



Operating instructions
SolutionBlock IO-Link master
Multiprotocol

GB

AL1591

Firmware: 1.2.x.y

Contents

1	Preliminary note	8
1.1	Legal and copyright information	8
1.2	Symbols used	8
1.3	Warnings	8
1.4	Safety symbol on the device	8
1.5	Change history	9
2	Safety instructions	10
3	Intended use	11
4	System description	12
4.1	IO-Link	12
4.1.1	IO-Link supply	12
4.1.2	Parameter storage and device validation	12
4.2	Parameter setting	12
4.3	Visual indication	13
4.4	IIoT interface	13
4.5	Fieldbus interface	13
4.5.1	PROFINET	13
4.5.2	EtherNet/IP	14
4.6	IODD management	14
4.6.1	IODD Extender	15
4.6.1.1	Example: IO-Link device KQ6005;	15
4.7	OPC UA server	16
4.8	Applications	16
4.8.1	Avahi	16
4.8.2	ecologConnect	16
4.8.3	Network Time Protocol (NTP)	17
4.8.4	Node-RED	17
5	Installation	18
5.1	Overview	18
5.2	Installing the device	18
6	Electrical connection	19
6.1	Overview	19
6.2	General wiring information	19
6.2.1	Connection technology	19
6.3	Fieldbus ports	20
6.4	IIoT ports	20
6.5	IO-Link ports (class B)	20
6.6	IO-Link ports (Class A)	21
6.7	Functional earthing	21
6.8	Voltage supply	21
6.8.1	Derating behaviour	22
6.9	Voltage output	23
7	Operating and display elements	24
7.1	Overview	24
7.2	LEDs	24
7.2.1	Status	24
7.2.1.1	Gateway and system	24
7.2.1.2	Fieldbus communication	24
7.2.2	Ethernet	25
7.2.3	Voltage supply	25
7.2.4	IIoT ports	26
7.2.5	IO-Link ports (class B)	26
7.2.6	IO-Link ports (class A)	26
8	Set-up	28
8.1	Connect with IIoT network	28
8.2	Connection to the fieldbus network	28

9	Settings	29
9.1	Parameter setting software	29
9.1.1	Supported parameter setting software	29
9.1.2	Connect to the device	29
9.1.2.1	Online parameter setting	30
9.1.2.2	Offline parameter setting	31
9.1.3	IoT interface	31
9.1.3.1	Configuring the security mode	31
9.1.3.2	Configuring IP settings	31
9.1.4	Fieldbus interface	32
9.1.4.1	Read status of the fieldbus connection	32
9.1.4.2	Setting the fieldbus type	33
9.1.4.3	Reading the IP configuration of the PROFINET interface	33
9.1.4.4	Reading the IP configuration of the EtherNet/IP interface	34
9.1.4.5	Setting the EtherNet/IP configuration mode	34
9.1.5	IO-Link ports	35
9.1.5.1	Setting the operating mode pin 2 (UA)	35
9.1.5.2	Setting the operating mode Pin 4 (US)	36
9.1.5.3	Setting the device validation and data storage	36
9.1.5.4	Setting the cycle time	37
9.1.5.5	Setting fail-safe values	38
9.1.6	Process data	39
9.1.6.1	Reading digital input data	39
9.1.6.2	Writing digital output data	39
9.1.6.3	Reading status information of the device	40
9.1.6.4	Reading timer counter values	40
9.1.6.5	Reading port events	40
9.1.7	Device information	41
9.1.7.1	Reading identification information	41
9.1.8	Device control	42
9.1.8.1	Resetting the device	42
9.1.8.2	Restarting the device	42
9.1.8.3	Identifying the device	42
9.1.9	Updating the firmware	43
9.2	IoT Core Visualizer	44
9.2.1	Calling up the IoT Core Visualiser	44
9.2.2	Device settings	44
9.2.2.1	Fieldbus settings	45
9.2.2.2	System time settings	45
9.2.2.3	Authentication settings	47
9.2.2.4	Communication settings	49
9.2.3	Notifications	54
9.2.3.1	Create new notifications	55
9.2.3.2	Delete notifications	55
9.2.4	IIoT Monitor	55
9.2.4.1	Creating a new monitoring task	56
9.2.4.2	Deleting a monitoring task	57
9.2.5	IoT Core elements	57
9.2.5.1	Searching for elements of the ifm IoT Core	58
9.2.5.2	Executing services of the ifm IoT Core	59
9.2.6	Parameter	59
9.2.6.1	Configuring the IIoT interface	60
9.2.6.2	Configuring the timer	61
9.2.6.3	Configuring the fieldbus interface	62
9.2.6.4	Reading identification information	64
9.2.6.5	Reading software information	65
9.2.7	Process data	65
9.2.7.1	Reading temperature values	65
9.2.7.2	Reading timer counter readings	66
9.2.7.3	Reading the status and diagnostic data of the device	66
9.2.7.4	Reading and writing process data	67
9.2.7.5	Reading IO-Link events	68

9.2.7.6	Reading port events	68
9.2.7.7	Reading and writing information about IO-Link devices	69
9.2.8	Update	70
9.2.8.1	Firmware update	70
9.2.9	Diagnostics	70
9.2.9.1	Downloading diagnostic data	71
9.2.10	IODD	71
9.2.10.1	Loading IODDs manually onto the device	72
9.2.10.2	Downloading IODD from the IODDfinder portal to the device	72
9.2.10.3	Deleting the IODD from the device	72
9.3	ifm IoT Core REST API	74
9.3.1	ifm IoT Core: General information	74
9.3.2	Accessing elements of the ifm IoT Core	74
9.3.2.1	GET request	74
9.3.2.2	POST request	75
9.3.2.3	Diagnostic codes	76
9.3.3	General functions	77
9.3.3.1	Example: Output a subtree	77
9.3.3.2	Example: Changing a parameter value	78
9.3.3.3	Example: Reading multiple elements sequentially	79
9.3.3.4	Example: Writing multiple elements sequentially	80
9.3.3.5	Example: Browsing device description	80
9.3.4	Device settings	81
9.3.4.1	Setting the fieldbus type	81
9.3.4.2	Configuring the security mode	81
9.3.4.3	Setting the system time	83
9.3.4.4	Configuring the timers	84
9.3.4.5	Configuring the status LED	86
9.3.5	IIoT interface	87
9.3.5.1	Configuring the IIoT interface	87
9.3.6	Connections	90
9.3.6.1	Using HTTPS connection	90
9.3.6.2	Using HTTP connection	90
9.3.6.3	Using MQTT client	91
9.3.6.4	Using MQTT command channel	93
9.3.6.5	Using the OPC UA server	96
9.3.7	Fieldbus interface	99
9.3.7.1	PROFINET: Reading the interface configuration	99
9.3.7.2	EtherNet/IP Reading the interface configuration	99
9.3.7.3	EtherNet/IP Setting the configuration mode	100
9.3.8	Software	101
9.3.8.1	Reading software information	101
9.3.8.2	Firmware update	101
9.3.9	Timer	103
9.3.9.1	Configuring the timers	103
9.3.10	Device status	105
9.3.10.1	Reading the operating status	105
9.3.10.2	Reading operating hours	105
9.3.10.3	Configuring the status LED	105
9.3.10.4	Reading temperatures	105
9.3.11	IO-Link ports	107
9.3.11.1	Setting operating mode pin 4 (C/Q)	107
9.3.11.2	Configuring the device validation and data storage	107
9.3.11.3	Setting the operating mode pin 2 (I/Q)	108
9.3.11.4	Switching the supply voltage of the ports on / off	108
9.3.11.5	Reading / writing process data of the ports	111
9.3.11.6	Accessing the data memory	111
9.3.11.7	Reading the port type	112
9.3.11.8	Reading Port events	113
9.3.12	IO-Link devices	115
9.3.12.1	Accessing data objects of the IO-Link device	115
9.3.12.2	Reading / writing process data of the IO-Link devices	116

9.3.12.3	Accessing interpreted data from the IO-Link device	119
9.3.12.4	Reading / writing identification information of the IO-Link device	119
9.3.12.5	Reading IO-Link events	120
9.3.13	Monitor	121
9.3.13.1	Using IIoT monitoring	121
9.3.14	IODD manager	123
9.3.14.1	Using IODD manager	123
9.3.14.2	Using IODD extender	124
9.3.15	Device control	127
9.3.15.1	Controlling the device	127
9.3.16	Device management	128
9.3.16.1	Manage external devices	128
9.3.17	Applications	131
9.3.17.1	Using NTP	131
9.3.17.2	Using ecologConnect	132
9.3.17.3	Using Node-RED	133
9.3.17.4	Using Avahi	133
9.3.18	Diagnostic and status information	135
9.3.18.1	Reading status and diagnostic information	135
9.3.18.2	Downloading diagnostic data	135
9.3.19	Identification information	136
9.3.19.1	Read identification information of the device	136
9.3.19.2	Read identification information of the IO-Link master	136
9.3.20	Using notifications	137
9.3.20.1	Event-triggered notifications	137
9.3.20.2	Time-triggered notifications	137
9.3.20.3	Example: Create notification job (JSON format)	138
9.3.20.4	Example: Change notification order	139
9.3.20.5	Example: Create notification order (CSV format)	140
9.3.20.6	Example: Create notification job (WebSocket)	141
9.3.20.7	Example: Output all active notification orders	143
9.3.20.8	Example: Delete notification order	144
9.4	PROFINET	145
9.4.1	Note: Startup package	145
9.4.2	Install the GSD file	145
9.4.3	Integrating the device into a project	145
9.4.3.1	Use S2 redundancy	145
9.4.3.2	Use Configuration-in-Run	146
9.4.4	Configuring the IO-Link master	146
9.4.5	Setting the function of the ports	146
9.4.6	Parameter setting of the ports	147
9.4.7	Use I&M data records	148
9.5	EtherNet/IP	149
9.5.1	Note: Startup package	149
9.5.2	Integrating the device into a project	149
9.5.2.1	Top-down mode	149
9.5.2.2	Independent mode	151
9.5.3	Accessing the process data	154
9.5.4	Accessing IO-Link events	154
9.5.5	Accessing diagnostic and status information	155
9.5.6	Using energy monitoring	155
9.5.7	Accessing IO-Link devices	156
9.5.8	Using acyclic services	156
9.5.8.1	Using the acyclic command channel	156
9.5.8.2	Acyclic port commands	157
9.5.8.3	EtherNet/IP mechanisms for acyclic commands	157
10	Maintenance, repair and disposal	158
10.1	Cleaning	158
10.2	Firmware update	158
10.3	Resetting the device via recovery mode	158
11	Appendix	160

11.1	ifm IoT Core	160
11.1.1	Profiles	160
11.1.2	Types	160
11.1.3	Services	161
11.1.3.1	Service: abortinstall	161
11.1.3.2	Service: add	161
11.1.3.3	Service: adddevice	161
11.1.3.4	Service: copy_iodd	162
11.1.3.5	Service: delete_iodd	162
11.1.3.6	Service: download_iodd	162
11.1.3.7	Service: enable	163
11.1.3.8	Service: extend	163
11.1.3.9	Service: factoryreset	163
11.1.3.10	Service: getdata	163
11.1.3.11	Service: getdatamulti	163
11.1.3.12	Service: getdevicelist	164
11.1.3.13	Service: getidentity	164
11.1.3.14	Service: getitems	165
11.1.3.15	Service: getssubscriberlist	166
11.1.3.16	Service: gettree	166
11.1.3.17	Service: gotobootloader	167
11.1.3.18	Service: install	167
11.1.3.19	Service: iolreadacyclic	167
11.1.3.20	Service: iolwriteacyclic	167
11.1.3.21	Service: mirror	168
11.1.3.22	Service: power_cycle	168
11.1.3.23	Service: power_off	168
11.1.3.24	Service: power_on	168
11.1.3.25	Service: querytree	168
11.1.3.26	Service: reboot	169
11.1.3.27	Service: remove	169
11.1.3.28	Service: removedevice	169
11.1.3.29	Service: reset	169
11.1.3.30	Service: setblock	170
11.1.3.31	Service: setdata	170
11.1.3.32	Service: setdatamulti	170
11.1.3.33	Service: signal	170
11.1.3.34	Service: start	170
11.1.3.35	Service: start_stream_get	171
11.1.3.36	Service: start_stream_set	171
11.1.3.37	Service: stop	171
11.1.3.38	Service: stream_get	171
11.1.3.39	Service: stream_set	172
11.1.3.40	Service: subscribe	172
11.1.3.41	Service: supported_devicetypes	173
11.1.3.42	Service: swinfo	174
11.1.3.43	Service: unmirror	174
11.1.3.44	Service: unsubscribe	174
11.1.3.45	Service: upload_certificate	175
11.1.3.46	Service: validation_useconnecteddevice	175
11.2	EtherNet/IP	176
11.2.1	Parameters	176
11.2.1.1	Configuration assembly (instance 199)	176
11.2.2	Cyclic data	178
11.2.2.1	Input assembly (instance 100)	178
11.2.2.2	Input Assembly (Instance 101)	182
11.2.2.3	Output Assembly (Instance 150)	188
11.2.2.4	Output Assembly (Instance 151)	190
11.2.3	Acyclical data	192
11.2.3.1	Acyclic command channel	192
11.2.3.2	Acyclic commands	197
11.2.4	Object directory	205

11.2.4.1	CIP object classes	205
11.2.4.2	CIP class and instance services	205
11.2.4.3	Identity Object (Class Code: 0x01)	206
11.2.4.4	Message Router Object (Class Code: 0x02)	208
11.2.4.5	Assembly Object (Class Code: 0x04)	209
11.2.4.6	Connection Manager Object (Class Code: 0x06)	210
11.2.4.7	Device Level Ring Object (Class Code: 0x47)	211
11.2.4.8	Quality Of Service Object (Class Code: 0x48)	212
11.2.4.9	IO-Link Request Object (Class Code: 0x80)	213
11.2.4.10	TCP/IP Object (Class Code: 0xF5)	219
11.2.4.11	Ethernet Link Object (Class Code: 0xF6)	221
11.2.4.12	LLDP Management Object (Class Code: 0x109)	223
11.3	PROFINET	224
11.3.1	Modules	224
11.3.1.1	IOLM Proxy Module	224
11.3.2	Submodules	225
11.3.2.1	IOLM Proxy Submodule: IO-Link master proxy	225
11.3.2.2	IOLD Proxy Submodule: IO-Link master proxy	225
11.3.3	Parameters	226
11.3.3.1	IOLM Proxy Module	226
11.3.3.2	IOLD Proxy Submodule	227
11.3.4	Cyclic data	234
11.3.4.1	IOLM proxy: IO-Link master proxy	234
11.3.4.2	IOLD Proxy: IO-Link n I / m O + PQI	236
11.3.4.3	IOLD Proxy: IO-Link n I + PQI	237
11.3.4.4	IOLD Proxy: IO-Link m O + PQI	238
11.3.4.5	IOLD Proxy: Digital Input	239
11.3.4.6	IOLD Proxy: Digital Output	240
11.3.5	Acyclical data	241
11.3.5.1	I&M data	241

1 Preliminary note

You will find instructions, technical data, approvals and further information using the QR code on the unit / packaging or at documentation.ifm.com.

1.1 Legal and copyright information

© All rights reserved by ifm electronic gmbh. No part of these instructions may be reproduced and used without the consent of ifm electronic gmbh.

All product names, pictures, companies or other brands used are the property of the respective rights owners.

1.2 Symbols used

- ✓ Requirement
- Instructions
- ▷ Reaction, result
- [...] Designation of keys, buttons or indications
- Cross-reference
-  Important note
Non-compliance may result in malfunction or interference.
-  Information
Supplementary note

1.3 Warnings

Warnings indicate the possibility of personal injury and damage to property. This enables safe product handling. Warnings are graded as follows:

	WARNING Warning of serious personal injury ▷ If the warning is not observed, fatal and serious injuries are possible.
	CAUTION Warning of minor to moderate personal injury ▷ If the warning is not observed, minor to moderate injuries are possible.
	ATTENTION Warning of damage to property ▷ If the warning is not observed, damage to property is possible.

1.4 Safety symbol on the device

-  Safety symbol on the device:
 - Adhere to the operating instructions for the safe operation of the device.

1.5 Change history

Issue	Subject	Date
00	New creation of the document	04 / 2025

2 Safety instructions

- The unit described is a subcomponent for integration into a system.
 - The system architect is responsible for the safety of the system.
 - The system architect undertakes to perform a risk assessment and to create documentation in accordance with legal and normative requirements to be provided to the operator and user of the system. This documentation must contain all necessary information and safety instructions for the operator, the user and, if applicable, for any service personnel authorised by the architect of the system.
- Read this document before setting up the product and keep it during the entire service life.
- The product must be suitable for the corresponding applications and environmental conditions without any restrictions.
- Only use the product for its intended purpose (→ Intended use).
- If the operating instructions or the technical data are not adhered to, personal injury and/or damage to property may occur.
- The manufacturer assumes no liability or warranty for any consequences caused by tampering with the product or incorrect use by the operator.
- Installation, electrical connection, set-up, operation and maintenance of the product must be carried out by qualified personnel authorised by the machine operator.
- Protect units and cables against damage.
- Replace damaged units, otherwise the technical data and safety will be impaired.
- Observe applicable documents.
- Store the product in its original packaging.

3 Intended use

The device may only be used for the following purposes:

- IO-Link master for configuration, management and operation of IO-Link devices
- Gateway between the Operational Technology (OT) level and Information Technology (IT) level
- Gateway between the operational technology level and a higher-level fieldbus level
- for pre-processing and aggregation of data from the OT level

Permitted use:

- in industrial food and beverage production environments as a permanently installed device

4 System description

4.1 IO-Link

The device offers the functionality of an IO-Link master (IO-Link standard: 1.1/4")

4.1.1 IO-Link supply

The device provides power supplies for IO-Link devices (sensors, actuators).

- The ports X1...X4 are IO-Link ports (Class B).
- The ports X5...X8 are IO-Link ports (Class A).

The device ensures fire protection for connected IO-Link devices by providing an energy-limited circuit at the ports (according to IEC61010-1 and class 2 to UL1310).

4.1.2 Parameter storage and device validation

The device supports the IO-Link functions device validation and automatic backup/restoration of parameter data of the connected IO-Link devices (Data Storage + Validation).

The following options are available:

Option	Validation of the connected IO-Link devices	Backing up the parameter data	Restoring the parameter data
No check and clear	No	No	No
Type compatible V1.0 device	Yes, check for compatibility with IO-Link standard V1.0.	No	No
Type compatible V1.1 device	Yes, check for compatibility with IO-Link standard V1.1.	No	No
Type compatible V1.1 with Backup + Restore	Yes, test compatibility with IO-Link standard V1.1 and identity of design (vendor ID, device ID)	Yes, automatic storage of the parameter values; Changes of the current parameter values will be stored.	Yes, automatic restoration of the parameter values when connecting an identical IO-Link device in the delivery state
Type compatible V1.1 with Restore	Yes, test compatibility with IO-Link standard V1.1 and identity of design (vendor ID, device ID)	No, there is no automatic backup; Changes to the parameter values are not stored	Yes, automatic restoration of the parameter values when connecting an identical IO-Link device in the delivery state

4.2 Parameter setting

The device can be configured with the following options:

- Parameter setting software
 - ifm moneo|IoT Core Cloud (art. no QCM100)
 - ifm moneo|IoT Core On-Premises (art. no QM9112)
 - ifm moneo|configure free
- ifm IoT Core:
 - ifm IoT Core Visualizer
 - REST API
- PROFINET
 - Projection software

- EtherNet/IP
 - Projection software

4.3 Visual indication

The device displays the following indications:

- Status and error display of the device
- Status and error display of the active fieldbus gateway
- Status and activity indication of the Ethernet connections
- Status of the voltage supplies US and UA
- Status and activity display of the IIoT connection
- Status and error indication of the IO-Link ports (class A)
- Status and error indication of the IO-Link ports (class B)

4.4 IIoT interface

The IIoT interface is used to integrate the device into the Operational Technology (OT) level and Informational Technology (IT) level.

The IIoT interface is realised with a 2-port switch (ports XI1 and XI2). Ports XI1 and XI2 operate in bridge mode (network bridge).

The IIoT interface supports the following methods for IP configuration (IPv4):

- static (manual configuration)
- DHCP (automatic configuration)
- APIPA - Link-local (automatic addressing)

The IIoT interface supports the following protocols:

- Transmission Control Protocol / Internet Protocol (TCP/IP)
- Hypertext Transfer Protocol (HTTP)
- Hypertext Transfer Protocol Secure (HTTPS)
- Message Queueing Telemetry Transport (MQTT)
- Message Queueing Telemetry Transport Secure (MQTTS)
- Websockets (WS)

4.5 Fieldbus interface

The device is integrated into a higher-level fieldbus level via the fieldbus interface.

The fieldbus interface is realised with a 2-port switch (ports XF1 and XF2).

The fieldbus interface supports several fieldbus types (multiprotocol).

Supported fieldbus types:

- [PROFINET \(→ 13\)](#)
- [EtherNet/IP \(→ 14\)](#)

4.5.1 PROFINET

Supported PROFINET functions:

- Device profile: PROFINET IO device
- Fast Startup (FSU)
- Prioritized Startup
- IRT Class C (no synchronous data)
- System redundancy S2
- Media redundancy (MRP Client)
- Dynamic reconfiguration
- Port Deactivation
- Device description: GSD file

4.5.2 EtherNet/IP

The device supports the following EtherNet/IP functions:

- Device profile: EtherNet/IP device
- Min. cycle time: 1 ms /RPI)
- Supported connection classes: 1, 3
- Supported connection types:
 - Exclusive owner
 - Input Only
 - Listen Only
- Predefined fieldbus objects
 - Identity Object (0x01)
 - Message Router Object (0x02)
 - Assembly Object (0x04)
 - Connection Manager (0x06)
 - DLR Object (0x47)
 - QoS Object (0x48)
 - IO-Link Request Object (0x80)
 - TCP/IP Interface Object (0xF5)
 - Ethernet Link Object (0xF6)
- Supported protocols: DHCP, BOOTP, ACD, DLR
- Device description: EDS file

4.6 IODD management

The device has an integrated repository for device description files for IO-Link devices (Input Output Device Description - IODD).

