------------------	---------------------

Table 67: I/O ports Ch. B (Pin 2) configured as digital outputs

## 17.7 LEDs

LED	Color	Description
U <sub>L</sub> /U <sub>AUX</sub>	Green	Auxiliary sensor/actuator voltage OK 18 V (+/- 1 V) < U <sub>L</sub> /U <sub>AUX</sub> < 30 V (+/- 1 V)
	Red*	Auxiliary sensor/actuator voltage LOW U <sub>L</sub> /U <sub>AUX</sub> < 18 V (+/- 1 V) or U <sub>L</sub> /U <sub>AUX</sub> > 30 V (+/- 1 V) * if "Report U <sub>L</sub> /U <sub>AUX</sub> supply voltage fault" is enabled.
	OFF	None of the above conditions.
U <sub>S</sub>	Green	System/sensor voltage OK 18 V (+/- 1 V) < U <sub>S</sub> < 30 V (+/- 1 V)
	Red	System/sensor voltage LOW U <sub>S</sub> < 18 V (+/- 1 V) or U <sub>S</sub> > 30 V (+/- 1 V)
	Red flashing	Device performs a factory reset (position of rotary encoding switches: 9-7-9)
X1 .. X8 A	OFF	None of the above conditions.
	Green	IO-Link COM Mode: IO-Link communication exists.
	Green flashing	IO-Link COM Mode: No IO-Link communication.
X1 .. X8 B	Yellow	Standard-I/O Mode: Status of digital input or output on C/Q (pin 4) line "on".
	OFF	None of the above conditions
	White	Status of digital input or digital output on pin 2 line "on".
P1 Lnk/Act P2 Lnk/Act	Red	Short circuit on pin 4 and pin 2 line. / All modes: Overload or short circuit on L+ (pin 1) line / communication error
	OFF	None of the above conditions.
	Green	Ethernet connection to another subscriber exists. Link detected.
P1 Lnk/Act P2 Lnk/Act	Yellow flashing	Data exchange with another subscriber.
	OFF	No connection to another subscriber. No link, no data exchange.

LED	Color	Description
BF	Red	Bus fault. No configuration, no or slow physical connection.
	Red flashing at 2 Hz	Link exists but no communication link to the EtherNet/IP controller.
	OFF	EtherNet/IP controller has established an active connection to the device.
DIA	Red	EtherNet/IP 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.
MS	Green	Device is ready for operation.
	Green flashing	Device is ready but not configured yet.
	Red	Serious error that cannot be resolved.
	Red flashing	Minor error that can be resolved Example: An incorrect or contradictory configuration is classified as a minor error.
	Flashing alternately:	The device is performing a self-test.
	Red Green	
	OFF	The device is switched off.

LED	Color		Description
NS	Green		Connected: The device has at least one connection.
	Green flashing		No connection: The device has no connection. IP address exists.
	Red		Duplicate IP address: The device has detected that the assigned IP address is already being used by another device.
	Red flashing		Connection has exceeded time limit or connection interrupted.
	Flashing alternately:		The device is performing a self-test.
	Red	Green	
	OFF		The device is switched off or has not been assigned an IP address.

Table 68: Information on the LED colors

## 17.8 Data transfer times

The following tables give an overview of the internal data transfer times of the Lion-X IO-Link Master with a connected IO-Link Device as digital I/O extension (Belden article 0960 IOL 380-021 16DIO Hub with a minimum cycle time of 1 ms).

There are three measured data direction values for each use case:

- ▶ **PLC to DO:** Transfer of a changed PLC output data to IO-Link Device digital output.
- ▶ **DI to PLC:** Transfer of a changed digital input signal on IO-Link Device to PLC.
- ▶ **Round-trip time (RTT):** Transfer of a changed PLC output data to IO-Link Device digital output. The digital output is connected to an IO-Link Device digital input. Transfer of the changed digital input signal on IO-Link Device to PLC.  $RTT = [PLC \text{ to } DO] + [DI \text{ to } PLC]$ .

The measured values are taken from the ethernet data transmission line. The values are therefore without PLC processing times and PLC cycle time.

The configurable digital input filter value on 0960 IOL 380-021 was set to "off" (0 ms).

For calculation of user specific data transfer and round-trip times of possible input filters, PLC processing and cycles times must be taken into calculation.

The measured values are valid for a maximum of 48 bytes of IO-Link data for the IO-Link Master in each direction (Input/Output).

**Use case 1:**

IO-Link Master configuration with enabled Web interface and *disabled* IIoT protocols

Data direction	Data transfer time in ms		
	Minimum	Average	Maximum
PLC to DO	3.7	6.0	7.7
DI to PLC	1.1	3.0	4.3
RTT	6.1	8.9	11.1

**Use case 2:**

IO-Link Master configuration with enabled Web interface and *enabled* IIoT protocols

Data direction	Data transfer time in ms		
	Minimum	Average	Maximum
PLC to DO	7.7	10.0	13.4
DI to PLC	3.3	4.4	5.6
RTT	12.1	14.3	17.0



## 18 Accessories

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

<http://www.beldensolutions.com>