IODD management offers the following functionality:

- Selective storage of IODDs on the device
- Management of the IODDs stored on the device (adding, deleting)
- Optional: Automatic download of IODDs from the IODDfinder portal ioddfinder.io-link.com
- Provision of IODDs for other components of the IoT Core, e.g. IODD extender

4.6.1 IODD Extender

The device has a function for automatically detecting IO-Link devices and displaying the interpreted data of the detecting IO-Link devices.

Available functions:

- Initial check of the entire IoT Core of the device for the presence of IO-Link devices (element profile: iolinkdevice_full)
- When changes are made to the tree structure of the IoT Core or when connected IO-Link devices are detected

For all detected IO-Link devices:

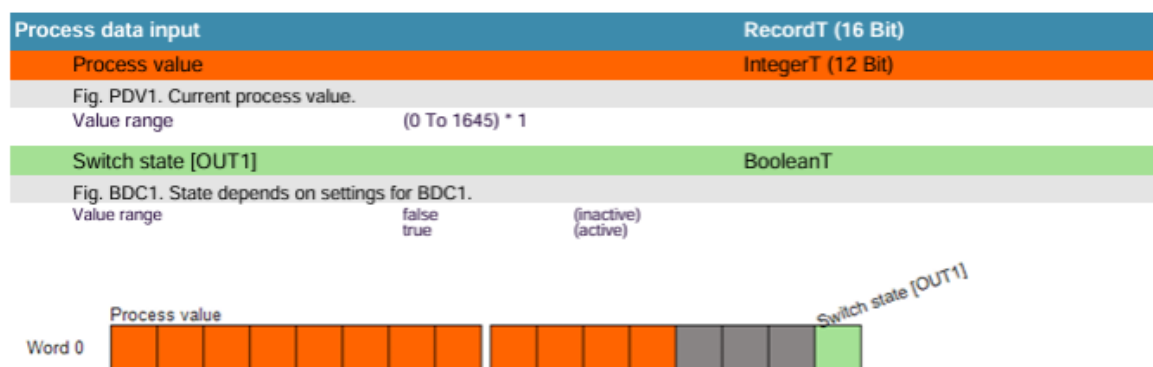
- Requesting the appropriate IODD from the IODD administration
- Automatic extension of the IoT Core tree in substructure /iolinkdevice
- Interpreting the data of the IO-Link device using the IODD

The IODD extender function is enabled in the factory settings.

4.6.1.1 Example: IO-Link device KQ6005;

- IO-Link input process data (record 16 bits) consisting of process value in °C (integer 12 bits) and switching status [OUT1] (Boolean 1 bit)

6 Observation



IO-Link input process data: 0x0160 = 0b0000 0001 0110 0000

- Process value: 0b0000 0001 0110 = 22°C
- Switching status [OUT1]: 0b0 = False

Display in the ifm IoT Core:

- Without IODD Extender:

```
...
iolinkdevice/
...
  "pdin": 0160
...
...
```

- With IODD Extender

```
...
iolinkdevice/
...
  pdin_interpreted
    process_data_input
      "process_value":22
      "switch_state_[out1]":false
    ...
  ...
...
```

4.7 OPC UA server

The device offers a server in accordance with the OPC Unified Architecture (OPC UA) standard.

The OPC UA server provides access to the following device data:

- Process data
- Identification information
- Diagnostic and status information
- Services

Supported security policies:

- Basic256Sha256
- Aes128_Sha256_RsaOaep
- Aes256_Sha256_RsaPss

Supported security modes:

- Sign&Encrypt

Access data set at the factory:

- User name: root
- Password: secret

The OPC UA server is disabled at the factory.

4.8 Applications

The following applications are installed on the device:

- [Avahi](#) (→ [16](#))
- [ecologConnect](#) (→ [16](#))
- [Network Time Protocol \(NTP\)](#) (→ [17](#))
- [Node-RED](#) (→ [17](#))

4.8.1 Avahi

The device supports the Avahi protocol.

The application is enabled at the factory.

4.8.2 ecologConnect

The device supports the ecologOne software. With ecologOne, the user can back up and restore configurations and user data of the device.



Information about ecologOne: www.ifm.com

The ecologConnect service must be enabled so that ecologOne can access the device.

The service is enabled at the factory.

4.8.3 Network Time Protocol (NTP)

The device supports the system time setting via Network Time Protocol (NTP).

The service is disabled by default.

4.8.4 Node-RED

The device supports the Node-RED platform for the creation of event-driven applications.



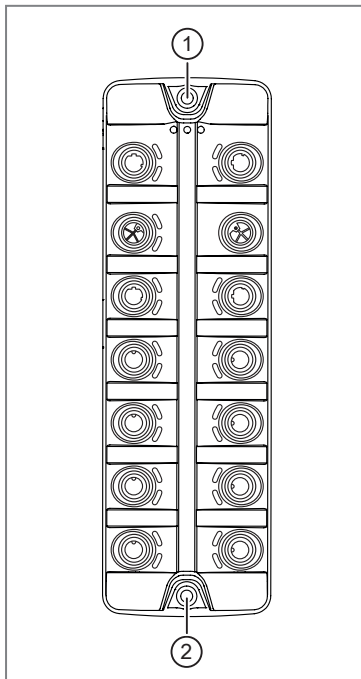
Further information about Node-RED: nodered.org

The Node-Red environment is pre-installed on the device.

Node-RED is disabled at the factory.

5 Installation

5.1 Overview



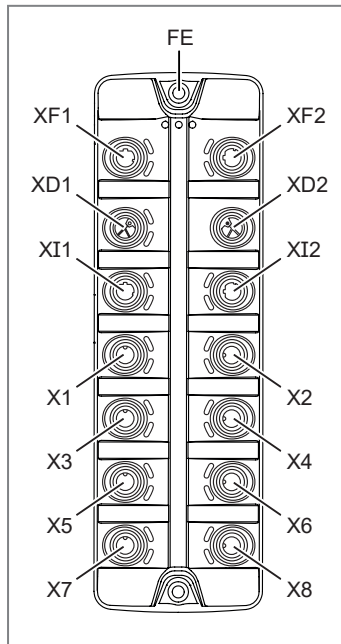
- 1: Upper mounting lug
- 2: Lower mounting lug

5.2 Installing the device

- ▶  Disconnect the power of the machine before installation.
- ▶ Use a flat mounting surface for installation.
- ▶ Observe the maximum tightening torque.
- ▶  Before installation and when replacing the appliance, ensure that the degree of soiling is observed (→ Technical data).
- ▶ Fasten the module onto the mounting surface using two M5 screws and washers (tightening torque: 1.8 Nm).

6 Electrical connection

6.1 Overview



- FE: Functional earth (FE)
- XF1: Fieldbus port 1 (PROFINET / EtherNet/IP)
- XF2: Fieldbus port 2 (PROFINET / EtherNet/IP)
- XD1: Voltage supply
- XD2: Voltage output
- XI1: IloT port 1
- XI2: IloT port 2
- X1: IO-Link port (class B)
- X2: IO-Link port (class B)
- X3: IO-Link port (class B)
- X4: IO-Link port (class B)
- X5: IO-Link port (class A)
- X6: IO-Link port (class A)
- X7: IO-Link port (class A)
- X8: IO-Link port (class A)

6.2 General wiring information



The unit must be connected by a qualified electrician.

- Observe the national and international regulations for the installation of electrical equipment.

The device is suitable for operation on PELV voltages.

- Observe the notes on IO-Link wiring.

This device contains components that may be damaged or destroyed by electrostatic discharge (ESD).

- Please observe the required precautions against electrostatic discharge!

The circuits are insulated from each other and from device surfaces that could be touched with basic insulation according to IEC 61010-1 (secondary circuit with max. 30 V DC, supplied from the mains circuit up to 300 V of overvoltage category II). The external wiring has to be carried out in a way that ensures the required separation from other circuits.

The communication interfaces are isolated from each other, from the electrical circuits and from the touch surfaces of the device with basic insulation in accordance with EN61010-1 (secondary circuit with maximum 30 V DC, derived from mains circuit up to 300 V of overvoltage category II). The external wiring has to be carried out in a way that ensures the required separation from other circuits.

6.2.1 Connection technology

The threaded connections in the device correspond to the M12 standard. To ensure compliance with the specified protection rating, only cables that comply with this standard may be used. In the case of self-assembled cables, the system manufacturer is responsible for the protection rating.

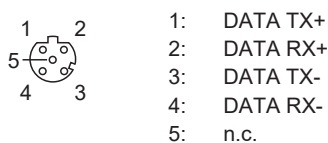
- Use connectors with gold-plated contacts.
- During installation, place the connectors vertically so that the coupling nut will not damage the thread.

- ▶ Before connecting the connector, check that there is an undamaged O-ring in the socket.
- ▶ Observe the coding of the connectors during installation.
- ▶ Provide all outgoing cables with suitable strain relief after a maximum of 400 mm. Observe the minimum bending radius of the cables (→ information from the cable manufacturer).
- ▶ Cover unused connections with protective covers. Tightening torque: 1.3 ± 0.1 Nm
- ▶ Check the screw connection 2 weeks after installation and retighten to the correct torque if necessary.

6.3 Fieldbus ports

The device is connected to the fieldbus network via ports XF1, XF2, e.g. fieldbus controller or additional fieldbus device.

Wiring:

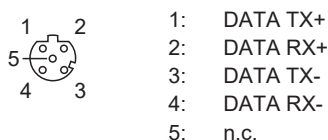


- ▶ Connect the device to the IT fieldbus via the ports XF1, XF2.
- ▶ For connection, use M12 connectors (minimum protection rating: IP65 / IP66 / IP67 / IP69K).
- ▶ Tighten the cable plug using 1.3 ± 0.1 Nm.

6.4 IloT ports

The device is connected IT network via the ports XI1, XI2.

Wiring:



- ▶ Connect the device to the IT network via the ports XI1, XI2.
- ▶ For connection, use M12 connectors (minimum protection rating: IP65 / IP66 / IP67 / IP69K).
- ▶ Tighten the cable plug using 1.3 ± 0.1 Nm.

6.5 IO-Link ports (class B)

IO-Link devices (sensors, actuators) are connected to the ports X1...X4.

The IO-Link ports meet the requirements of IO-Link specification 1.1.4.

- The connected IO-Link devices may only be supplied via the corresponding ports.
- The ports each have an additional digital output (utilisation category DC-13 according to standard IEC 60947-5-1, 20 W.).
- ▶ Observe the maximum current rating of the supply voltages US and UA: → Technical data

The ports are equipped with short-circuit detection (US).

The ports have voltage peak and overvoltage protection.

Wiring:



- 1: L+ (US; sensor supply)
- 2: L+ (UA; Actuator supply / DO2)
- 3: L- (US; sensor supply)
- 4: IO-Link / DI1 / DO1
- 5: L- (UA; actuator supply)

- ▶ Connect the IO-Link devices to ports X1...X4. Maximum cable length per port: 20 m.
- ▶ For connection, use M12 connectors (minimum protection rating: IP65 / IP66 / IP67 / IP69K).
- ▶ Tighten the cable plug using 1.3 ± 0.1 Nm.

6.6 IO-Link ports (Class A)

IO-Link devices (sensors, actuators) are connected to the ports X5...X8.

The IO-Link ports meet the requirements of IO-Link specification 1.1.4.

- The connected IO-Link devices may only be supplied via the corresponding ports.
- The ports have an additional digital input each (type 2, according to IEC 61131-2).
- Port X8 has an additional digital input to put the device into recovery mode.

The ports are equipped with short-circuit detection (US).

The ports have voltage peak and overvoltage protection.

Wiring:



Ports X5...X7

- 1: L+ (US; sensor supply)
- 2: DI2
- 3: L- (US; sensor supply)
- 4: IO-Link / DI1 / DO1
- 5: n.c.

Port X8

- 1: L+ (US; sensor supply)
- 2: DI2
- 3: L- (US; sensor supply)
- 4: IO-Link / DI1 / DO1
- 5: Reserved Recovery Pin

- ▶ Connect the IO-Link devices to ports X5...X8. Maximum cable length per port: 20 m.
- ▶ For connection, use M12 connectors (minimum protection rating: IP65 / IP66 / IP67 / IP69K).
- ▶ Tighten the cable plug using 1.3 ± 0.1 Nm.

6.7 Functional earthing

The FE potential is connected to the following points of the device:

- Upper mounting lug of the housing
- Port XD1: socket thread
- Ports XD1 and XD2: Pin 5 (FE)
- Ports XF1 and XF2: socket thread
- Ports XI1 and XI2: socket thread



To ensure electrical interference protection, proper function and safe operation of the appliance:

- ▶ Connect the FE connection to the system earth or GND of the supply via the shortest route.

6.8 Voltage supply

The device is connected to the supply voltages US and UA via the XD1 port.

The US supply voltage supplies the device and the sensors connected to the ports X1...X8.

The UA supply voltage supplies the actuators connected to the X1...X4 ports.

The XD1 port is equipped with overvoltage protection (US, UA).

The XD1 port is equipped with reverse polarity protection (US, UA).



WARNING

No internal fusing of the supply voltages US and UA for daisy chaining

- ▷ Fire hazard in the event of overload or short circuit
- ▶ Limit or fuse the supply voltages US and UA externally.
- ▶ Observe technical data, derating values and applicable regulations!
- ▷ Protection is guaranteed.

Wiring:



- 1: L+ (US)
- 2: L- (UA)
- 3: L- (US)
- 4: L+ (UA)
- 5: FE

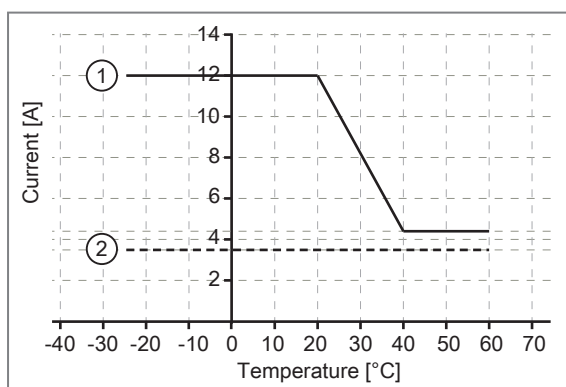
- ▶ Disconnect power.
- ▶ Connect the device via port XD1 to the supply voltage US with 24 V DC (18...28 V PELV).
- ▶ Connect the device via port XD1 to the supply voltage UA with 24 V DC (18...28 V PELV).
- ▶ Use an EVCA-type L-coded M12 plug connection from ifm for the connection (min. protection rating: IP65 / IP66 / IP67 / IP69K).
- ▶ Tighten the cable sockets according to the torque specifications indicated by the cable vendor. Maximum permissible tightening torque: 0.9 Nm

6.8.1 Derating behaviour

The current I_{US} available at ports X1...X8, the current I_{UA} available at ports X1...X4, and the currents $I_{US-daisy-chain}$ and $I_{UA-daisy-chain}$ available at port XD2 depend on the ambient temperature of the device.

- $I_{US}(XD1) = I_{US}(X1...X8) + I_{US-daisy-chain}(XD2)$
- $I_{UA}(XD1) = I_{UA}(X1...X4) + I_{UA-daisy-chain}(XD2)$

Height: 2000 m



- 1: $I_{US-daisy-chain}(XD2), I_{UA-daisy-chain}(XD2)$
- 2: $I_{US}(X1...X8), I_{UA}(X1...X4)$

Example:

Temperature	$I_{US}(X1...X8)$	$I_{US-daisy-chain}(XD2)$	$I_{UA}(X1...X4)$	$I_{UA-daisy-chain}(XD2)$
20 °C	3.6 A	12 A	3.6 A	12 A
30 °C	3.6 A	8 A	3.6 A	8 A
40 °C	3.6 A	4.3 A	3.6 A	4.3 A

6.9 Voltage output

An additional device can be supplied via the XD2 port (daisy chain). The supply voltages US and UA are looped through from port XD1 to port XD2.

Wiring:

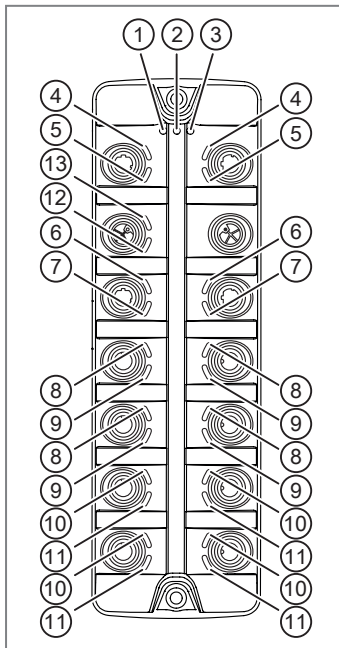


- 1: L+ (US)
- 2: L- (UA)
- 3: L- (US)
- 4: L+ (UA)
- 5: FE

- ▶ Disconnect the unit from the voltage supply.
- ▶ Optional: Connect an additional device to port XD2.
- ▶ Use an EVCA-type L-coded M12 plug connection from ifm for the connection (min. protection rating: IP65 / IP66 / IP67 / IP69K).
- ▶ Tighten the cable plug using 1.3 ± 0.1 Nm.

7 Operating and display elements

7.1 Overview



- 1: Status: RDY
- 2: Status: BF (PROFINET) / NET (EtherNet/IP)
- 3: Status: SF (PROFINET) / MOD (EtherNet/IP)
- 4: Ethernet: LINK
- 5: Ethernet: ACT
- 6: IIoT: LINK
- 7: IIoT: ACT
- 8: IO-Link (Class B):
- 9: IO-Link (Class B): UA/DO
- 10: IO-Link (Class A):
- 11: IO-Link (Class A): DI
- 12: Voltage supply: UA
- 13: Voltage supply: US

7.2 LEDs

7.2.1 Status

7.2.1.1 Gateway and system

Device in standard mode:

LED	Description	Colour	State	Description
RDY	Gateway and system status	--	Off	Gateway: Not active or reboots
		Green	Flashes (5 Hz)	Gateway: Error
			Flashes (200 ms on, 800 ms off)	Gateway: Firmware update running
			On	Gateway: OK
		red	Flashes (2 Hz)	Error / power supply error

7.2.1.2 Fieldbus communication



The flashing behaviour of the LEDs depends on the active fieldbus type.

PROFINET

LED	Description	Colour	Status	Description
BF	BF (Bus Failure): PROFINET status	--	Off	PROFINET: error-free
		red	Flashes (1 Hz)	PROFINET: no connection to the controller

LED	Description	Colour	Status	Description
BF	BF (Bus Failure): PROF- INET status	red	On	PROFINET: bus error
SF	SF (System Failure): sys- tem status	--	Off	System: error-free
		red	On	System: error in the gateway or at least one IO-Link port reports a warning/alarm (temperature, overcur- rent, overvoltage, undervoltage, short circuit)

EtherNet/IP

LED	Description	Colour	State	Description
NET	NET (Network Status): net- work status	--	Off	Not powered, no IP address: no voltage applied or voltage applied but IP address not yet configured
		Green / red / yel- low	flashes quickly, al- ternately	Self-test: Device in self-test after switching on
		red / green / off	flashes, alternating	Flashing sequence: Visual identification of the de- vice
		Green	Flashes (1 Hz)	No connection: no CIP connection established and no Exclusive Owner connection timeout
			On	Connected: At least one CIP connection estab- lished and no timeout for Exclusive Owner connec- tion
		red	flashes (1 Hz)	Connection Timeout: Timeout for Exclusive Owner connection
			On	Duplicate IP: IP address used twice
MOD	MOD (Module Status): Eth- erNet/IP module status	--	Off	No power: no voltage is applied or the applied volt- age is too low
		Green / red / yel- low	flashes quickly, al- ternately	Self-test: Device in self-test after switching on
		red / green / off	flashes, alternating	Flashing sequence: Visual identification of the de- vice
		Green	Flashes (1 Hz)	Standby: device not yet configured (no IP address)
			On	Device operational: Device functions correctly
		red	flashes (1 Hz)	Major recoverable fault: serious, recoverable error (e.g. incorrect configuration)
			On	Major unrecoverable fault: serious, unrecoverable error (e.g. module failed)

7.2.2 Ethernet

LED	Description	Colour	State	Description
LINK	Status of the connection	--	Off	No Ethernet connection
		Green	On	Ethernet connection established
ACT	Status of the data transmis- sion	--	Off	No data transmission
		Yellow	flashes	Data transmission

7.2.3 Voltage supply

LED	Description	Colour	State	Description
US	Status of the US supply voltage	--	Off	No supply voltage applied

LED	Description	Colour	State	Description
US	Status of the US supply voltage	Green	On	Applied supply voltage in the normal range 18...28 V
		Yellow	On	Applied supply voltage in the warning range 18...20 V or 28...33 V
		red	On	Applied supply voltage too low (<18 V) or too high (>33 V) or device switched off due to overload on IO-Link ports
UA	status of the UA supply voltage	--	Off	No supply voltage applied
		Green	On	Applied supply voltage in the normal range 18...28 V
		Yellow	On	Applied supply voltage in the warning range 18...20 V or 28 ...33 V
		red	On	Applied supply voltage too low (<18 V) or too high (>33 V) or device switched off due to overload on IO-Link ports

7.2.4 IIoT ports

LED	Description	Colour	State	Description
LINK	Status of the connection	--	Off	No Ethernet connection
		Green	On	Ethernet connection established
ACT	Status of the data transmission	--	Off	No data transmission
		Yellow	flashes	Data transmission

7.2.5 IO-Link ports (class B)

LED	Description	Colour	State	Description
	Status of the IO-Link port (pin 4)	--	Off	Port mode DI / DO: Pin 4 (C/Q) = LOW level
		Yellow	On	Port mode DI / DO: Pin 4 (C/Q) = HIGH level
		Green	Flashes (1 Hz)	Port mode IO-Link: no IO-Link communication.
			On	Port mode IO-Link: IO-Link communication
		red	Flashes (1.2 Hz)	Port mode IO-Link: Short circuit / overload on US or IO-Link communication error
UA/DO	Status of the IO-Link port (pin 2)	--	Off	Mode DO / Type A Supply: Pin 2 (I/Q) = OFF
		Yellow	On	Mode DO: Pin 2 (I/Q) = ON
		Green	On	Mode Type B Supply: Pin 2 (I/Q) = ON (not switchable)
		red	On	Error: Overcurrent or undervoltage + overvoltage

7.2.6 IO-Link ports (class A)

LED	Description	Colour	State	Description
	Status of the IO-Link port (pin 4)	--	Off	Port mode DI / DO: Pin 4 (C/Q) = LOW level
		Yellow	On	Port mode DI / DO: Pin 4 (C/Q) = HIGH level
		Green	Flashes (1 Hz)	Port mode IO-Link: no IO-Link communication.
			On	Port mode IO-Link: IO-Link communication
		red	Flashes (1.2 Hz)	Port mode IO-Link: Short circuit / overload on US or IO-Link communication error
DI	Status of the digital input (pin 2)	--	Off	Digital input Pin 2 = LOW level

LED	Description	Colour	State	Description
DI	Status of the digital input (pin 2)	Yellow	On	Digital input Pin 2 = HIGH level

8 Set-up

8.1 Connect with IIoT network

The device is connected to the following instances via the IIoT interface:

- Parameter setting software, e.g. ifm moneo|configure
- Additional ifm IIoT devices at OT level, e.g. additional IO-Link masters
- IIoT applications, e.g. MQTT server or OPC UA clients

Behaviour of the IIoT interface during initial set-up:

- IP configuration via DHCP is active in the factory settings. The device is trying to obtain a valid IP configuration from a DHCP server in the network.
 - If no valid IP configuration can be obtained from a DHCP server, the IIoT interface is assigned an IP address in the address range 169.254.0.0 to 169.254.255.255 using the link-local protocol.
- Connect the device to the IIoT network via port XI1, XI2.
- Call up the IoT Core Visualiser via the automatically assigned IP address.
- If necessary, set the IP configuration of the IIoT interface to manual addressing.

8.2 Connection to the fieldbus network

In the factory settings, the fieldbus interface has the following configuration:

Parameter	PROFINET	EtherNet/IP
IP address	0.0.0.0	192.168.1.250
Subnetwork	0.0.0.0	255.255.255.0
gateway	0.0.0.0	0.0.0.0

The IP configuration of the fieldbus interface can only be set via the respective fieldbus configuration software.

- ✓ The required fieldbus type is set.
- Connect the device to the fieldbus network via port XF1, XF2.
- Scan the network for devices in the fieldbus configuration software.
- Set the IP configuration of the fieldbus interface as required.

9 Settings

9.1 Parameter setting software

9.1.1 Supported parameter setting software

Supported parameter setting options:

- ifm moneo|IIoT Core On-Premises (art. no QM9112)
- ifm moneo|IIoT Core Cloud (art. no QCM100)
- ifm moneo|configure free



► For correct installation: observe the installation instructions for ifm moneo.

- Install the desired parameter setting software.
- Activate the licences required for operation.
- ▷ The installed software can be used to configure the device.

9.1.2 Connect to the device

The connection to the device is established using a specific network scan. The connection parameters to be set depend on the following device functions:

- Authentication (login function)
- HTTP communication



With a standard DCP scan, ifm moneo finds the device, but can only establish a connection if authentication is disabled and HTTP communication is enabled.

Requirements:

- Click on the [Configure devices] tile.
 - ▷ The [Configure devices] view appears.
- Click on the [Network scan] button.
 - ▷ A dialogue window appears.
- Select option [Specific IP address or host name].
- In the [Device type] field, select [Generic IoT-Core device].
 - ▷ An input screen with additional options appears.

► Enter the IP address of the device in the [Specific IP address or host name] field.

If HTTP communication is disabled:

► Enable the sliders [HTTPS] and [Trust insecure certificates] .

If the login function is enabled:

► Enter the user name admin in the [User name] field and enter the selected password in the [Password] field (default: password).

► Click on [SCAN].

▷ moneo scans the network for the device.

▷ moneo shows the device found and the connected IO-Link devices.

► In the line of the device: Click on ⚙️.

▷ A detailed view of the device appears.

9.1.2.1 Online parameter setting

The parameters of the IO-Link master and the connected sensors and actuators can be set before installation and set-up or during operation.



If you change parameters during operation, this will influence the function of the plant.

► Ensure that there will be no malfunctions in your plant.

During the parameter setting process, the IO-Link master and the connected IO-Link devices stay in operating mode. They continue to perform their functions with the existing parameters until the parameter setting has been completed.

9.1.2.2 Offline parameter setting

The device supports offline parameter setting. The user creates a configuration for the IO-Link master and the connected IO-Link devices without being connected to the device. The configuration created can be stored as a file (*.lrp) and loaded to the device and activated at a later point.

9.1.3 IoT interface

The [IoT] menu provides access to the parameters of the IoT interface.

9.1.3.1 Configuring the security mode

The device offers a security mode. The security mode protects access to the device via the IoT interface (ports XI1, XI2). The following conditions apply when safety mode is enabled:

- Access only by authentication (password protection)
- Access only via encrypted connection (TLS - Transport Layer Security)

Security mode is enabled in the factory settings and cannot be disabled (user name: admin, password: password). The user can set a user-specific password.

When establishing a connection to the device, the user name and password are requested. After entering the correct password, the user can access the device and the connected IO-Link devices without restriction. At the end of the current session of the parameter setting software, the user name and password are requested again, e.g. after a restart.

Available parameters:

Parameter	Description	Value range	Access
[Security mode HTTPS]	Status of the security mode	• Disabled: security mode disabled • Enabled: security mode enabled	ro ¹
[Security password]	Password for authentication NOTE! The set password is not displayed.		wo ²
[Authentication enabled]	Authentication status	• Disabled: Authentication disabled • Enabled: Authentication enabled	ro ¹

¹ Read only

² write only

To set a user-specific password:

Requirements:

- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- Select the [IoT] menu.
 - ▷ The menu page shows the current settings.
- In the [Security password] field, enter the user-specific password.
- Save the changes on the device.
 - ▷ The user-specific password is enabled.

9.1.3.2 Configuring IP settings

For communication with the IT network, the IP parameters of the IoT interface must be configured.

Available parameters:

Parameter	Description	Value range	Access
[Network Mode]	Method for the IP configuration of the interface	- Disabled: Schnittstelle deaktiviert - Static: DHCP client disabled IP parameters are set by the user - DHCP: DHCP client enabled; IP parameters are set by the network's DHCP server. (Default) - link-local: IP settings are set automatically (address range: 169.254.0.0 – 169.254.255.255, subnet mask: 255.255.0.0)	rw ¹
[IP address]	IP address of the IoT interface	e.g. 192.168.1.100 169.254.x.x (default)	rw ²
[Subnet mask]	Subnet mask of the Ethernet network	e.g. 255.255.255.0 255.255.0.0 (default)	rw ²
[Default gateway IP address]	IP address of the network gateway	e.g. 192.169.1.1 0.0.0.0 (default)	rw ²
[Static DNS servers]	Static DNS server	e.g. 192.169.1.100	rw ²
[DNS servers]	Active DNS servers	e.g. 192.169.1.100	ro ³
[MAC address]	MAC address of the IoT interface	e.g. 00:02:01:05:66:C1	ro ³
[Hostname]	Designation of the device in moneo (diagram: ifm article number-serial number)	ifm-AL1591-000242900024	ro ³

¹ Read and write

² Can only be changed if static addressing is active

³ Read only

To configure the IoT interface:

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The detailed view of the device is active.
- ▶ Select [IoT] menu.
 - ▷ The menu page shows the current settings.
- ▶ Set the parameters.
- ▶ Write the changed values to the device.
- ▷ IP settings of the IoT interface are configured.

9.1.4 Fieldbus interface

The [Fieldbus] menu provides access to the parameters of the fieldbus interface.

9.1.4.1 Read status of the fieldbus connection

Available parameters:

Parameter	Description	Value range	Access
[Fieldbus connection status]	Status of the fieldbus connection	- Disconnected: interrupted - Connected: connected	ro ¹

¹ Read only

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device is established (online).

- Select the [Fieldbus] menu.
- ▷ The menu page shows the status of the active fieldbus connection.

9.1.4.2 Setting the fieldbus type

The device can be operated with one of the following fieldbuses:

- PROFINET
- EtherNet/IP



When the fieldbus type is changed, the active fieldbus connection is interrupted.

Available parameters:

Parameter	Description	Value range	Access
[Fieldbus type]	Active fieldbus type	• Profinet: PROFINET is active • Ethernet/IP: EtherNet/IP is active	ro ¹

¹ Read only

To set one fieldbus type:

- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- Select the [Fieldbus] menu.
 - ▷ The menu page shows the active fieldbus type.

To activate PROFINET:

- Click on [Switch to Profinet] next to the [Fieldbus type Profinet] field.
- ▷ PROFINET field bus is enabled.
- ▷ The [Fieldbus type] field shows the change progress.
- ▷ After successful modification: A confirmation message appears.

To activate EtherNet/IP:

- Next to [Fieldbus type EtherNetIP] click on [Switch to EtherNetIP] .
- ▷ EtherNet/IP field bus is enabled.
- ▷ The [Fieldbus type] field shows the change progress.
- ▷ After successful modification: A confirmation message appears.

9.1.4.3 Reading the IP configuration of the PROFINET interface

The IP configuration of the PROFINET interface can only be set via the PROFINET configuration software.

Available parameters:

Parameter	Description	Value range	Access
[IP address]	IP address of the PROFINET interface	e.g. 192.168.10.100 • 0.0.0.0 (default)	ro ¹
[Subnet mask]	Subnet mask of the PROFINET network	e.g. 255.255.255.0 • 0.0.0.0 (default)	ro ¹
[Default gateway IP address]	IP address of the network gateway	e.g. 192.168.10.1 • 0.0.0.0 (default)	ro ¹
[Hostname]	Station Name of the device	E.g. iol.plant1	ro ¹

Parameter	Description	Value range	Access
[MAC address]	MAC address of the device's PROFINET interface	e.g. 00:02:01:0f:c8:8f	ro ¹
[Fieldbus firmware]	Version of the PROFINET firmware	E.g. 1.0.0	ro ¹

¹ Read only

To display the IP configuration of the PROFINET fieldbus interface:

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device is established (online).
- ✓ PROFINET is set as the active fieldbus type.
- Select the [Fieldbus] menu.
- ▷ The menu page shows the current configuration of the PROFINET interface.

9.1.4.4 Reading the IP configuration of the EtherNet/IP interface

The IP configuration of the Ether/Net-IP interface can only be set via the EtherNet/IP configuration software.

Available parameters:

Parameter	Description	Value range	Access
[IP address]	IP address of the EtherNet/IP interface	e.g. 192.168.10.100 • 192.168.1.250 (default)	ro ¹
[Subnet mask]	Subnet mask of the EtherNet/IP network	e.g. 255.255.255.0 • 255.255.255.0 (default)	ro ¹
[Default gateway IP address]	IP address of the network gateway	e.g. 192.168.10.1 • 0.0.0.0 (default)	ro ¹
[Hostname]	Host name of the device	E.g. iol.plant1	ro ¹
[MAC address]	MAC address of the device's EtherNet/IP interface	e.g. 00:02:01:0f:c8:8f	ro ¹
[Fieldbus firmware]	Version of the EtherNet/IP firmware	E.g. 1.1	ro ¹

¹ Read only

To display the IP configuration of the EtherNet/IP field bus interface:

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device is established (online).
- ✓ EtherNet/IP field bus is enabled.
- Select the [Fieldbus] menu.
- ▷ The menu page shows the current configuration of the EtherNet/IP interface.

9.1.4.5 Setting the EtherNet/IP configuration mode

The device supports the EtherNet/IP configuration modes Top-Down and Independent. For the Independent configuration mode, the user must set the length of the process data and the arrangement of the bytes within the process data.

Available parameters:

Parameter	Description	Value range	Access
[Independent mode]	EtherNet/IP configuration mode	- Off: Top-down mode active -- Configuration via Ethernet/IP PLC - On: Independent mode active - configuration via device settings	rw ¹
[Process data length]	Length of IO-Link process data per port	2 Bytes Input 2 Bytes Output: 2 bytes input data, 2 bytes output data 4 Bytes Input 4 Bytes Output: 4 bytes input data, 4 bytes output data 8 Bytes Input 8 Bytes Output: 8 bytes input data, 8 bytes output data 16 Bytes Input 16 Bytes Output: 16 bytes input data, 16 bytes output data 32 Bytes Input 32 Bytes Output: 32 bytes input data, 32 bytes output data	rw ¹
[Swap]	Arrangement of the bytes in process data	- Disabled: as Array of Bytes - Enabled: as Integer16 value; When the process data is updated, the bytes in the integer16 value are swapped	rw ¹

¹ Read and write

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device is established (online).
- ✓ Ethernet/IP field bus is enabled.
- ✓ The detailed view of the device is active.
- ▶ Select the [Fieldbus] menu.
 - ▷ The menu page shows the current settings.
- ▶ Set the parameters.
- ▶ Save the changes on the device.
- ▷ Ethernet/IP configuration mode is set.

9.1.5 IO-Link ports

The [Port x](x:1...8) menus provide access to the parameters of the IO-Link ports X1...X8.

9.1.5.1 Setting the operating mode pin 2 (UA)

Pin 2 of ports X1...X4 supports the following operating modes:

- Off: Port operates as IO-Link port (class A); Pin 2 is disconnected from the power supply
- On: Port operates as IO-Link port (class B); voltage UA is connected at pin 2;
- Digital output (DO): Port operates as a digital switching output; A binary output signal is available at pin 2

Available parameters:

Name	Description	Value range	Access
[Mode Pin2 UA]	Operating mode of pin 2 (I/Q) of the port	- Off (IO-Link Type A Supply): IO-Link Port Class A - On (IO-Link Type B Supply): IO-Link Port Class B - Digital Output: Digital switching output	rw ¹

¹ Read and write

To set the operating mode of pin 2 (UA) of a port:

Requirements:

- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
 - ▷ The menu page shows the current settings.
- ▶ Set the operating mode.
- ▶ Write the changed values to the device.
- ▷ Operating mode of pin 2 (UA) of the port has been set.

9.1.5.2 Setting the operating mode Pin 4 (US)

Pin 4 of ports X1...X8 supports the following operating modes:

- Disabled: No data transmission at pin 4 (C/Q) of the port
- Digital input: Binary input signal at pin 4 (C/Q) of the port
- Digital output: Binary output signal at pin 4 (C/Q) of the port
- IO-Link: IO-Link data transmission via pin 4 (C/Q) of the port

The set operating mode must match the operating mode of the device connected to the port (sensor, actuator, IO-Link device).

Available parameters:

Name	Description	Value range	Access
[Mode Pin4 US]	Operating mode of Pin 4 (C/Q) of the port	• Disabled: Port deactivated • DI: Digital input • DO: Digital output • IO-Link: IO-Link process data	rw ¹

¹ Read and write

Requirements:

- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- ▶ Select the [Port x] (x = 1...8) menu.
 - ▷ The menu page shows the current settings.
- ▶ Set the operating mode.
- ▶ Save the changes on the device.
- ▷ Operating mode for pin 4 (C/Q) of the port has been set.

9.1.5.3 Setting the device validation and data storage

Note the remarks: [Parameter storage and device validation](#) (→ [12](#))



Device validation and backup / restore of parameter values are only available if the “IO-Link” operating mode of the port has been enabled.

For options [Type compatible V1.1 device with Backup + Restore] and [Type compatible V1.1 device with Restore]: If the [Vendor ID] or [Device ID] parameters are changed in online mode, the data memory is deleted and a new backup of the parameter values of the connected IO-Link device is created in the IO-Link master.

Available parameters:

Name	Description	Value range	Access
[Validation / Data Storage]	Validation of the connected IO-Link devices and automatic backup/recovery of the parameter sets of the IO-Link device	- No check and clear: - no validation - No data storage and data recovery - Type compatible V1.0 device: - Validation (IO-Link V1.0) - No data storage and data recovery - Type compatible V1.1 device: - Validation (IO-Link V1.1) - No data storage and data recovery - Type compatible V1.1 device with Backup + Restore: - Validation (IO-Link V1.1) - Data storage and data recovery - Type compatible V1.1 device with Restore: - Validation (IO-Link V1.1) - Data recovery - No data storage	rw ^{1 2}
[Vendor ID]	Vendor ID of the IO-Link device to be validated against	E.g. 310: ifm electronic gmbh	rw ^{1 2}
[Device ID]	Device ID of the IO-Link device to be validated against	E.g. 1129 TCC501 (ifm temperature sensor)	rw ^{1 2}

¹ Read and write

² Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to the port

To configure the device validation and the backup / restore of parameter values:

Requirements:

- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- ✓ Operating mode of pin 4 (US) of the port is IO-Link.
- ▶ Select the [Port x] > [IO-Link] menu (x = 1...8).
 - ▷ The menu page shows the current settings.
- ▶ Set the parameters for device validation and backup / restore of parameter values.
- ▶ With active validation: Enter the manufacturer ID and device ID of the IO-Link device.
- ▶ Save the changes on the device.
- ▷ Device validation and backup / restore of parameter values have been configured.

9.1.5.4 Setting the cycle time

In the IO-Link operating mode, the cycle time of the port can also be set.

Available parameters:

Name	Description	Value range	Access
[Cycle time actual]	Current cycle time between the IO-Link master and the IO-Link device (value in µs)	1: 1 µs ... 132800: 132800 µs	ro ¹
[Cycle time preset]	Cycle time between the IO-Link master and the IO-Link device (value in µs) NOTE! Observe the technical data of the IO-Link device!	0: Fastest possible cycle time 1: 1 µs ... 132800: 132800 µs	rw ²

Name	Description	Value range	Access
[Bit rate]	Transmission rate between the IO-Link master and the IO-Link device	• COM1: 4.8k baud • COM2: (38.4 kbaud) • COM3: 230.4kbaud	ro ¹

¹ Read only

² Read and write; parameter can only be changed if no connection to fieldbus controller is established

Requirements:

- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- ✓ Operating mode of pin 4 (US) of the port is IO-Link.
- ▶ Select the [Port x] > [IO-Link] menu (x = 1...8).
- ▶ Setting the cycle time
- ▶ Write the changed values to the device.
- ▷ The cycle time of the IO-Link port has been set.

9.1.5.5 Setting fail-safe values

When the EtherNet/IP fieldbus is enabled for configuration mode [Independent mode], the user can set the fallback values for the port outputs. The fail-safe values will only be enabled if the EtherNet/IP connection is interrupted.

Available parameters:

Parameter	Description	Value range	Access
[Fail-safe digital out]	Drop-out value for the output on pin 4 (C/Q) for "Digital output (DO)" operating mode	• Reset: LOW level (default) • Old: Last active process value • Set: HIGH level	rw ¹
[Fail-safe IO-Link]	Fall-back value of the output on pin 4 (C/Q) for "IO-Link" operating mode	• Off: Fail-safe disabled (default) • Reset: Fail-safe enabled, reset process value • Old: Fail-safe enabled, last active process value • Pattern: Fail-safe enabled, byte sequence	rw ¹
[Fail-safe DO/UA]	Fall-back value of the output on pin 2 (I/Q) for "Digital output" operating mode ²	• Reset: LOW level (default) • Old: Last active process value • Set: HIGH level	rw ¹

¹ Read and write; parameter can only be changed if no connection to fieldbus controller is established

² Parameters only available for IO-Link ports (Class B)

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device is established (online).
- ✓ EtherNet/IP field bus is enabled.
- ✓ Configuration mode [Independent mode] is active.
- ▶ Select the [Port x] (x = 1...8) menu.
 - ▷ The menu page shows the current settings.
- ▶ Setting fail-safe values
- ▶ Write the changed values to the device.
- ▷ Fail-safe values are configured.

9.1.6 Process data

The dashboard area provides access to the device's process data.

To display the dashboard area:

- ✓ The parameter value editor for the device is open.
- In the vertical bar at the right side of the screen: Click on .
- ▷ The dashboard view is displayed.

To hide the dashboard area:

- ✓ The dashboard area is displayed.
- Click on the vertical bar in the middle of the screen: Click on .
- ▷ The dashboard view is hidden.

9.1.6.1 Reading digital input data

The input data can optionally be displayed in a time-value curve diagram.

Available data per port:

Date	Description	Value range	Access
iolinkmaster/port[x]/iqin	Status of the digital input on pin 4 of port x (x: 1...8) (operating mode: DI)	• 0: LOW • 1: HIGH	ro ¹

¹ Read only

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The dashboard area is displayed.
- Select the [Dashboard] tab.
- ▷ View shows the status of the digital input data.

9.1.6.2 Writing digital output data

Available information:

Element	Description	Value range	Access
iolinkmaster/port[x]/iqout	Status of the digital output on pin 4 of port x (x: 1...8) (operating mode: DO)	• Off: LOW • On: HIGH	rw ¹

¹ Read and write

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The dashboard area is displayed.
- ✓ Operating mode pin 4 (US) of the IO-Link port is digital output (DO).
- Set the initial value.
- Write the changed values to the device.
- ▷ Set initial value is valid.

9.1.6.3 Reading status information of the device

Available information:

Parameter	Description	Value range	Access
devicestatus/temperature/cpu	Temperature of the CPU (value in °C)	E.g. 37	ro ¹
devicestatus/temperature/power5v	Temperature of the power supply unit (value in °C)	E.g. 31	ro ¹
devicestatus/temperature/emmc	Temperature of the eMMC memory chip (value in °C)	E.g. 31	ro ¹
devicestatus/temperature/board	Temperature of the circuit board (value in °C)	e.g. 32.5	ro ¹

¹ Read only

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The dashboard area is displayed.
- Select the [Dashboard] tab.
- ▷ View shows the current status information of the device.

9.1.6.4 Reading timer counter values

Available information:

Parameter	Description	Value range	Access
timer[1]/counter	Counter reading of the timer timer[1]	0...4294967295	ro ¹
timer[2]/counter	Counter reading of the timer timer[2]	0...4294967295	ro ¹

¹ Read only

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device is established (online).
- ✓ The dashboard area is displayed.
- Select the [Dashboard] tab.
- ▷ The view shows the current counter values of the timers.

9.1.6.5 Reading port events

Port events are notifications of events that occur on the IO-Link ports of the device.

Available information:

Parameter	Description	Value range	Access
iolinkmaster/port[x]/portevent	Port event at port x (x: X1...X8) Structure: 0xYYZZ00 0xYY: Device Status - Status of the IO-Link devices 0xZZ: Connection Status - Status of the connection	Device status: 0x00: No IO-Link device connected 0x40: Wrong IO-Link device connected 0x80: IO-Link device connected and in PREOPERATE state 0xFF: IO-Link device connected and in OPERATE state Connection Status: 0x00: Port deactivated 0x01: Port activated, but no device connected 0x02: Port activated and in "IO-Link" operating mode 0x03: Port activated and in "DI" operating mode 0x04: Port enabled and in "DO" operating mode 0x1B: Repetitions detected 0x1E: Short circuit detected at pin 4 0x42: Wrong revision of the IO-Link device 0x43: Wrong vendor ID (V1.1 Revision) 0x44: Wrong device ID (V1.1 Revision) 0x45: Wrong vendor ID (V1.0 Revision) 0x46: Wrong device ID (V1.0 Revision) 0x48: Wrong cycle time	ro ¹

¹ Read only

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The dashboard area is displayed.
- Select the [Dashboard] tab.
- ▷ The view shows the current port events of the device's ports.

9.1.7 Device information

The [Info] menu provides access to the identification information of the device.

9.1.7.1 Reading identification information

Available information:

Name	Description	Value range	Access
[Order number]	Article number	AL1591	ro ¹
[Product code]	Product designation	IO-LINK MASTER SOL MP 8P IP69K	ro ¹
[Production date]	Production time (date, time)	e.g. "15.07.2024, 15:13:16.36"	ro ¹
[Vendor]	Vendor	ifm electronic gmbh	ro ¹
[SW-Revision]	Firmware version	E.g. 1.0.3.6	ro ¹
[HW-Verison]	Hardware version	E.g. 1.2.1.1	ro ¹
[HW-Revision]	Hardware version	E.g. AA	ro ¹
[Serial number]	Serial number	E.g. 000194610104	ro ¹

Name	Description	Value range	Access
[Fieldbus type]	Fieldbus	PROFINET / EtherNet/IP	ro ¹

¹ Read only

To read the device information:

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The detailed view of the device is active.
- ▶ Select [Info] menu.
- ▷ The menu page shows the identification information.

9.1.8 Device control

The [Firmware] menu provides access to functions for controlling the device.

9.1.8.1 Resetting the device

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The detailed view of the device is active.
- ▶ Select the [Firmware] menu.
- ▶ Click on [Factory reset].
- ▷ The device is reset to factory settings.
- ▷ All parameters are set to their default values.
- ▷ The device is read again.

9.1.8.2 Restarting the device

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The detailed view of the device is active.
- ▶ Select the [Firmware] menu.
- ▶ Click on [Reboot].
- ▷ The device is restarted.
- ▷ All set parameter values will be retained.

9.1.8.3 Identifying the device

The device offers a signalling function for visual identification.

Requirements:

- ✓ The parameter setting software has been started.
- ✓ The connection to the device has been established.
- ✓ The detailed view of the device is active.

- ▶ Select the [Firmware] menu.
- ▶ Click on [Blink] .
 - ▷ Status LEDs of the device flash (flashing duration: 60 s).
- ▷ After the fixed flashing time has elapsed, the status LEDs stop flashing automatically.

9.1.9 Updating the firmware



When updating to a firmware with a changed major or minor version (e.g. V1.x.y.z → V2.x.y.z or Vx.1.y.z → Vx.2.y.z), all settings are reset to the factory settings and all files and user-specific data are also deleted from the device.

- ▶ Back up all settings, files and user-specific data on the device before updating the firmware.

To update the firmware of the device:

Requirements:

- ✓ New firmware has been downloaded. documentation.ifm.com
- ✓ ifm moneo|configure has been started.
- ✓ The connection to the device is established (online).
- ✓ The detailed view of the device is active.
- ▶ In the header: Click on ⬆.
- ▷ A dialogue window appears.
- ▶ Click on [Import firmware file].
- ▷ The file explorer appears.
- ▶ Select the firmware file (.bin) and click on [Open].
- ▷ The new firmware file is selected.
- ▶ Click on [OK].
- ▷ The firmware of the device is updated.
- ▷ The status of the update process is displayed.
- ▷ The firmware of the device is updated.

9.2 IoT Core Visualizer

The IoT Core Visualizer provides a graphical user interface to access the functions of the ifm IoT Core.

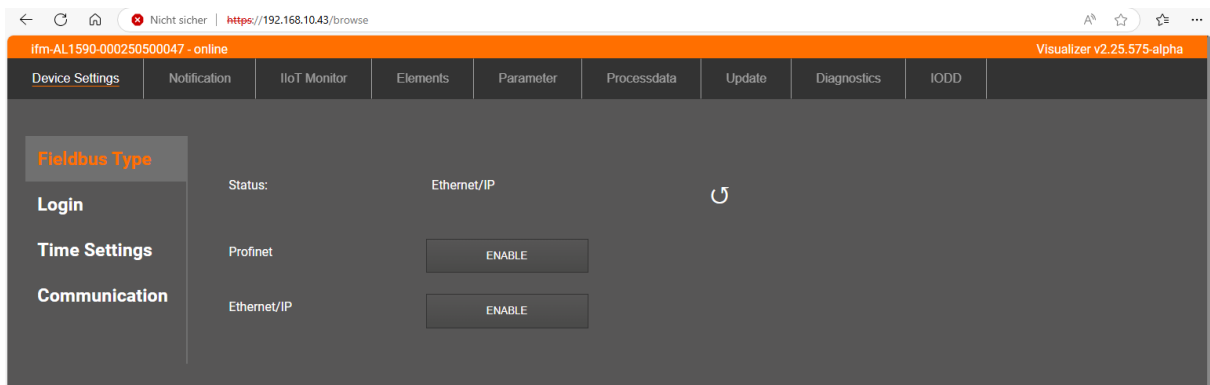
The ifm IoT Core Visualiser supports the following web browsers:

- Google Chrome, version 61 or higher
- Mozilla Firefox, version 93 or higher
- Microsoft Edge, version 107 or higher

9.2.1 Calling up the IoT Core Visualiser

Requirements:

- ✓ IIoT interface of the device has a valid IP configuration.
- ▶ Start web browser.
- ▶ Go to the following URL: `https://<ip-address>/web/subscribe` (z. B. `https://192.168.82.100/web/subscribe`)
 - ▷ The window for entering the login data appears.
- ▶ Enter user name and password (default values: User name: admin, password: password) and confirm entry.
- ▷ The web browser displays the IoT Core Visualizer start page.



The navigation menu at the top provides access to the following functions:

- [Device Settings]: [Device settings](#) (→ [44](#))
- [Notification]: [Notifications](#) (→ [54](#))
- [IIoT Monitor]: [IIoT Monitor](#) (→ [55](#))
- [Elements]: [IoT Core elements](#) (→ [57](#))
- [Parameter]: [Parameter](#) (→ [59](#))
- [Processdata]: [Process data](#) (→ [65](#))
- [Update]: [Update](#) (→ [70](#))
- [Diagnostics]: [Diagnostics](#) (→ [70](#))
- [IODD]: [IODD](#) (→ [71](#))

9.2.2 Device settings

The [Device settings] menu provides access to the basic settings of the device.

Requirements:

- ✓ IoT Core Visualizer has been started.

- Select [Device Settings] menu.
- ▷ The menu page shows the configuration options.

Available options:

- [Fieldbus type]: [Fieldbus settings](#) (→ [45](#))
- [Time Settings]: [System time settings](#) (→ [45](#))
- [Login]: [Authentication settings](#) (→ [47](#))
- [Communication]: [Communication settings](#) (→ [49](#))

9.2.2.1 Fieldbus settings

The device can be operated with one of the following fieldbuses:

- PROFINET
- EtherNet/IP



When the fieldbus type is changed, the active fieldbus connection is interrupted.



Changing the enabled fieldbus type changes the structure of the IoT Core tree.

- Reload the IoT Core Visualiser after changing the fieldbus type.

Available parameters:

Parameter	Description	Value range
[Status]	Active fieldbus type	• Changing mode...: The fieldbus settings are changed • Profinet: PROFINET is active • Ethernet/IP: EtherNet/IP is active

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- Select the [Fieldbus Type] submenu.
- ▷ The menu page shows the active fieldbus type.

To activate PROFINET:

- Next to [Profinet], click on [ENABLE].
- ▷ PROFINET is enabled.
- ▷ The Status field shows the change progress.
- ▷ After successful modification: A confirmation message appears.

To activate EtherNet/IP:

- Next to [Ethernet/IP], click on [ENABLE].
- ▷ EtherNet/IP is enabled.
- ▷ The Status field shows the change progress.
- ▷ After successful modification: A confirmation message appears.

9.2.2.2 System time settings

The user can customise the system time settings.

- ✓ IoT Core Visualizer has been started.

- ✓ The [Device Settings] menu is active.
- ▶ Select the [Time Settings] submenu.
- ▷ The menu page shows current system time.
- ▶ Optional: Click on [REFRESH].
- ▷ The system time is read and displayed again.

Available configuration options:

- [Setting the system time manually \(→ !\[\]\(01cfdf4da932ada5e695e1ddf76b8a01_img.jpg\) 46\)](#)
- [Using the system time of the laptop / PC \(→ !\[\]\(d0f5339c1571384449fe1558f2cdd63d_img.jpg\) 46\)](#)
- [Synchronising the system time with an NTP server \(→ !\[\]\(bef5b5a431bef79102d4d966a33136d1_img.jpg\) 47\)](#)

Setting the system time manually

The user can enter the system time in the following format:

Format	Description
YYYY-MM-DDTHH:mm:ss	• YYYY: Year • MM: Month • DD: Day • T: Separator between date and time • HH: Hour • mm: Minute • ss: Seconds

Example: 2025-04-26T06:31:25

To set the system time (date, time) of the device manually:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Time Settings] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Enable the option [Set time manually].
- ▶ Enter the required system time in the input field in the format YYYY-MM-DDTHH:mm:ss.
- ▶ Click on [SUBMIT].
 - ▷ A confirmation message appears.
- ▷ The system time entered is adopted.
- ▷ The menu page shows current system time.

Using the system time of the laptop / PC

The user can use the system time of the connected laptop / PC to set the system time of the device.

To adopt the system time of the laptop / PC:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Time Settings] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Enable the option [Use local system time].
- ▶ Click on [SUBMIT].

- ▷ If previously active: NTP service is stopped.
- ▷ Device adopts the system time of the connected laptop / PC.
- ▷ The menu page shows current system time.

Synchronising the system time with an NTP server

Alternatively, the user can use NTP servers from the Network Time Foundation for synchronisation.

The user can optionally use their own NTP server for synchronisation.

To synchronise the system time:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Time Settings] submenu.
- ▶ Enable the option [Use NTP server].
 - ▷ The current status of the NTP service is displayed.
- ▶ Optional: Enter the URL of the user-specific NTP server in the input field and click [ADD NTP URL] to add the URL to the list of NTP servers.
- ▶ Optional: Click on [X] next to a listed NTP server to delete the NTP server from the list.
- ▶ Click on [SUBMIT].
 - ▷ List of NTP servers is updated.
 - ▷ NTP service is started.
 - ▷ The current status of the NTP service is displayed.
- ▷ The system time of the device is synchronised with the NTP server.
- ▷ The menu page shows current system time.

9.2.2.3 Authentication settings

The user can customise the authentication settings.

Available parameters:

Parameter	Description	Value range
[User]	Current username	admin
[Password]	Current password	e.g. password (default)
[Enable]	Status of the authentication function	•  : function disabled •  : function enabled
[New password]	New password	
[Confirm new password]	Confirm new password	

To change the authentication settings:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Login] submenu.
- ▷ The menu page shows the current settings.

Supported configuration options:

- [Activating the authentication \(→ 48\)](#)
- [Disabling the authentication \(→ 48\)](#)

- [Changing the password \(→ 48\)](#)

Activating the authentication

The user can activate or disable authentication via login.



After reactivating the authentication function, the password is reset to the default value password.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ✓ Authentication is disabled.
- ▶ Select the [Login] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Enable the slider [Enable].
- ▶ Click on [SUBMIT].
 - ▷ A confirmation message appears.
- ▷ The authentication function is enabled.
- ▷ The default values apply for the user name and password ([User]: admin, [Password]: password).
- ▷ Access to IoT Core REST API and IoT Core Visualiser only works with valid authentication information.

Disabling the authentication

To disable the authentication function, the user must be logged in to the device.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ✓ Authentication is enabled.
- ▶ Select the [Login] submenu.
 - ▷ The menu page shows the current settings.
 - ▷ Slider [Enable] cannot be changed.
- ▶ Enter the user name in the [User] field: admin
- ▶ Enter the valid password in the [Password] field.
 - ▷ When valid login information is entered, the slider [Enable] can be changed.
- ▶ Disable the slider [Enable].
- ▶ Click on [SUBMIT].
 - ▷ A confirmation message appears.
- ▷ The authentication function is disabled.
- ▷ No authentication information is required to access the IoT Core REST API and IoT Core Visualiser.

Changing the password

The user can set a user-specific password. The user must enter the valid authentication information to make the change.

Requirements:

- ✓ IoT Core Visualizer has been started.

✓ The [Device Settings] menu is active.

► Select the [Login] submenu.

▷ The menu page shows the current settings.

► Enter the user name in the [User] field: admin

► Enter the valid password in the [Password] field.

Enter the desired password in the New password field.

Enter the desired password again in the Confirm new password field for confirmation.

► Click on [SUBMIT].

▷ A confirmation message appears.

▷ The new password is valid.

Once the password has been successfully changed, the IoT Core Visualiser login screen appears.

► Log in with the new password.

9.2.2.4 Communication settings

The user can customise the communication settings of the device.

Requirements:

✓ IoT Core Visualizer has been started.

✓ The [Device Settings] menu is active.

► Select the [Communication] submenu.

▷ The submenu shows available communication settings.

Supported communication protocols:

- [HTTP]: [Using the HTTP protocol \(→ □ 49\)](#)
- [HTTPS]: [Using the HTTPS protocol \(→ □ 50\)](#)
- [MQTT]:
- [OPC UA]: [Using the OPC UA protocol \(→ □ 52\)](#)

Using the HTTP protocol

The user can configure access to the device via the IIoT interface (ports XI1, XI2) using the HTTP protocol.

✓ IoT Core Visualizer has been started.

✓ The [Device Settings] menu is active.

✓ Submenu [Communication] is active.

► Select the [HTTP] submenu.

▷ The menu page shows the current settings.

To enable the HTTP protocol:

► Enable the slider [Enable].

► Click on [SUBMIT].

▷ HTTP protocol is enabled.

▷ The device can be reached via the HTTP protocol (port 80).

To disable the HTTP protocol:

► Disable the slider [Enable].

► Click on [SUBMIT].

- ▷ HTTP protocol is disabled.
- ▷ The device cannot be reached via the HTTP protocol.

Using the HTTPS protocol

The user can configure access to the device via the IIoT interface (ports XI1, XI2) using the HTTPS protocol. The HTTPS protocol is enabled at the factory. The HTTPS protocol cannot be disabled.

The user can optionally save an SSL certificate (*.pfx, *.pem) on the device to confirm its identity and enable a secure connection to be established.

To load an SSL certificate onto the device:

Requirements:

- ✓ SSL certificate for the device is available.
- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Communication] > [HTTPS] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Click on [CHOOSE FILE].
 - ▷ File explorer appears.
- ▶ Select SSL certificate and click [Open] to confirm the selection.
 - ▷ Selected file is displayed.
- ▶ Click on [SUBMIT].
 - ▷ SSL certificate is stored on the device.
 - ▷ HTTPS connections to the device are displayed as secure.

Configuring MQTT client

The device offers an MQTT client.

Available parameters:

Name	Description	Value range	Access
[Enable]	Status of the MQTT client	• : disable MQTT client • : enable MQTT client	rw ¹
[Version]	Supported MQTT standard	• V311: Version 3.11	ro ²
[QoS]	Quality of service level	• QoS0: QoS level 0 - PUBLISH (without confirmation) (default) • QoS1: QoS level 1 - PUBLISH > PUBREC (single confirmation) • QoS2: QoS level 2 - PUBLISH > PUBREC > PUBREL > PUBCOMP (double confirmation)	rw ¹
[Keep Alive]	Time period without data transmission after which a PINGREQ is sent to the MQTT broker (value in s) NOTE! MQTT broker responds to PINGREQ with PINGRESP	• 0: 0 s ... • 15: 15 s (default) ... • 60: 60 s	rw ¹
[Will Topic]	Fallback topic if connection is interrupted	E.g. failsafe	rw ¹
[Will Message]	Fallback message if connection is interrupted	z. B not valid	rw ¹
[Client ID]	Unique identifier of the MQTT client (max. 23 characters)		ro ²

Name	Description	Value range	Access
[Security]	Security settings optional or mandatory	- optional: Optional - mandatory: Mandatory	rw ¹
[Certificate]	Certificate (value as STRING)		rw ¹
[Key]	Personal key (value as STRING)		rw ¹
[Validate Server Certificate]	Validation option for MQTT broker certificates	- none: no validation - required: Mandatory validation - required_no_ca: Mandatory validation, not for certificates from certificate authorities (default)	rw ¹
[Trusted Server Certificates]	List of certificates from trusted MQTT brokers (multiple certificates separated by commas)		rw ¹

¹ Read and write

² Read only

To configure the MQTT client:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Communication] > [MQTT] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Navigate to the [Client] area.
- ▶ Set the parameters.
- ▶ Click on [SUBMIT].
- ▷ MQTT client is enabled with the selected parameter values.

To obtain the current status of the MQTT client:

- ▶ Click on [REFRESH].
- ▷ The menu page shows the current status of the MQTT client and the set parameter values.

Configuring the MQTT command channel

Available parameters:

Name	Description	Value range	Access
[Enable]	Status of the MQTT command channel	: disable MQTT command channel : enable MQTT command channel	rw ¹
[Broker IP]	IP address of the MQTT broker	e.g. 192.168.100.1	rw ¹
[Broker Port]	Port number of the MQTT broker	E.g. 1883 (default)	rw ¹
[Topic]	Command topic that MQTT clients monitor for requests from the ifm IoT Core	e.g. iotcore-request (default)	rw ¹
[Default Reply Topic]	Default reply topic if no reply topic is specified in the request	e.g. iotcore-reply (default)	rw ¹
[User]	User name for authentication on the MQTT broker	E.g. user1	rw ¹
[Password]	Password for authentication on the MQTT broker	E.g. passw0rd	rw ¹
[Version]	MQTT version	V311: Version 3.11	ro ²

Name	Description	Value range	Access
[QoS]	Quality of service level for responses to the MQTT broker	0: QoS0 – QoS level 0 - PUBLISH (without confirmation) (default) 1: QoS1 – QoS level 1 - PUBLISH > PUBREC (single confirmation) 2: QoS2 - QoS level 2 - PUBLISH > PUBREC > PUBREL > PUBCOMP (double confirmation)	rw ¹
[Keep Alive]	Time period without data transmission after which a PINGREQ is sent to the MQTT broker (value in s) NOTE! PLEASE NOTE! Please note: MQTT broker responds to PINGREQ with PINGRESP	0: 0 s ... 15: 15 s (default) ... 60: 60 s	rw ¹
[Will Topic]	Fallback topic if connection is interrupted	E.g. failsafe	rw ¹
[Will Message]	Fallback message if connection is interrupted	E.g. connection disrupted	rw ¹
[Client ID]	Unique identifier of the MQTT client (max. 23 characters)	E.g. d7d9212215904c93a52db151ae129	rw ¹
[Security]	Security settings optional or mandatory	- disabled: disabled (default) - required: obligatory	rw ¹
[Certificate]	Certificate of the MQTT client (value as STRING)		rw ¹
[Key]	Personal key of the MQTT client (value as STRING)		rw ¹
[Validate Server Certificate]	Validate certificates of the MQTT broker	- none: no validation - required: obligatory - required_no_ca: mandatory, no certificates from certificate authorities (default)	rw ¹
[Trusted Server Certificates]	Array with trusted certificates		rw ¹

¹ Read and write

² Read only

To configure the MQTT command channel:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ▶ Select the [Communication] > [MQTT] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Navigate to the [Command Channel] area.
- ▶ Set the parameters.
- ▶ Click on [SUBMIT].
- ▷ MQTT command channel is enabled with the selected parameter values.
- ▷ IoT Core establishes a connection to the MQTT broker.

To obtain the current status of the MQTT command channel:

- ▶ Click on [REFRESH].
- ▷ The menu page shows the current status of the MQTT command channel and the set parameter values.

Using the OPC UA protocol

The device can be operated as an OPC UA server.

An OPC UA connection can be used to retrieve process data as well as diagnostic and status information about the device from an OPC UA client.

Enabling or disabling the OPC UA server

The OPC UA server is disabled in the factory settings.

To enable the OPC UA server:

- ▶ Enable the slider [Enable].
- ▶ Click on [SUBMIT].
- ▷ OPC UA server of the device is enabled.
- ▷ An OPC UA client can access the device with the set parameters.

To disable the OPC UA server:

- ▶ disable the slider [Enable].
- ▶ Click on [SUBMIT].
- ▷ OPC UA server is disabled.

Configuring the OPC UA server

Available parameters:

Name	Description	Value range	Access
[Port]	Port for communication with OPC UA client	Default: 4843	rw ¹
[Supported security schemes]	Supported security policies: • Basic256Sha256 – Encryption: AES256 – Data integrity: SHA-256 • Aes128Sha256RsaOaep – Encryption: AE128 – Data integrity: SHA-256 – Confidentiality: RSA-OAEP • Aes256Sha256RsaPss – Encryption: AES256 – Data integrity: SHA-256 – Confidentiality: RSA-PSS NOTE! At least one security policy must be active!	• : disable security policy • : Enable security policy	rw ¹

¹ Read and write

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ✓ OPC UA server is disabled.
- ▶ Select the [OPC UA] submenu.
- ▶ Set the parameters of the OPC UA server.
- ▶ Click on [SUBMIT].
 - ▷ Parameters are set.
- ▶ Enable the OPC-UA server.
- ▷ OPC UA server is started with the set parameters.

Storing certificates and private keys on the device

The device supports the following protective measures:

- Private keys to ensure the authenticity and confidentiality of data (file format: *.pfx - Personal Information Exchange)
- Certificates for the encryption of the connection (file format: *.der - Distinguished Encoding Rules)

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Device Settings] menu is active.
- ✓ OPC UA server of the device is disabled.
- ▶ Select the [OPC UA] submenu.
- ▶ Click on CHOOSE FILE next to the [Upload private key from local PC] field.
- ▶ Click on CHOOSE FILE next to the [Upload certificate from local PC] field.
- ▶ Click on [SUBMIT].
 - ▷ Selected files are transferred to the device and saved.
- ▶ Enable the OPC-UA server.
- ▶ Establish connection between OPC UA client and OPC UA server.

9.2.3 Notifications

The [Notification] menu page provides access to the notification function of the ifm IoT Core.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ▶ Select [Notification] menu.
- ▷ The menu page shows the active notifications.

CID	Channel	Broker/Server	Event	Data	Is Persistent	Unsubscribe
3	http	192.168.10.1:80/dump	ifm-AL1590-000242900022/timer[1]/counter/datachanged	ifm-AL1590-000242900022/iolinkmaster/port[8]/iolinkdevice/pdin ifm-AL1590-000242900022/devicestatus/temperature/cpu	✓	

The following information is displayed for each active notification:

Name	Description	Value range
[CID]	Identifier of the notification order	E.g. 14
[Channel]	Communication protocol used for the transmission of the notification	• http: HTTP protocol • mqtt: MQTT protocol • mqttts: MQTTS protocol • ws: WebSocket protocol
[Broker/Server]	Target system (URL incl. topic)	E.g. 192.168.10.1:80/datadump
[Event]	Event that triggers the notification	E.g. /timer[1]/counter/datachanged
[Data]	Data points whose values are transmitted with the notification	E.g. /fieldbussetup/connectionsstatus

Available options:

- [Create new notifications](#) (→ [55](#))
- [Delete notifications](#) (→ [55](#))

9.2.3.1 Create new notifications

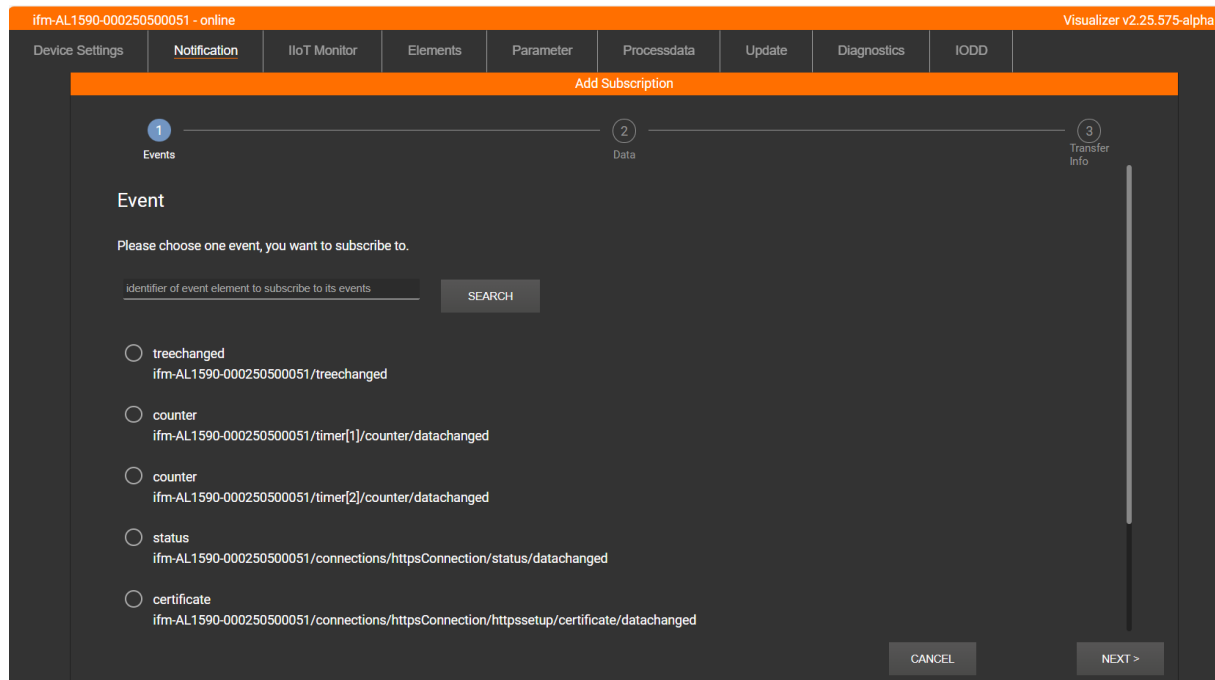
New notifications are created using a wizard.



For time-controlled notifications via timer 1 or timer 2, the user must set the interval time of the relevant timer and start the timer.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ Menu [Notification] is active.
- Click on [ADD NEW...].
 - ▷ The wizard for creating notifications appears.



- Use the wizard to select the desired notification parameters step by step.
- ▷ The menu page shows the created notification.

9.2.3.2 Delete notifications

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ Menu [Notification] is active.
- ✓ At least one notification has been created.
- Click on  in column [Unsubscribe] in the notification row.
- ▷ The selected notification is deleted.

9.2.4 IIoT Monitor

The [IIoT Monitor] menu provides access to the device's monitoring options.

Requirements:

- ✓ IoT Core Visualizer has been started.
- Select [IIoT Monitor] menu.
- ▷ The menu page shows the active monitoring tasks in a tabular overview.

ifm-AL1590-000250500051 - online							Visualizer v2.25.575-alpha	
Device Settings	Notification	IoT Monitor	Elements	Parameter	Processdata	Update	Diagnostics	IODD
ADD NEW...								
ID	Callback	Sample Interval [ms]	Notify Interval [ms]	Data	Only Changed Values	Is Persistent	Unsubscribe	
5	http://192.168.10.100/datadump	10000	100000	ifm-AL1590-000250500051/iolinkmaster/port[2]/iolinkdevice/pdin	-	✓		

The following information is shown for each monitoring task:

Name	Description	Value range
[ID]	Identifier of the monitoring task	0...2147483647
[Callback]	Destination address for notifications (URL format)	E.g. http://192.168.10.100/datadump
[Sample Interval [ms]]	Sampling interval (value in ms)	e.g. 10000: 10 s
[Notify Interval [ms]]	Notification interval (value in ms)	e.g. 100000: 100 s
[Data]	List of monitored data points	E.g. /iolinkmaster/port[2]/iolinkdevice/pdin
[Only Changed Values]	Frequency of data storage	-: Save all values ✓: Only save a new value if it has changed from the previous value
[Is Persistent]	Storage duration of the monitoring task	-: Active until the next restart of the device ✓: Permanently active
[Unsubscribe]	Delete monitoring task	: Delete monitoring task

Available options:

- [Creating a new monitoring task \(→ 56\)](#)
- [Deleting a monitoring task \(→ 57\)](#)

9.2.4.1 Creating a new monitoring task

New monitoring tasks are created using a wizard.

Available parameters:

Name	Description	Value range	Access
[Subscription ID]	Identification of the monitoring task	0...2147483647 -1: Automatic allocation of the identifier	rw ¹
[Notification Receiver]	Target address to which the notifications are to be sent (value in URI format)	E.g. http://192.168.10.100/datadump	rw ¹
[Notification Interval [ms]]	Interval at which a notification is to be sent with the monitored data values (value in ms)	e.g. 10000: 10 s	rw ¹
[Sample Interval [ms]]	Interval at which the data values are recorded (value in ms)	e.g. 3000: 3 s	rw ¹
[Only changed values]	Frequency of saving the data values	: Transfer all recorded data values : Only transfer data value if there is a change to the last recorded value	rw ¹

Name	Description	Value range	Access
[Shall be persistent]	Period of validity of the monitoring task	: Monitoring task active as long as device is switched on : Monitoring task active even after restarting the device	rw ¹

¹ Read and write

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [IIoT Monitor] menu is active.
- ▶ Click on [ADD NEW...].
 - ▷ The input screen for creating a new monitoring task appears.
- ▶ Enter the required parameter values for the monitoring task in the input screen (→ Table).
- ▶ Select the data points to be monitored from the list. If required, filter the displayed data points ([All], [Parameter], [Processdata]).
- ▶ Click on [SUBMIT].
 - ▷ A monitoring task with the selected parameters is created.
 - ▷ The previous view appears.
 - ▷ Created monitoring is displayed in the overview view.
 - ▷ The created monitoring task is active.

OR:

- ▶ Click on [CANCEL].
 - ▷ The creation of the monitoring task is cancelled.
 - ▷ The previous view appears.

9.2.4.2 Deleting a monitoring task

Requirements:

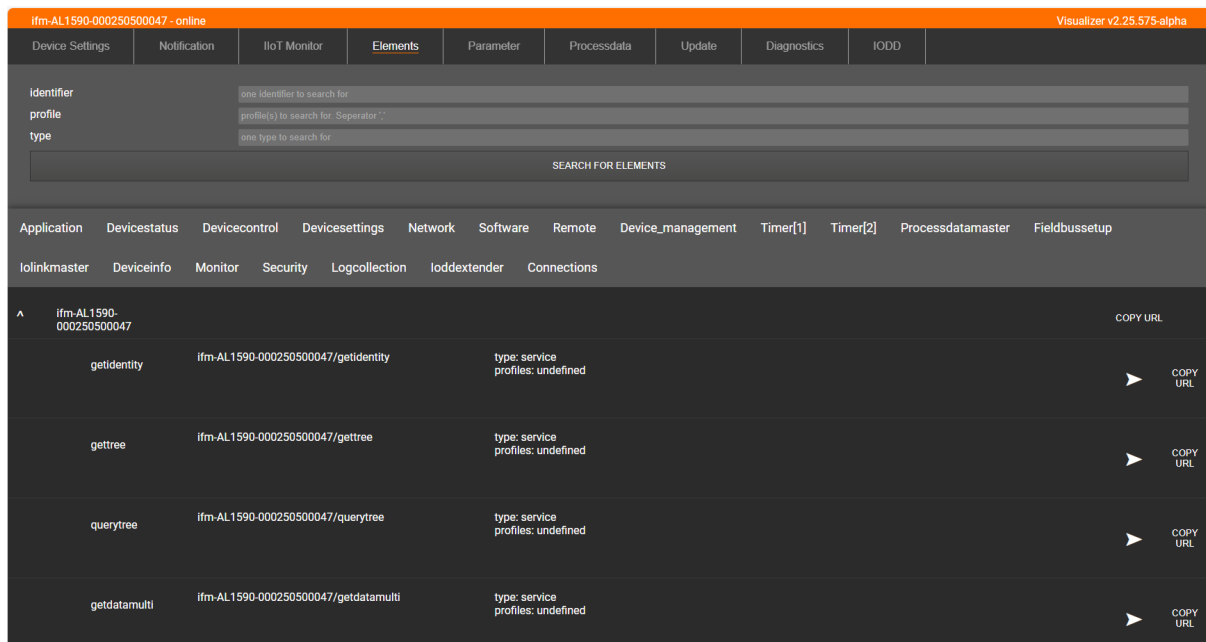
- ✓ IoT Core Visualizer has been started.
- ✓ The [IIoT Monitor] menu is active.
- ✓ At least one monitoring task has been created.
- ▶ Click on  in the [Unsubscribe] column in the monitoring task row.
 - ▷ Selected monitoring task is deleted.

9.2.5 IoT Core elements

The [Elements] menu provides access to the elements of the ifm IoT Core of the device.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ▶ Select [Elements] menu.
 - ▷ The menu page shows the tree view of the ifm IoT Core of the device.



The view shows the following information:

- Input mask for the search function
- Structural elements (type: structure) of the 1st level of the ifm IoT Core
- Name of the element
- Path of the element in the ifm IoT Core
- Type ([type]) and profile ([profiles]) of the element
- Data type ([Type]), coding ([Encoding]) of the data points
- Valid values ([valuelist]) or minimum value ([min]), maximum value ([max]) and default value ([default]) of a data point

Notes on use:

- ▶ Search the ifm IoT Core of the device for specific elements, [Searching for elements of the ifm IoT Core \(→ 58\)](#).
- ▶ Click on $\vee/\wedge$ to expand/collapse the tree view.
- ▶ Click [COPY URL] to copy the path of the element to the clipboard.
- ▶ Click on $\triangleright$ to execute the service, [Executing services of the ifm IoT Core \(→ 59\)](#).

9.2.5.1 Searching for elements of the ifm IoT Core

The user can search the ifm IoT Core for elements with certain properties and output the search results.

Available search criteria:

- [identifier]: Name of the element
- [profile]: Profile of the element
- [typ]: Type of the element

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Elements] menu is active.
- ▶ Select the search criteria in the [identifier], [profile] and [type] selection lists.
- ▶ Click on [SEARCH FOR ELEMENTS].

- ▶ IoT Core Visualiser searches the IoT Core for elements with the selected search criteria.
- ▷ The result list shows all the elements found.
- ▷ Elements with the selected search criteria are highlighted in colour.

9.2.5.2 Executing services of the ifm IoT Core

The user can execute individual services in the tree view of the ifm IoT Core.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Elements] menu is active.
- ▶ Navigate to the service in the tree view of the IoT Core.
- ▶ Click on > in the service line.
 - ▷ A dialogue window appears.
- ▶ For services with request parameters: Enter the request parameters in JSON format in the input field. Details on enquiry parameters: [Services](#) (→ [161](#))
- ▶ Click on [REQUEST].
- ▷ The service is running.
- ▷ The dialogue window shows the response to the request.

Example: Setting the interval of the timer

Task: The interval of timer 1 should be set to 20 seconds.

Solution: In the IoT Core Visualiser, write the value 20000 to the data point /timer[1]/interval using the setdata service.

- ✓ IoT Core Visualizer has been started.
- ✓ The [Elements] menu is active.
- ▶ Navigate to the /timer[1]/interval/setdata element in the tree view of the device's IoT Core.
- ▶ Click on > in the service line.
 - ▷ A dialogue window appears.
- ▶ Insert the following code into the input field:

```
{"newValue":20000}
```

- ▶ Optional: Click [VALIDATE JSON] to check the entered code for correctness.
- ▶ Click on [REQUEST].
- ▷ The service is running.
- ▷ The dialogue window shows the response to the request.

9.2.6 Parameter

The Parameter menu page provides access to the configuration options of the device.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ▶ Select [Parameter] menu.
- ▷ The menu page shows the available parameters of the device.

BILD

Notes on operation:

Each parameter displayed corresponds to a data point of type parameter of the ifm IoT Core of the device. A parameter can be read-only or readable and changeable.

The following information is displayed for each data point:

- [Type]: Data type, e.g. structure, string, uint32
- [Namespace]: Namespace, e.g. json
- [Encoding]: Coding of the data value
- [Valuation]: Value range, e.g. min: 50, max: 3600000

Various functions can be executed on each data point:

- Click  to read the data point again.
- Click [Copy URL] to copy the URL of the data point to the clipboard.
- Change a data value and then click  to apply the changes.

9.2.6.1 Configuring the IIoT interface

The [Network] submenu provides access to the configuration options of the IIoT interface.

Available parameters:

Name	Description	Value range	Access
[br0] > [macaddress]	MAC address of the interface	E.g. 00:02:01:10:4F:BD	ro ¹
[br0] > [hostname]	Host name of the device Default value: ifm-artno-serial • ifm: supplier name • artno: Article number of the device • serial: Serial number of the device	e.g. ifm-AL1591-000242900022	rw ²
[br0] > [ipv4] > [mode]	Desired IP configuration mode	• Static: Manual IP configuration • DHCP: Receive IP configuration from DHCP server (default) • link-local: IP configuration via link-local protocol	rw ²
[br0] > [ipv4] > [gateway]	IP address of the network gateway	E.g. 192.168.10.1	rw ³
[br0] > [ipv4] > [address0]	Structure for IP addressing		
[br0] > [ipv4] > [address0] > [mode]	Set IP configuration mode	• Static: Manual IP configuration • DHCP: Receive IP configuration from DHCP server • link-local: IP configuration via link-local protocol	ro ¹
[br0] > [ipv4] > [address0] > [address]	IP address of the interface	E.g. 192.168.10.43	rw ^{2 3}
[br0] > [ipv4] > [address0] > [subnetmask]	Subnet mask of the network segment	E.g. 255.255.255.0	rw ^{2 3}
[br0] > [ipv4] > [dnsservers]	Set DNS server	E.g. ["192.168.10.100"]	ro ¹
[br0] > [ipv4] > [staticdnsservers]	Static DNS server List with comma-separated IP addresses • Input scheme: ["ip1", "ip2", ...] • Empty list: []	E.g. ["91.239.100.100", "89.233.43.71"]	rw ²

¹ Read only

² Read and write

³ Can only be changed if static addressing is active

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- ▶ Select the [Network] submenu.
 - ▷ The menu page shows the current settings.

Manual IP configuration:

- ▶ Set parameter [br0] > [ipv4] > [mode] to value Static .
- ▶ Click on  to save the changes to the device.
 - ▷ Manual IP configuration is active.
- ▶ Set parameters [address] and [subnetmask] .
- ▶ Next to substructure [address0], click on  to save the changes on the device.
 - ▷ Static IP configuration is active.
 - ▷ The menu page shows the current IP configuration.

IP configuration via DHCP:

- ▶ Set parameter [br0] > [ipv4] > [mode] to value DHCP.
- ▶ Click on  to save the changes to the device.
 - ▷ IP configuration via DHCP is active.
- ▷ IIoT interface obtains IP configuration from DHCP server.
- ▷ The menu page shows the current IP configuration.

IP configuration via link-local protocol:

- ▶ Set parameter [br0] > [ipv4] > [mode] to value link-local.
- ▶ Click on  to save the changes to the device.
 - ▷ IP configuration via link-local protocol is active.
- ▷ IIoT interface obtains IP configuration from defined address range.
- ▷ The menu page shows the current IP configuration.

9.2.6.2 Configuring the timer

The [Timer[1]] and [Timer[2]] submenus provide access to the settings for timers 1 and 2.

Available parameters:

Parameter	Description	Value range	Access
[interval]	Counting interval of the timer (value in ms)	• 50: 50 ms ... • 10000: 10000 ms (default) ... • 3600000: 3600000 ms	rw ¹

¹ Read and write

To configure the timers:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- ▶ Select the [Timer[1]] > [Timer[2]] submenu.
 - ▷ The menu page shows the current settings.

- Set the parameters.
- Click on  to save the changes to the device.
- ▷ Timers of the device are configured.

9.2.6.3 Configuring the fieldbus interface

The [Fieldbussetup] submenu provides access to the current status of the fieldbus interface.

Available parameters:

Name	Description	Value range	Access
[connectionstatus]	Status of the fieldbus connection	0: Disconnected – connection timeout 1: Connected – connection established	ro ¹
[fieldbustype]	Active fieldbus	0: Profinet – PROFINET active 2: EtherNet/IP - EtherNet/IP active	ro ¹

¹ Read only

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- Select the [Fieldbussetup] submenu.
- ▷ The menu page shows the current status of the fieldbus interface.

Available options:

- [PROFINET: Reading the IP configuration \(→ !\[\]\(591a3696e5f3ca3c5de70cfd4ae1f01f_img.jpg\) 62\)](#)
- [EtherNet/IP Reading the IP configuration \(→ !\[\]\(ae56479749f7ad8ecea91a8332d3b1af_img.jpg\) 63\)](#)
- [EtherNet/IP Setting the configuration mode \(→ !\[\]\(05fd94ee87e0706d8ab5817a137bfe02_img.jpg\) 63\)](#)

PROFINET: Reading the IP configuration

The IP configuration of the PROFINET interface can only be set via the PROFINET configuration software.

Available data points:

Name	Description	Value range	Access
[ipaddress]	IP address of the PROFINET interface	e.g. 192.168.10.100 • 0.0.0.0 (default)	ro ¹
[subnetmask]	Subnet mask of the PROFINET network	e.g. 255.255.255.0 • 0.0.0.0 (default)	ro ¹
[ipdefaultgateway]	IP address of the network gateway	e.g. 192.168.10.1 • 0.0.0.0 (default)	ro ¹
[macaddress]	MAC address of the device's PROFINET interface	e.g. 00:02:01:0f:c8:8f	ro ¹
[hostname]	Station Name of the device (max. 240 characters)	E.g. iol.plant1	ro ¹
[fieldbusfirmware]	Version of the PROFINET firmware	E.g. 1.1	ro ¹

¹ Read only

To display the IP configuration of the PROFINET interface:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.

- ✓ PROFINET is set as the active fieldbus type.
- Select the [Fieldbussetup] > [Network] submenu.
- ▷ The menu page shows current IP configuration of the PROFINET interface.

EtherNet/IP Reading the IP configuration

The IP configuration of the Ether/Net-IP interface can only be set via the EtherNet/IP configuration software.

Available parameters:

Parameter	Description	Value range	Access
[dhcp]	Method for IP address assignment	• 0: Static - manual address assignment • 1: DHCP - address assignment by DHCP server • 2: BOOTP - address assignment via bootstrap protocol	ro ¹
[ipaddress]	IP address of the EtherNet/IP interface	e.g. 192.168.10.100 • 192.168.1.250 (default)	ro ¹
[subnetmask]	Subnet mask of the EtherNet/IP network	e.g. 255.255.255.0 • 255.255.255.0 (default)	ro ¹
[ipdefaultgateway]	IP address of the network gateway	e.g. 192.168.10.1 • 0.0.0.0 (default)	ro ¹
[hostname]	EtherNet/IP name of the device (max. 240 characters)	E.g. iol.plant1	ro ¹
[macaddress]	MAC address of the device's EtherNet/IP interface	e.g. 00:02:01:0f:c8:8f	ro ¹
[fieldbusfirmware]	Version of the EtherNet/IP firmware	e.g. 3.4.0.7 (EtherNet/IP adapter)	ro ¹

¹ Read only

To display the IP configuration of the EtherNet/IP interface:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- ✓ EtherNet/IP field bus is enabled.
- Select the [Fieldbussetup] > [Network] submenu.
- ▷ The menu page shows current IP configuration of the EtherNet/IP interface.

EtherNet/IP Setting the configuration mode

The device supports the EtherNet/IP configuration modes Top-Down and Independent.

For the configuration mode Independent, the user must set the length of the process data and the arrangement of the bytes within the process data. Optionally, fallback values can be set in case the connection to the EtherNet/IP controller is interrupted.

Available parameters:

Name	Description	Value range	Access
[independentmode]	EtherNet/IP configuration mode	• 0: Off - top-down mode active -- configuration via EtherNet/IP PLC • 1: On - independent mode active - configuration via device settings	rw ¹

Name	Description	Value range	Access
[processdataconfiguration]	Length of IO-Link process data per port	0: 2 Bytes Input 2 Bytes Output - 2 bytes input data, 2 bytes output data 1: 4 Bytes Input 4 Bytes Output - 4 bytes input data, 4 bytes output data: 2: 8 Bytes Input 8 Bytes Output - 8 bytes input data, 8 bytes output data 3: 16 Bytes Input 16 Bytes Output - 16 bytes input data, 16 bytes output data 4: 32 Bytes Input 32 Bytes Output - 32 bytes input data, 32 bytes output data	rw ¹
[swap]	Arrangement of the bytes in process data	- as Array of Bytes 1: Enabled - as an integer16 value; When the process data is updated, the bytes in the integer16 value are swapped	rw ¹
[port[x]] > [failsafeiolink]	Fallback value for IO-Link process data on pin 4 (C/Q) of port x (x: 1...8), operating mode: - IO-Link.	0: Off - fail-safe disabled (default) 1: Reset - fail-safe enabled, reset value 2: Old - fail-safe enabled, last active process value 3: Pattern - fail-safe enabled, byte sequence	rw ²
[port[x]] > [failsafedigitalout]	Drop-out value of the digital output on pin 4 (C/Q) of port x (x: 1...8), operating mode: Digital output (DO):	0: Reset - LOW level 1: Old - last active process value 2: Set - HIGH level	rw ¹
[port[x]] > [failsafeiqout]	Drop-out value for digital output on pin 2 (I/Q) of port x (x: 1...4), operating mode: Digital output	0: Reset - LOW level (default) 1: Old - last active process value 2: Set - HIGH level	rw ¹

¹ Read and write; parameter can only be changed if no connection to fieldbus controller is established

² Read and write

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- ✓ EtherNet/IP is set as the active fieldbus type.
- ▶ Select the [Fieldbussetup] > [Configuration] submenu.
 - ▷ The menu page shows the current settings.
- ▶ Select the configuration mode.
- ▶ For the configuration mode Independent: Set additional parameters.
- ▶ Click on  to save the changes to the device.
- ▷ The selected configuration mode is active.

9.2.6.4 Reading identification information

The [Deviceinfo] menu provides access to the device's identification information.

Available parameters:

Name	Description	Value range	Access
[serialnumber]	Serial number	E.g. 000194610104	ro ¹
[productname]	Product name	IO-LINK MASTER SOL MP 8P IP69K	ro ¹

Name	Description	Value range	Access
[productcode]	Product designation	AL1591	ro ¹
[ordernumber]	Article number	AL1591	ro ¹
[productiondate]	Production date (format: DD.MM.YYYY, hh:mm:ss)	E.g. 15.07.2024, 13:16.36	ro ¹
[hwrevision]	Hardware version	E.g. AA	ro ¹
[swrevision]	Firmware version	E.g. 1.0.3.6	ro ¹
[vendor]	Vendor	ifm electronic gmbh	ro ¹
[swversion]	Software version	E.g. 1.0.3.6	
[hwversion]	Hardware version	E.g. 1.2.1.1	ro ¹
[fieldbustype]	Fieldbus type:	- Profinet: PROFINET - EtherNet/IP: EtherNet/IP	ro ¹

¹ Read only

To access the identification information:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- Select the [deviceinfo] submenu.
- ▷ The menu page shows the identification information of the device.

9.2.6.5 Reading software information

The [Software] submenu provides access to status information on the device's software components.

Available parameters:

Name	Description	Value range	Access
[firmware]	Container for firmware		
[firmware] > [container] > [chunk-size]	Size of a sub-segment of the software container (value in bytes)	1048576	ro ¹
[firmware] > [container] > [max-size]	Size of the software container (value in bytes)	586956800	ro ¹
[firmware] > [version]	Version of the software component	E.g. IO Link Master 1.1.0.1	ro ¹
[firmware] > [type]	Type of software component	firmware	ro ¹

¹ Read only

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Parameter] menu is active.
- Select the [Software] submenu.
- ▷ The menu page shows information about the software components of the device.

9.2.7 Process data

Notes on operation:

9.2.7.1 Reading temperature values

Available information:

Name	Description	Value range	Access
[cpu]	Temperature of the CPU (value in °C)	E.g. 37	ro ¹
[power5v]	Temperature of the power supply unit (value in °C)	E.g. 31	ro ¹
[emmc]	Temperature of the eMMC memory chip (value in °C)	E.g. 31	ro ¹
[board]	Temperature of the circuit board (value in °C)	e.g. 32.5	ro ¹

¹ Read only

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Processdata] menu is active.
- Select the [Devicestatus] > [Temperature] submenu.
- ▷ The menu page shows the temperature values of the appliance components.

9.2.7.2 Reading timer counter readings

Available information:

Parameter	Description	Value range	Access
[counter]	Counter reading of the timer	0...4294967295	ro ¹

¹ Read only

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Processdata] menu is active.
- Call up substructure [timer[1]] or [timer[2]].
- ▷ The menu page shows the current counter reading of the selected timer.

9.2.7.3 Reading the status and diagnostic data of the device

Available information:

Parameter	Description	Value range	Access
[voltage]	Voltage value of device supply US (value in mV)	0...40000	ro ¹
[voltage] > [unit]	Unit of voltage	mV	ro ¹
[current]	Current value of the device supply US (value in mA)	0...40000	ro ¹
[current] > [unit]	Unit of current	mA	ro ¹
[supervisionstatus]	Status of the US device supply	• 0: OK - no error • 1: WARNING: undervoltage - warning - undervoltage • 2: WARNING: overvoltage - overvoltage • 3: FAULT - error	ro ¹
[supervisionstatus_ua]	Status of the UA device supply	• 0: OK - no error • 1: WARNING: undervoltage - warning - undervoltage • 2: WARNING: overvoltage - overvoltage • 3: FAULT - error	ro ¹

¹ Read only

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Processdata] menu is active.
- Select the [Processdatamaster] submenu.
- ▷ The menu page shows the status and diagnostic information of the device.

9.2.7.4 Reading and writing process data

Available parameters:

- IO-Link ports (class A)

Name	Description	Value range	Access
[iqin]	Digital input (pin 2)	• 0: LOW • 1: HIGH	ro ¹
[iolinkdevice] > [pdin]	Input data (pin 4)	Operating mode DI • 0x00: LOW • 0x1: HIGH Operating mode IO-Link • Input value in hexadecimal representation	ro ^{1 2}
[iolinkdevice] > [pdout]	Output data (pin 4)	Operating mode DO • 0x00: LOW • 0x1: HIGH Operating mode IO-Link • Output value in hexadecimal representation	rw ^{3 4}

¹ Read only

² Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to the port

³ Read and write

⁴ Read and write; parameter can only be changed if no connection to fieldbus controller is established

- IO-Link ports (class B):

Name	Description	Value range	Access
[iqout]	Digital output (pin 2)	• 0: LOW • 1: HIGH	rw ^{1 2}
[iolinkdevice] > [pdin]	Input data (pin 4)	Operating mode DI • 0x00: LOW • 0x1: HIGH Operating mode IO-Link • Input value in hexadecimal representation	ro ^{3 4}
[iolinkdevice] > [pdout]	Output data (pin 4)	Operating mode DO • 0x00: LOW • 0x1: HIGH Operating mode IO-Link • Output value in hexadecimal representation	rw ^{1 2}

¹ Read and write

² Read and write; parameter can only be changed if no connection to fieldbus controller is established

³ Read only

⁴ Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to the port

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Processdata] menu is active.
- Select the [Iolinkmaster] > [port[x]] submenu (x: 1...8).
- ▷ The menu page shows input and output data
- Optional: Enter the desired value in the output data fields.

► Click on  to save the changes to the device.

▷ The configured process values are set.

9.2.7.5 Reading IO-Link events

The device supports IO-Link events. IO-Link events are event and error messages. IO-Link events can be generated in the IO-Link master and in the connected IO-Link devices. IO-Link events generated in the IO-Link devices are forwarded to the IO-Link master and stored there.

Available parameters:

Name	Description	Value range	Access
[iolinkevent]	Active IO-Link event Structure of an IO-Link event (7 bytes): 0xIIMTTVVSSCCCC • 0xII: Instance (IO-Link event qualifier - event trigger) • 0xMM: Mode (IO-Link event qualifier - event mode) • 0xTT: Type (IO-Link event qualifier - event type) • 0xVV: Validity (IO-Link event qualifier - validity) • 0xSS: Source (IO-Link event qualifier - event source) • 0xCCCC: Event Code (IO-Link event code)	Instance: • 0x00: unknown • 0x01: PL (Physical Layer) • 0x02: DL (Data Layer) • 0x03: AL (Application Layer) • 0x04: APPL (Application) Mode: • 0x40: One-time event or warning (single shot) • 0xC0: Error disappeared • 0x80: Error appeared Type: • 0x10: Notification • 0x20: Warning • 0x30: Error Validity: • 0x00: valid • 0x40: invalid Source: • 0x00: IO-Link device • 0xFF: IO-Link master Event Code: • → IO-Link specification	ro ^{1 2}
[iolinkeventlist]	Array with IO-Link events that have occurred	Per IO-Link event: • → Parameter [iolinkevent]	ro ^{1 2}

¹ Read only

² Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to the port

Requirements:

✓ IoT Core Visualizer has been started.

✓ The [Processdata] menu is active.

► Select the [Iolinkmaster] > [port[x]] > [iolinkdevice] submenu (x: 1...8).

▷ The menu page shows the IO-Link events of the port.

9.2.7.6 Reading port events

A port event contains information about events on the IO-Link port (e.g. IO-Link device connected or disconnected, change of the operating mode of the port).

Available parameters:

Name	Description	Value range	Access
[portevent]	Port event Structure: 0x00ZZYY 0xYY: Device Status - status of the IO-Link device 0xZZ: Connection Status - status of the connection	DeviceStatus: 0x00: No IO-Link device connected 0x40: Wrong IO-Link device connected 0x80: IO-Link device connected and in PREOPERATE state 0xFF: IO-Link device connected and in OPERATE state Connection Status: 0x00: Port deactivated 0x01: Port activated, but no device connected 0x02: Port activated and in "IO-Link" operating mode 0x03: Port activated and in "DI" operating mode 0x04: Port enabled and in "DO" operating mode 0x1B: Repetitions detected 0x1E: Short circuit detected at pin 4 0x42: Wrong revision of the IO-Link device 0x43: Wrong vendor ID (V1.1 Revision) 0x44: Wrong device ID (V1.1 Revision) 0x45: Wrong vendor ID (V1.0 Revision) 0x46: Wrong device ID (V1.0 Revision) 0x48: Wrong cycle time	ro ¹
[porteventlist]	Array with port events that have occurred	Per port event: → Parameter [portevent]	ro ¹

¹ Read only

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Processdata] menu is active.
- Select the [Iolinkmaster] > [port[x]] submenu (x: 1...8).
- ▷ The menu page shows the port events of the port.

9.2.7.7 Reading and writing information about IO-Link devices

Available parameters:

Name	Description	Value range	Access
[vendorid]	Vendor ID of the IO-Link device	0...65535	ro ¹
[deviceid]	Device ID of the IO-Link device	0...16777215	ro ¹
[productname]	Article number of the IO-Link device	E.g. TCC501	ro ¹
[reversion]	IO-Link ID of the IO-Link device	0: 1.0 - standard 1.0 1: 1.1 - standard 1.1	ro ^{2 3}
[serial]	Serial number of the IO-Link device (12 digits)	E.g. 000008500706	ro ¹

Name	Description	Value range	Access
[status]	State of the IO-Link device	0: not connected - not connected 1: preoperate - PREOPERATE state 2: operate - OPERATE state 3: communication error - communication error 4: deactivated - port disabled 5: digital input - digital input 6: digital output – digital output 7: power off - supply voltage switched off 8: not available - not available	ro ¹
[applicationspecifictag]	Application-specific tag	***: *** (Default)	rw ^{4 3}
[functiontag]	Function identifier	E.g. temperature1	rw ^{4 3}
[locationtag]	Location identifier	E.g. plant1	rw ^{4 3}

¹ read only; Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to port

² Read only

³ Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to the port

⁴ Read and write

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [Processdata] menu is active.
- ▶ Select the [Iolinkmaster] > [port[x]] > [iolinkdevice] submenu (x: 1...8).
- ▷ The menu page shows information about the IO-Link device.
- ▶ Optional: Enter device-specific identifiers.
- ▶ Click on  to save the changes to the device.

9.2.8 Update

The [Update] menu page allows you to update the firmware of the device:

9.2.8.1 Firmware update



When updating to a firmware with a changed major or minor version (e.g. V1.x.y.z → V2.x.y.z or Vx.1.y.z → Vx.2.y.z), all settings are reset to the factory settings and all files and user-specific data are also deleted from the device.

- ▶ Back up all settings, files and user-specific data on the device before updating the firmware.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ New firmware has been downloaded. documentation.ifm.com
- ▶ Select the [Update] menu.
 - ▷ Menu page shows information about the current firmware version.
- ▶ Click on [Load software file] and select the new firmware file (*.bin).
- ▷ The firmware of the device is updated.
- ▷ The progress of the update process is displayed.
- ▷ After successful update: The device reboots automatically.

9.2.9 Diagnostics

The [Diagnostics] menu provides access to the device's diagnostic data.

9.2.9.1 Downloading diagnostic data

Available information:

Name	Description	Value range
Status	Status of the download	- ready to download: ready to download the log file - currently downloading....: Log file is being downloaded
Chunksize	Size of the data segment (value in bytes)	1048576

To download the log file of the device:

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The menu page [Diagnostics] is active.
- ▶ Select the [Logs] submenu.
- ▶ Click on [Download].
- ▷ The log file is downloaded to laptop/PC.

9.2.10 IODD

The [IODD] menu provides access to the management of the IO-Link Device Description (IODD) and the IODD extender function.

- ▶ Note the remarks: [IODD management \(→ 14\)](#)

Requirements:

- ✓ IoT Core Visualizer has been started.
- ▶ Select [IODD] menu.

The menu page shows a list of the IODDs stored on the device.

ifm-AL1590-000242900022 - online

Device Settings | Notification | I/O Monitor | Elements | Parameter | Processdata | Update | Diagnostics | **IODD**

Upload IODD from local PC

Auto download from IODDFinder ☒

Path to IODDs on device

List of IODD files in the current path of the device

/etc/ifm/iotcore/iodd/ifm-000215-20161025-IODD1.1.zip	
/etc/ifm/iotcore/iodd/ifm-000172-20150909-IODD1.1.zip	
/etc/ifm/iotcore/iodd/ifm-000329-20180117-IODD1.1.zip	
/etc/ifm/iotcore/iodd/ifm-000174-20210526-IODD1.1.zip	
/etc/ifm/iotcore/iodd/ifm-0001E4-20231005-IODD1.1.zip	
/etc/ifm/iotcore/iodd/ifm-0003CE-20200506-IODD1.1.zip	
/etc/ifm/iotcore/iodd/ifm-000175-20150804-IODD1.1.zip	

Available options:

- [Loading IODDs manually onto the device \(→ 72\)](#)
- [Downloading IODD from the IODDFinder portal to the device \(→ 72\)](#)

- [Deleting the IODD from the device \(→ 72\)](#)

9.2.10.1 Loading IODDs manually onto the device

The user can load both individual IODDs (*.xml) and several IODDs packed in a ZIP archive (*.zip) onto the device.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [IODD] menu is active.
- ✓ IODDs are available in the file system of the laptop/PC.
- ▶ Click on the [Click to add IODD xml file or zipped bundle] field.
 - ▷ File explorer appears.
- ▶ Select IODD or ZIP archive.
- ▶ Click on [Open] to confirm the selection.
 - ▷ The name of the selected file is displayed.
- ▶ Next to [Upload IODD from local PC], click on [Upload].
- ▷ IODDs are loaded onto the device and saved in the default directory, e.g. `/etc/ifm/iotcore/iodd`.
- ▷ List shows the IODDs saved in the standard directory.

9.2.10.2 Downloading IODD from the IODDfinder portal to the device

The IODDs of all compliant IO-Link devices are provided via the online portal ioddfinder.io-link.com.

If an unknown IO-Link device is connected to an IO-Link port of the device, the device attempts to download the appropriate IODD from the ioddfinder.io-link.com portal using the recognised Vendor ID and Device ID.

Available parameters:

Name	Description	Value range	Access
[Auto download from IO-DDFinder]	Enable / disable automatic download of the IODD	•  : Do not download IODD •  : Download IODD automatically	rw ¹

¹ Read and write

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [IODD] menu is active.
- ✓ The device can access the Internet.
- ▶ Enable the slider next to the [Auto download from IODDFinder] field.
- ▷ The function for automatically downloading IODDs is enabled.

9.2.10.3 Deleting the IODD from the device

The user can delete the IODDs saved in the standard directory individually.

Requirements:

- ✓ IoT Core Visualizer has been started.
- ✓ The [IODD] menu is active.

- ▶ Click on the  icon next to the IODD to be deleted.
 - ▷ A confirmation message appears.
- ▶ Click [OK] to delete the IODD.
- ▷ IODD in the directory is deleted.

OR:

- ▶ Click on [CANCEL] to cancel the process.
- ▷ IODD is not deleted.
- ▷ The previous view appears.

9.3 ifm IoT Core REST API

9.3.1 ifm IoT Core: General information

The device has the ifm IoT-Core. The ifm IoT-Core represents the functionality of the device. The functionality consists of a set of data, services and events. The elements of the ifm IoT-Core are arranged in a JSON object in a hierarchical tree structure. The ifm IoT-Core makes these elements available to the outside world via standard interfaces (e.g. HTTP). This allows the user and other devices to access data (parameters, process data), events and functions (services) of the device.

9.3.2 Accessing elements of the ifm IoT Core

Any element of the ifm IoT Core is accessed via its address (e.g. root/port1/pin2). The address is composed of the path leading to the element (root/port1) and the identifier of the element (pin2).

The user can access the ifm IoT Core via HTTP requests. The following methods are supported:

- [GET request \(→ 74\)](#)
- [POST request \(→ 75\)](#)

The response message to a request contains a diagnostic code. The diagnostic code provides information about the result of the processing.

- [Diagnostic codes \(→ 76\)](#)

9.3.2.1 GET request



A GET request can only be executed if the login function is disabled and HTTP communication is enabled.

Access: reading

Syntax of the request:

```
http://ip/data_point/service
```

Element	Description
ip	IP address of the IIoT interface
data_point	Data point incl. path to be accessed
service	Service to access the data point

Syntax of the response:

```
{
  "cid":id,
  "data":{
    "value":resp_data},
  "code":diag_code
}
```

Field	Element	Description
cid	id	Correlation ID for the assignment of request and response
data	resp_data	Value of the data point; depending on the data type of the data point
code	diag_code	Diagnostic code

Example: Execute GET request

Request:

```
http://192.168.10.43/deviceninfo/ordernumber/getdata
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/deviceinfo/ordernumber/getdata",
  "data": {
    "value": "AL1591",
    "timestamp": 1742396320530
  }
}
```

9.3.2.2 POST request

Access: reading, writing

Syntax of the request:

```
{
  "code": "code_id",
  "cid": id,
  "adr": "data_point/service",
  "data": {req_data},
  "auth": {"user": "usr_id", "passwd": "password"}
}
```

Field	Parameter	Description
code	code_id	Service class - request: Enquiry - transaction: transaction - event: Event
cid	id	Correlation ID for the assignment in pairs of request and response; Numeric identifier to be freely assigned by the user
adr	data_point	Data point incl. path; which is to be accessed
	service	Service to access the data point
data ¹	req_data	Data sent to the ifm IoT Core (e.g. new values); syntax depending on the service
auth ²	user	User name (default: admin); Value as BASE64-encoded STRING (Default: YWRtaW4=)
	passwd	Password: Value encoded as BASE64 STRING

¹ optional; only required for services that send data to the ifm IoT Core (e.g. setdata)

² optional; only required if security mode is activated

Syntax of the response:

```
{
  "cid": id,
  "code": diag_code,
  "data": {resp_data},
  "adr": "data_point/service"
}
```

Field	Parameter	Description
cid	id	Correlation ID for the assignment in pairs of request and response; Numeric identifier to be freely assigned by the user
code	diag_code	Diagnostic code
data	resp_data	Values returned by the ifm IoT Core; syntax depending on the service
adr	data_point	Data point incl. path; which is to be accessed
	service	Service to access the data point

A device-specific error code is issued in the event of incorrect acyclical access to a connected IO-Link device.

Syntax of the response in the event of incorrect access:

```
{
  "cid":id,
  "error":err_code,
  "code":diag_code,
  "adr":"data_point/service"
}
```

Field	Parameter	Description
cid	id	Correlation ID for the assignment in pairs of request and response; Numeric identifier to be freely assigned by the user
error	err_code	Device-specific error code (→ IO Device Description (IODD) of the IO-Link device)
code	diag_code	Diagnostic code
adr	data_point	Data point incl. path; which is to be accessed
	service	Service to access the data point

Example: Execute POST request

Request:

```
{
  "code":"request",
  "cid":1,
  "adr":"/deviceinfo/ordernumber/getdata"
}
```

Response:

```
{
  "code":200,
  "cid":1,
  "adr":"/deviceinfo/ordernumber/getdata",
  "data": {
    "value":"AL1591",
    "timestamp":1742396320530
  }
}
```

9.3.2.3 Diagnostic codes

Code	Text	Description
200	OK	Request successfully processed
230	OK but needs reboot	Request successfully processed; Device needs to reboot

Code	Text	Description
231	OK but block request not finished	Request successfully processed; but block request not finished
232	Data has been accepted, but internally modified	New values have been accepted but adjusted by the device (master cycle time)
233	IP settings (of IoT-Port) have been updated. Application needs to reload device. Wait at least 1 second before reloading the device.	The IP settings have been successfully changed, IO-Link master will be reloaded; wait for at least 1 second
400	Bad request	Invalid request
401	Unauthorized	Unauthorized request
403	Forbidden	Forbidden request
500	Internal server error	Internal fault. detailed information in the "error" field
503	Service unavailable	Service not available (e.g. IO-Link port in wrong operating mode; no IO-Link device on IO-Link port)
507	Insufficient Storage	The internal memory for notifications is full (max. 16 kB)
530	The requested data is invalid	Invalid process data
531	IO-Link Error	Error in IO-Link master / IO-Link device; detailed information in the "error" field
532	PLC connected Error	Error when writing data because IO-Link master is still connected to fieldbus PLC

9.3.3 General functions

The device has the type device. The following services can be applied to the root element of the IoT Core:

Service	Description
../gettree	Provide the complete tree or subtree of the device description (JSON)
../getidentity	Read identification information of the unit
../getdatamulti	Reading multiple elements sequentially
../getsubscriberlist	Provide a list of all active notification subscriptions
../querytree	Search device description for specific elements
../setdatamulti	Write multiple elements sequentially

The following services can be applied to elements of the type data depending on the configured access rights:

Service	Description
../getdata	Read the value of the element
../setdata	Write the value of the element

9.3.3.1 Example: Output a subtree

Task: All direct sub-elements of the devicecontrol structure node are to be output.

Solution: Use the service gettree to output the required subtree (root node: /devicecontrol, sublevels to be shown: 1)

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/gettree",
  "data": {
    "adr": "/devicecontrol",
    "level": 1
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/gettree",
  "data": {
    "type": "structure",
    "identifier": "devicecontrol",
    "adr": "/devicecontrol",
    "profiles": [
      "devicereset"
    ],
    "subs": [
      {
        "type": "service",
        "identifier": "reboot",
        "adr": "/devicecontrol/reboot"
      },
      {
        "type": "service",
        "identifier": "factoryreset",
        "adr": "/devicecontrol/factoryreset"
      },
      {
        "type": "service",
        "identifier": "signal",
        "adr": "/devicecontrol/signal"
      },
      {
        "type": "service",
        "identifier": "gotobootloader",
        "adr": "/devicecontrol/gotobootloader"
      }
    ]
  }
}
```

9.3.3.2 Example: Changing a parameter value

Task: The operating mode of pin 4 (C/Q) of port 1 should be set to digital input.

Solution: Use the setdata service to set the data point /iolinkmaster/port[1]/mode to the value 1 (Digital Input).

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/iolinkmaster/port[1]/mode/setdata",
  "data": {
    "newvalue": 1
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/iolinkmaster/port[1]/mode/setdata"
}
```

9.3.3.3 Example: Reading multiple elements sequentially

Task: The following current values of the device are to be read consecutively: Serial number, operating status

Solution: Read the values of the following data points with the getdatamulti service:

- Serial number: /deviceinfo/serialnumber
- Operating status: /devicestatus/operatingstate

Request:

```
{
  "cid": 1,
  "code": "request",
  "adr": "/getdatamulti",
  "data": { "datatosend": [
    "/deviceinfo/serialnumber",
    "/devicestatus/operatingstate"
  ] }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/getdatamulti",
  "data": {
    "/deviceinfo/serialnumber": {
      "code": 200,
      "data": "000250500047",
      "timestamp": 1742521856941
    },
    "/devicestatus/operatingstate": {
      "code": 200,
      "data": 2,
      "timestamp": 1742521856946
    }
  }
}
```

9.3.3.4 Example: Writing multiple elements sequentially

Task: The operating mode of pin 4 (C/Q) of port 3 should be set to the value DI (digital input) and the operating mode of pin 4 (C/Q) of port 4 should be set to DO (digital output).

Solution: Use the setdatamulti service to write the desired values to the data points /iolinkmaster/port[3]/mode and /iolinkmaster/port[4]/mode .

Request ("data" field):

```
{
  "code": "request",
  "cid": 1,
  "adr": "/setdatamulti",
  "data": {
    "datatosend": {
      "/iolinkmaster/port[3]/mode": 1,
      "/iolinkmaster/port[4]/mode": 2
    }
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/setdatamulti",
  "data": {
    "/iolinkmaster/port[3]/mode": {
      "code": 200
    },
    "/iolinkmaster/port[4]/mode": {
      "code": 200
    }
  }
}
```

9.3.3.5 Example: Browsing device description

Task: List all elements with the profile timer.

Solution: Use the querytree service to search the device description with the timer (profile) parameter.

Request:

```
{
  "cid": 1,
  "code": "request",
  "adr": "querytree",
  "data": {
    "profile": "timer"
  }
}
```

Response:

```
{
  "cid":4711,
  "data":{
    "adrList":[
      "/timer[1]",
      "/timer[2]"
    ]
  },
  "code": 200
}
```

9.3.4 Device settings

9.3.4.1 Setting the fieldbus type

Substructure: /devicesettings/fieldbustype

Available data points:

Parameter	Description	Value range	Access
../status	Active fieldbus type	0: Profinet - PROFINET 2: EtherNet/IP – EtherNet/IP	ro ¹

¹ Read only

Applicable services:

Name	Description
../profinet/enable	Enable PROFINET fieldbus type
../ethernetip/enable	Enable EtherNet/IP fieldbus type

Example: Enable PROFINET

Task: PROFINET should be set as the active fieldbus type of the device.

Solution: Execute the enable service on the Profinet node.

Request:

```
{
  "code":"request",
  "cid": 1,
  "adr": "/devicesettings/fieldbustype/profinet/enable"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/devicesettings/fieldbustype/profinet/enable"
}
```

9.3.4.2 Configuring the security mode



After reactivating the authentication function, the password is reset to the default value password.

Substructure: /security

Available data points:

Name	Description	Value range	Access
../securitymode	Status of the security mode NOTE! Security mode is always active and cannot be disabled.	0: Disabled – security mode disabled 1: Enabled – security enabled	ro ¹
../authentication_enabled	Enable / disable authentication function (login)	- false: disable login function - true: Enable login function	rw ²
../password	Password for the authentication function NOTE! Value as Base64-encoded string	e.g. password (default)	wo ³

¹ Read only

² Read and write

³ write only

Disabling the authentication

Task: The authentication function (login) of the device should be disabled.

Solution: Use the setdata service to write the value false to the data point authentication_enabled.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/security/authentication_enabled/setdata",
  "data": {
    "newvalue": true
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/security/authentication_enabled/setdata"
}
```

The user can access the ifm IoT Core (REST API, IoT Core Visualiser) without providing authentication data (user name, password).

Changing the password

Task: The factory-set password is to be changed. New password: Passw0rd

Solution: Use the setdata service to write the new password as a Base64-encoded hex string to the password data point. New password (Base64-encoded): UGFzc3cwcmQ=



For the Base64 coding of STRING values: www.base64encode.org

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/security/password/setdata",
  "data": {
    "newvalue": "UGFzc3cwcmQ="
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/security/password/setdata"
}
```

9.3.4.3 Setting the system time

Substructure: /devicesettings/systemtime

Available data points:

Name	Description	Value range	Access
../datetime	Current system time; Format UTC (milliseconds from 01.01.1970)	0 ms ... 18 446 744 073 709 551 615 ms	rw ¹
../timezone	UTC time zone List of IANA time zones: utctime.info/time-zone	e.g. "Europe/Berlin"	rw ¹

¹ Read and write

Example: Setting the system time

Task: The system time should be set to 13.02.2025, 19:26:22.

Solution: The desired system time corresponds to the Unix timestamp 1739471182. Use the setdata service to write the new value 1739471182 to the data point datetime.

Request:

```
{
  "cid": 1,
  "code": "request",
  "adr": "/devicesettings/timezone/datetime/setdata",
  "data": {
    "newvalue": 1739471182
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/devicesettings/timezone/datetime/setdata"
}
```

Example: Setting the time zone

Task: The time zone should be set to Berlin (Germany).

Solution: Use the setdata service to write the value Europe/Berlin to the data point timezone (reference: utctime.info/timezone).

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/devicesettings/systemtime/timezone/setdata",
  "data": {
    "newvalue": "Europe/Berlin"
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/devicesettings/systemtime/timezone/setdata"
}
```

9.3.4.4 Configuring the timers

The device has 2 timers. The timers can be used for time-triggered transmission of notifications.

The timers are delivered enabled from the factory.

Substructure: timer[1] | timer[2]

Available data points:

Name	Description	Value range	Access
../counter	Counter reading	0...4294967295	ro ¹
../interval	Counting interval (value in ms)	1000: 1000 ms ... 86400000: 86400000 ms	rw ²
../interval/unit	Unit of the counting interval	ms: milliseconds	ro ¹
../status	Counter status	0: stopped - counter stopped 1: running - counter running	ro ¹

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start the counter
../status/stop	Stop counter

Example: Set the counting interval

The counting interval of timer 2 is to be set to 2.5 s.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
  "data": {
    "newvalue": 2500
  }
}
```

Response:

```
{
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
  "code": 200
}
```

Example: Start the counter

The counter of timer 2 is configured and is to be started.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/status/start"
}
```

Response:

```
{
  "cid": 1,
  "adr": "/timer[2]/status/start",
  "code": 200
}
```

Example: Stop the counter

The counter of timer 2 is running and is to be stopped.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/status/stop"
}
```

Response:

```
{
  "cid": 1,
  "adr": "/timer[2]/status/stop",
  "code": 200
}
```

9.3.4.5 Configuring the status LED

Substructure: /devicestatus/statusled

Available data points:

Name	Description	Value range	Access
../status/type	Type of status LED	• rgb: LED with RGB colour space	ro ¹
.../status/rgb_value	RGB colour value of the status LED (value as hexadecimal number)	• 000000: Off • FF0000: Red • 00FF00: Green • FFFF00: Yellow	rw ²
../status/name	Name of the status LED	• Status LED	ro ¹
../status/assign	Instance for colour assignment	• 1: User - user	ro ¹
../status/frequency	Flashing frequency of the status LED (value in mHz)	• 0: Off • 1: 1 mHz ... • 2 147 483 647: 2 147 483 647 mHz	rw ²

¹ Read only

² Read and write

9.3.5 IIoT interface

9.3.5.1 Configuring the IIoT interface

Substructure: /network/br0

Available data points:

Name	Description	Value range	Access
../macaddress	MAC address of the interface	E.g. 00:02:01:10:4F:BD	ro ¹
../hostname	Host name of the device Default value: ifm-artno-serial - ifm: supplier name - artno: Article number of the device - serial: Serial number of the device	e.g. ifm-AL1591-000242900022	rw ²
../ipv4/	Substructure for IPv4 address		
../ipv4/mode	Desired IP configuration mode	1: Static - manual IP configuration 2: DHCP - receive IP configuration from DHCP server (default) 4: link-local - IP configuration via link-local protocol	rw ²
../ipv4/gateway	IP address of the network gateway	E.g. 192.168.10.1	rw ²
../ipv4/dnsservers	Set DNS server	E.g. 192.168.10.100	ro ¹
../ipv4/staticdnsservers	Static DNS server List with comma-separated IP addresses - Input scheme: ["IP1","IP2",...] - Empty list: []	e.g. ["91.239.100.100","89.233.43.71"]	rw ²
../ipv4/address0	Substructure for address 0		
../ipv4/address0/mode	Set IP configuration mode	1: Static - manual IP configuration 2: DHCP - receive IP configuration from DHCP server 4: link-local - IP configuration via link-local protocol	ro ¹
../ipv4/address0/address	IP address of the interface	e.g. 192.168.10.43	rw ³
../ipv4/address0/subnetmask	Subnet mask of the network segment	e.g. 255.255.255.0	rw ³

¹ Read only

² Read and write

³ Read and write; parameter can only be changed by using the service setblock and if manual addressing mode is enabled

Applicable services:

Name	Description
../ipv4/address0/setblock	Write data points address and subnetmask of the substructure address0 contiguously

Example: Setting an IP address manually

Task: The IP configuration of the IIoT interface should be changed from DHCP to manual addressing (IP address: 192.168.10.43, subnet mask: 255.255.255.0) can be changed.

Solution: Use the setdata service to write the desired addressing mode to the /network/br0/ipv4/mode data point. Then use the setblock service to write the desired IP configuration to the /network/br0/ipv4/address0 structure.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/network/br0/ipv4/mode/setdata",
  "data": {
    "newvalue": 1
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/network/br0/ipv4/mode/setdata"
}
```

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/network/br0/ipv4/address0/setblock",
  "data": {
    "datatoset": {
      "address": "192.168.10.43",
      "subnetmask": "255.255.255.0"
    }
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 2,
  "adr": "/network/br0/ipv4/address0/setblock",
}
```

Example: Adding static DNS servers

Task: The IP addresses 91.239.100.100 and 89.233.43.71 are to be used as static DNS servers for name resolution.

Solution: Use the setdata service to write the IP addresses to the staticdnsservers data point.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/network/br0/ipv4/staticdnsservers/setdata",
  "data": {
    "newvalue": ["91.239.100.100", "89.233.43.71"]
  }
}
```

Response:

```
{  
  "code":200,  
  "cid":1,  
  "adr":"/network/br0/ipv4/staticdnsservers/setdata",  
}
```

9.3.6 Connections

9.3.6.1 Using HTTPS connection

Substructure: /connections/httpsConnection

Available data points:

Name	Description	Value range	Access
../status	Status of the connection	0: stopped -HTTPS communication stopped 1: running - HTTPS communication started	ro ¹
../type	Type of connection	https (default)	ro ¹
../httpssetup	Substructure for settings		
../httpssetup/certificate_info	Information about the HTTPS certificate stored on the device		ro ¹
../httpssetup/certificate	HTTPS certificate (value as HEX-STRING)		rw ²
../httpssetup/port	Port for HTTPS connection	443 (default)	ro ¹

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start HTTPS communication
../status/stop	Stop HTTPS communication

9.3.6.2 Using HTTP connection

Substructure: /connections/httpConnection

Available data points:

Name	Description	Value range	Access
../status	Status of the connection	0: stopped - connection stopped 1: running - connection active (default)	ro ¹
../type	Type of connection	http (default)	ro ¹
../httpsetup	Substructure for settings		
../httpsetup/allowed_services	Services of the ifm IoT Core permitted via the HTTP connection	0: All services allowed - all services allowed 1: Only getidentity service allowed - only service getidentity allowed (default)	rw ²
../httpsetup/port	Port used by the connection	80 (default)	ro ¹

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start HTTP communication
../status/stop	Stop HTTP communication

Example: Enabling the HTTP connection

Task: HTTP connections to the device should be permitted.

Solution: Activating HTTP communication with the start service.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/connections/httpConnection/status/start"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/connections/httpConnection/status/start"
}
```

Once HTTP communication has been enabled, the device can be accessed via an HTTP connection, port 80.

9.3.6.3 Using MQTT client

Substructure: /connections/mqttclient

Available data points:

Name	Description	Value range	Access
../type	Type of connection	• mqtt: MQTT	ro ¹
../status	status of the application	- stopped: application has been stopped - running: application has been started	rw ²
../mqttsetup	Substructure for MQTT client settings		
../mqttsetup/version	Version of the MQTT client	• V311: Version 3.11	ro ¹
../mqttsetup/qos	Quality of service level	0: QoS0 – QoS level 0 - PUBLISH (without confirmation) (default) 1: QoS1 – QoS level 1 - PUBLISH > PUBREC (simple confirmation) 2: QoS2 - QoS Level 2 - PUBLISH > PUBREC > PUBREL > PUBCOMP (double confirmation)	rw ²
../mqttsetup/keepalive	Time period without data transmission after which a PINGREQ is sent to the MQTT broker (value in s) NOTE! Please note: MQTT broker responds to PINGREQ with PINGRESP	0: deactivated ... 15: 15 s (default) ... 60: 60 s	rw ²
../mqttsetup/willtopic	Fallback topic if connection is interrupted	E.g. failsafe	rw ²
../mqttsetup/willmessage	Fallback message if connection is interrupted	E.g. not valid	rw ²
../mqttsetup/clientid	Unique identifier of the MQTT client (max. 23 characters)	E.g. 61af17e497d6499ab95cc253f5b0df-de	rw ²

Name	Description	Value range	Access
../mqttsetup/securesetup	Substructure for security settings for MQTTS connections		
../mqttsetup/securesetup/secure	Security settings optional or mandatory	2: optional - optional	ro ¹
../mqttsetup/securesetup/certificate	Certificate (value as STRING)		
../mqttsetup/securesetup/key	Personal key (value as STRING)		
../mqttsetup/securesetup/validate-servercertificate	Validation option for MQTT broker certificates	0: none – keine Validierung 1: required – verpflichtende Validierung 2: required_no_ca: verpflichtende Validierung, nicht für Zertifikate von Zertifizierungsstellen (Certificate Authorities) (Default)	rw ²
../mqttsetup/securesetup/trusted-servercertificates	List of certificates from trusted MQTT brokers (multiple certificates separated by commas)		rw ²

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start MQTT client
../status/stop	Stop MQTT client

Example: Publishing voltage value to MQTT broker

Task: The temperature of the device is to be published to an MQTT broker every 5 seconds (IP address: 192.168.82.100, port: 1883, topic: deviceCurrent).

Solution:

► Setting the interval of timer 1

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[1]/interval/setdata",
  "data": {
    "newvalue": 5000
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/timer[1]/interval/setdata"
}
```

► Publishing voltage value to MQTT broker

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/timer[1]/counter/datachanged/subscribe",
  "data": {
    "callback": "mqtt://192.168.82.100:1883/deviceTemp",
    "datatosend": ["processdatamaster/current"]
  }
}
```

Response:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/timer[1]/counter/datachanged/subscribe"
}
```

9.3.6.4 Using MQTT command channel

Several connections can be established to an MQTT broker.

Substructure: /connections/mqttxcommandchannel

Available data points:

Name	Description	Value range	Access
../type	Type of connection	mqtt: MQTT connection	ro ¹
../status	Status of the MQTT command channel	- init: Initialisation state - stopped: MQTT command channel stopped - running: MQTT command channel launched - error: error state	ro ¹
../mqttxcmdchannel			
../mqttxcmdchannel/status	Status of the command channel	- init: Initialisation state - stopped: MQTT command channel stopped - running: MQTT command channel launched - error: error state	ro ¹
../mqttxcmdchannel/mqttxcmdchannel-setup	Substructure for MQTT command channel settings		
../mqttxcmdchannel/mqttxcmdchannel-setup/brokerip	IP address of the MQTT broker	e.g. 192.168.100.1	rw ²
../mqttxcmdchannel/mqttxcmdchannel-setup/brokerport	Port number of the MQTT broker	e.g. 1883 (default)	rw ²
../mqttxcmdchannel/mqttxcmdchannel-setup/cmdtopic	Command topic that MQTT clients monitor for requests from the ifm IoT Core	e.g. iotcore-request (default)	rw ²
../mqttxcmdchannel/mqttxcmdchannel-setup/defaultreplytopic	Default reply topic if no reply topic is specified in the request	e.g. iotcore-reply (default)	rw ²
../mqttxcmdchannel/mqttxcmdchannel-setup/user	User name for authentication on the MQTT broker	E.g. user1	rw ²
../mqttxcmdchannel/mqttxcmdchannel-setup/password	Password for authentication on the MQTT broker	E.g. passw0rd	wo ³
../mqttxcmdchannel/mqttxcmdchannel-setup/version	MQTT version	V311: Version 3.11	ro ¹

Name	Description	Value range	Access
../mqttcmdchannel/mqttcmdchannel-setup/qos	Quality of service level for responses to the MQTT broker	0: QoS0 – QoS level 0 - PUBLISH (without conformation) (default) 1: QoS1 – QoS level 1 - PUBLISH > PUBREC (simple conformation) 2: QoS2 - QoS Level 2 - PUBLISH > PUBREC > PUBREL > PUBCOMP (double conformation)	ro ¹
../mqttcmdchannel/mqttcmdchannel-setup/keepalive	Time period without data transmission after which a PINGREQ is sent to the MQTT broker (value in s) NOTE! PLEASE NOTE! Please note: MQTT broker responds to PINGREQ with PINGRESP	0: 0 s ... 15: 15 s (default) ... 60 s	rw ²
../mqttcmdchannel/mqttcmdchannel-setup/willtopic	Fallback topic if connection is interrupted	E.g. failsafe	rw ²
../mqttcmdchannel/mqttcmdchannel-setup/willmessage	Fallback message if connection is interrupted	e.g not valid	rw ²
../mqttcmdchannel/mqttcmdchannel-setup/clientid	Unique identifier of the MQTT client (max. 23 characters)	E.g. d7d9212215904c93a52db151ae129	rw ²
../mqttcmdchannel/mqttcmdchannel-setup/securesetup	Substructure for MQTTS connection settings		
../mqttcmdchannel/mqttcmdchannel-setup/securesetup/secure	Security settings optional or mandatory	0: disabled – disabled (default) 1: required - mandatory	rw ²
../mqttcmdchannel/mqttcmdchannel-setup/securesetup/certificate	Certificate of the MQTT client (value as STRING)		rw ²
../mqttcmdchannel/mqttcmdchannel-setup/securesetup/key	Personal key of the MQTT client (value as STRING)		rw ²
../mqttcmdchannel/mqttcmdchannel-setup/securesetup/validateservercertificate	Validate certificates of the MQTT broker	0: none - no validation 1: required - mandatory 2: required_no_ca: mandatory, no certificates from certificate authorities (default)	rw ²
../mqttcmdchannel/mqttcmdchannel-setup/securesetup/trustedservercertificates	Array with trusted certificates		rw ²

¹ Read only² Read and write³ write only

Applicable services:

Name	Description
../status/stop	Stop the MQTT command channel
../status/start	Start the MQTT command channel

Configuring the MQTT command channel

The configuration and activation of the MQTT command channel comprises the following steps:

1. set quality of service level
2. Enable the MQTT connection
3. Set the IP address of the MQTT broker.
4. Set the port number of the MQTT broker.
5. Set command topic

6. Set standard response topic.



Make sure that the MQTT broker can be reached and that the selected port of the MQTT broker is released for data transfer.

The wildcards “+” and “#” are not supported in topic names.

Example: Configure and activate MQTT command channel

Task: Configuring and activating the MQTT command channel (IP address of the MQTT broker): 192.168.82.100, port: 1883, command topic: reqTopic, standard response topic: respTopic, Quality of Service: Level 2)

Solution:

- Check whether MQTT broker can be reached and the port has been released.
- Set QoS level.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/connections/mqttConnection/MQTTSetup/cmdChannel/QoS/setdata"
  "data": {"newvalue": 2}
}
```

}

- Activate MQTT connection

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "connections/mqttConnection/MQTTSetup/cmdChannel/Status/start"
}
```

- Set the IP address of the MQTT broker.

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "connections/mqttConnection/mqttCmdChannel/mqttCmdChannelSetup/
brokerIP/setdata",
  "data": {
    "newvalue": "192.168.82.100"
  }
}
```

- Set the port number of the MQTT broker.

Request:

```
{
  "code": "request",
  "cid": 3,
  "adr": "connections/mqttConnection/mqttCmdChannel/mqttCmdChannelSetup/
brokerPort/setdata",
  "data": {
    "newvalue": "1883"
  }
}
```

► Set the command topic.

Request:

```
{
  "code": "request",
  "cid": 4,
  "adr": "connections/mqttConnection/mqttCmdChannel/mqttCmdChannelSetup/
cmdTopic/setdata",
  "data": {
    "newvalue": "reqTopic"
  }
}
```

► Set standard response topic.

Request:

```
{
  "code": "request",
  "cid": 5,
  "adr": "/connections/mqttConnection/mqttCmdChannel/mqttCmdChannelSetup/
defaultReplyTopic/setdata",
  "data": {
    "newvalue": "respTopic"
  }
}
```

9.3.6.5 Using the OPC UA server

Substructure: /connections/opcua_server

Available data points:

Name	Description	Value range	Access
../status	Status of the connection	0: stopped: stopped 1: running: sample	ro ¹
../settings	Configuration of the OPC UA server A configuration consists of the port (key "port") and an array of supported security profiles (key "security_profiles"). Each security profile has a name (key "name") and a status (key "enabled")	Example: Configuring the OPC UA server (→ 97)	rw ²

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start OPC UA server

Name	Description
../status/stop	Stop OPC UA server
../upload_certificate	Load certificate or private key onto the device

Example: Configuring the OPC UA server

Task: The configuration of the running OPC UA server should be changed as follows: New port: 4840; Disable security policy "Aes128Sha256RsaOaep".

Solution: As the OPC UA server can only be configured in a disabled state, the solution consists of the following 3 steps.

- Stop OPC UA server. Execute the stop service on the status data point.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/connections/opcua_server/status/stop"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/connections/opcua_server/status/stop"
}
```

- Change configuration: Use the setdata service to write the JSON object with the changed key/value pairs to the settings data point.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/connections/opcua_server/settings/setdata",
  "data": {
    "newvalue": {
      "port": 4843,
      "security_profiles": [
        {
          "name": "Aes128Sha256RsaOaep",
          "enabled": false
        }
      ]
    }
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/connections/opcua_server/settings/setdata"
}
```

- Start the OPC UA server: Execute the start service on the status data point.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/connections/opcua_server/status/start"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/connections/opcua_server/status/start"
}
```

9.3.7 Fieldbus interface

The /fieldbussetup substructure contains different elements depending on the fieldbus type set.

- PROFINET:
 - [PROFINET: Reading the interface configuration \(→ 99\)](#)
- EtherNet/IP:
 - [EtherNet/IP Reading the interface configuration \(→ 99\)](#)
 - [EtherNet/IP Setting the configuration mode \(→ 100\)](#)

9.3.7.1 PROFINET: Reading the interface configuration

Substructure: /fieldbussetup

Available data points:

Name	Description	Value range	Access
../fieldbusfirmware	Firmware of the fieldbus	E.g. 1.0.3.6	ro ¹
../connectionstatus	Status of the fieldbus connection	• 0: Disconnected - not connected • 1: Connected - connected	ro ¹
../fieldbustype	Active fieldbus type	• 0: PROFINET	ro ¹
../network/ipaddress	IP address of the interface	e.g. 192.200.0.100 • 0.0.0.0 (default)	ro ¹
../network/subnetmask	Subnet mask of the network segment	e.g. 255.255.192.0 • 0.0.0.0 (default)	ro ¹
../network/ipdefaultgateway	IP address of the network gateway	e.g. 192.200.63.1 • 0.0.0.0 (default)	ro ¹
../network/macaddress	MAC address of the fieldbus interface	e.g. 00:02:01:0E:10:7C	ro ¹
../network/hostname	Station name	E.g. alxx9x	ro ¹

¹ Read only

9.3.7.2 EtherNet/IP Reading the interface configuration

Substructure: /fieldbussetup

Available data points:

Name	Description	Value range	Access
../fieldbusfirmware	Firmware of the fieldbus	E.g. 1.0.3.6	ro ¹
../connectionstatus	Status of the fieldbus connection	• 0: Disconnected - not connected • 1: Connected - connected	ro ¹
../fieldbustype	Active fieldbus type	• 2: EtherNet/IP	ro ¹
../dhcp	Active method for IP address assignment	• 0: Static - manual address assignment • 1: DHCP - DHCP client • 2: BOOTP - address assignment via BOOTP (Bootstrap Protocol)	ro ¹
../network/ipaddress	IP address of the interface	e.g. 192.200.0.100 • 192.168.1.250 (default)	ro ¹
../network/subnetmask	Subnet mask of the network segment	e.g. 255.255.192.0 • 255.255.255.0 (default)	ro ¹
../network/ipdefaultgateway	IP address of the network gateway	e.g. 192.200.63.1 • 0.0.0.0 (default)	ro ¹
../network/macaddress	MAC address of the fieldbus interface	e.g. 00:02:01:0E:10:7C	
../network/hostname	Hostname	E.g. alxx9x	ro ¹

¹ Read only

9.3.7.3 EtherNet/IP Setting the configuration mode

Substructure: /fieldbussetup/configuration

Available data points:

Name	Description	Value range	Access
../independentmode	Configuration mode	0: Off - independent mode off (default) 1: On - independent mode on	rw ^{1 2}
../processdataconfiguration	Length of the process data	0: 2 Bytes Input 2 Bytes Output - 2 byte inputs, 2 byte outputs 1: 4 Bytes Input 4 Bytes Output - 4 byte inputs, 4 byte outputs 2: 8 Bytes Input 8 Bytes Output - 8 bytes inputs, 8 bytes outputs 3: 16 Bytes Input 16 Bytes Output - 16 bytes inputs, 16 bytes outputs 4: 32 Bytes Input 32 Bytes Output - 32 bytes inputs, 32 bytes outputs (default)	rw ^{1 2}
../swap	Swap of the bytes in process data	0: Disabled - disabled 1: Enabled – enabled (default)	rw ^{1 2}
../port[x]/failsafedigitalout	Fallback value for digital output - pin 4 of port x (x:1...8), operating mode DO	0: Reset - LOW level (default) 1: Old - last active process value 2: Set - HIGH level	rw ^{1 2}
../port[x]/failsafeiolink	Fallback value for IO-Link output data - pin 4 of port x (x: 1...8), operating mode IO-Link	0: Off - fail-safe disabled (default) 1: Reset - fail-safe enabled, reset value 2: Old - fail-safe enabled, last active process value 3: Pattern: Fail-safe enabled, byte sequence	rw ^{1 2}
../port[x]/failsafeiqout	Fallback value for digital output - pin 2 of port x (x:X1...X4), operating mode DO	0: Reset - LOW level (default) 1: Old - last active process value 2: Set - HIGH level	rw ¹

¹ Read and write

² Read and write; parameter can only be changed if no connection to fieldbus controller is established

9.3.8 Software

9.3.8.1 Reading software information

Substructure: /software

Applicable services:

Name	Description
../swinfo	Read information about the software components of the device

Example: Reading software information

Task: The identification and status information about the software components should be read.

Solution: Execute the swinfo service on the software structure

Request:

```
{
  "code": 10,
  "cid": 1,
  "adr": "/software/swinfo"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/software/swinfo",
  "data": [
    {
      "ProductionSystem": {
        "Name": "Production system",
        "Version": "1.1.0.1",
        "Active": "true"
      }
    },
    {
      "RecoverySystem": {
        "Name": "Recovery system",
        "Version": "1.0.0.5",
        "Active": "false"
      }
    }
  ]
}
```

9.3.8.2 Firmware update



ifm electronic recommends using the IoT Core Visualizer ([→ IoT Core Visualizer 44](#)) to update the firmware.

Substructure: /software/firmware

Available data points:

Name	Description	Value range	Access
../version	Firmware version	1.0.3.6	ro ¹

Name	Description	Value range	Access
../type	Software type	• firmware	ro ¹
../installstatus	Display of the installation status	• progress: 0...100 • value: %	ro ¹
../container	Structure for firmware (BLOB)	-	wo ²
../container/maxsize	Container size (in bytes)	e.g. 4194304	ro ¹
../container/chunksize	Size of a data segment (in bytes)	e.g. 4096	ro ¹
../container/size	Size of firmware file in container (in bytes)	e.g. 634523	ro ¹

¹ Read only

² write only

Applicable services:

Name	Description
../install	Install firmware
../abortinstall	Cancelling the installation process
../container/stream_set	Transfer an individual data segment
../container/start_stream_set	Start sequential transmission of several data segments

9.3.9 Timer

9.3.9.1 Configuring the timers

The device has 2 timers. The timers can be used for time-triggered transmission of notifications.

The timers are delivered enabled from the factory.

Substructure: timer[1] | timer[2]

Available data points:

Name	Description	Value range	Access
../counter	Counter reading	0...4294967295	ro ¹
../interval	Counting interval (value in ms)	1000: 1000 ms ... 86400000: 86400000 ms	rw ²
../interval/unit	Unit of the counting interval	ms: milliseconds	ro ¹
../status	Counter status	0: stopped - counter stopped 1: running - counter running	ro ¹

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start the counter
../status/stop	Stop counter

Example: Set the counting interval

The counting interval of timer 2 is to be set to 2.5 s.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
  "data": {
    "newvalue": 2500
  }
}
```

Response:

```
{
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
  "code": 200
}
```

Example: Start the counter

The counter of timer 2 is configured and is to be started.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/status/start"
}
```

Response:

```
{
  "cid": 1,
  "adr": "/timer[2]/status/start",
  "code": 200
}
```

Example: Stop the counter

The counter of timer 2 is running and is to be stopped.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/status/stop"
}
```

Response:

```
{
  "cid": 1,
  "adr": "/timer[2]/status/stop",
  "code": 200
}
```

9.3.10 Device status

9.3.10.1 Reading the operating status

Substructure: /devicestatus

Available data points:

Name	Description	Value range	Access
../operatingstate	Operating status of the unit	0: PREINIT - pre-initialisation 1: INIT - initialisation 2: OPERATING - normal operation 3: WARNING - warning 4: FATAL_ERROR - fatal error 5: SYSTEM_STOP - system stopped 6: UPDATE - firmware update 7: SHUTDOWN - device is shut down	ro ¹

¹ Read only

9.3.10.2 Reading operating hours

Substructure: /devicestatus

Available data points:

Name	Description	Value range	Access
../operatinghours	Operating hours (value in h)	e.g. 12: 12 hours	ro ¹

¹ Read only

9.3.10.3 Configuring the status LED

Substructure: /devicestatus/statusled

Available data points:

Name	Description	Value range	Access
../status/type	Type of status LED	rgb: LED with RGB colour space	ro ¹
.../status/rgb_value	RGB colour value of the status LED (value as hexadecimal number)	000000: Off - FF0000: Red 00FF00: Green - FFFF00: Yellow	rw ²
../status/name	Name of the status LED	Status LED	ro ¹
../status/assign	Instance for colour assignment	1: User - user	ro ¹
../status/frequency	Flashing frequency of the status LED (value in mHz)	0: Off 1: 1 mHz ... 2 147 483 647: 2 147 483 647 mHz	rw ²

¹ Read only

² Read and write

9.3.10.4 Reading temperatures

Substructure: /devicestatus/temperature

Available data points:

Name	Description	Value range	Access
../cpu	Processor temperature (value in °C)	E.g. 35	ro ¹

Name	Description	Value range	Access
../cpu/unit	Temperature unit	• °C: Degree Celsius	ro ¹
../board	Temperature of the circuit board (value in °C)	E.g. 30.5	ro ¹
../board/unit	Temperature unit	• °C: Degree Celsius	ro ¹
../emmc	Temperature of the EMMC memory (value in °C)	E.g. 30	ro ¹
../emmc/unit	Temperature unit	• °C: Degree Celsius	ro ¹
../power5V	Temperature at power 5V contact (value in °C)	E.g. 29.5	ro ¹
../power5V/unit	Temperature unit	• °C: Degree Celsius	ro ¹

¹ Read only

9.3.11 IO-Link ports

9.3.11.1 Setting operating mode pin 4 (C/Q)

Substructure: /iolinkmaster/port[x] (x: 1...8)

Available data points:

Name	Description	Value range	Access
.../mode	Operating mode of pin 4 (US) of the port	0: Disabled - port deactivated 1: DI - digital input 2: DO - digital output 3: IO-Link - IO-Link data	rw ¹
../mastercycletime_actual	Current cycle time of the data transfer between port and IO-Link device (value in µs)	1: 1 µs ... 132800: 132800 µs	ro ²
../mastercycletime_actual/unit	Unit of the cycle time	µs Microseconds	ro ²
../comspeed	Current transfer rate between port and IO-Link device	0: COM1 - 4.8 kBaud 1: COM2 - 38.4 kBaud 2: COM3 - 230.4 kBaud	ro ²
../mastercycletime_preset	Cycle time of the data transfer between port and IO-Link device (value in µs); NOTE! Observe the technical data of the IO-Link device!	0: Automatically set the best possible cycle time 1: 1 µs ... 132800: 132800 µs	rw ¹
../mastercycletime_preset/unit	Unit of the cycle time	µs Microseconds	ro ²

¹ Read and write; parameter can only be changed if no connection to fieldbus controller is established

² Read only

9.3.11.2 Configuring the device validation and data storage



Note the remarks: [Parameter storage and device validation](#) (→ [12](#))



Device validation and backup / restore of parameter values are only available if the “IO-Link” operating mode of the port has been enabled.

For options [Type compatible V1.1 device with Backup + Restore] and [Type compatible V1.1 device with Restore]: If the [Vendor ID] or [Device ID] parameters are changed in online mode, the data memory is deleted and a new backup of the parameter values of the connected IO-Link device is created in the IO-Link master.

Substructure: /iolinkmaster/port[x] (x: 1...8)

Available data points:

Name	Description	Value range	Access
../validation_datastorage_mode	Mode for data storage and device validation	0: No check and clear - no validation - No data storage and data recovery 1: Type compatible V1.0 device - Validation (IO-Link V1.0) - No data storage and data recovery 2: Type compatible V1.1 device - Validation (IO-Link V1.1) - No data storage and data recovery 3: Type compatible V1.1 device with Backup + Restore - Validation (IO-Link V1.1) - Data storage and data recovery 4: Type compatible V1.1. device with Restore - Validation (IO-Link V1.1) - Data recovery - No data storage	rw ¹
../validation_vendorid	Manufacturer ID for device validation	0...65535	rw ¹
../validation_deviceid	Device ID for device validation	0...16777215	rw ¹

¹ Read and write; parameter can only be changed if no connection to fieldbus controller is established

Applicable services:

Name	
../validation_useconnecteddevice	Validate connected IO-Link device

9.3.11.3 Setting the operating mode pin 2 (I/Q)

Substructure: /iolinkmaster/port[x] (x: 1...4)

Available data points:

Name	Description	Value range	Access
../powercontrol/mode_ua	Operating mode pin 2 (I/Q) of the port	0: Off (IO-Link type a supply) - supply OFF - IO-Link port (class A) 1: On (IO-Link type b supply) - supply ON - IO-Link port (class B) 2: Digital output – digital output	rw ¹

¹ Read and write

9.3.11.4 Switching the supply voltage of the ports on / off

The user can switch off the supply voltage US of a port (pin 1) temporarily or permanently, e.g. to reset a connected IO-Link device.

Substructure: /iolinkmaster/port[x] (x: 1...8)

Applicable services:

Service	Description
../power_on	Switch on the supply voltage US of the port
../power_off	Switch off the supply voltage US of the port
../power_cycle	Switch off the supply voltage US of the port and switch it on again after a defined period of time

Example: Restarting the IO-Link device with a delay

Task: To restart the IO-Link device at port 5 of the device, the supply voltage US should be switched off and switched on again after 1.5 seconds.

Solution: Execute the service `power_cycle` and pass the desired switch-off duration as parameter `off_duration`; conversion of the switch-off duration. 1.5 s = 1500 ms.

Request:

```
{
  "cid": 1,
  "code": "request",
  "adr": "/iolinkmaster/port[5]/power_cycle",
  "data": {
    "off_duration": 1500
  }
}
```

Response:

```
{
  "cid": 1,
  "code": 200,
  "adr": "/iolinkmaster/port[5]/power_cycle"
}
```

Example: Reading an IO-Link process value (operating mode "IO-Link")

Task: Read the current temperature value of the ifm IO-Link device TN2531 at port 2.

Solution: Use the `getdata` service to read the data point `pdin` and then extract the temperature value from the read process value.

Request:

```
{
  "code": "request",
  "cid": 4711,
  "adr": "/iolinkmaster/port[2]/iolinkdevice/pdin/getdata"
}
```

Response:

```
{
  "cid": 4711,
  "data": { "value": "03C9" },
  "code": 200
}
```

The read process value is displayed in hexadecimal format. The process value is given in hexadecimal format. Besides the temperature value the return value comprises additional information (→ IO Device Description (IODD) of the sensor). The temperature value is shown in bits 2 to 15.

0x03C9 = 0b1111001001

Temperature value: 0b11110010 = 242

This means: The current temperature value is 24.2 °C.

Example: Reading an IO-Link process value (operating mode "IO-Link")

Task: The buzzer of the DV2500 at port 2 is to be switched on. The DV2500 operates in On/Off mode.

Solution: Use the setdata service to write the value of the data point pdout . The IODD of the DV2500 shows the structure of the process value (e.g. LED activity). The buzzer is switched via bit 40 of the process value (OFF = 0, ON = 1).

Course:

1. Read the current process value: [Example: Reading an IO-Link process value \(operating mode "IO-Link"\)](#) (→ [109](#)).
2. Set bit 40 of the read value to 1.
3. Write new process value.

Example:

Read process value:

0x0000 0000 004D = 0b0000 0000 0000 0000 0000 0000 0000 0000 0000 0100 1101

New process value:

0b0000 0001 0000 0000 0000 0000 0000 0000 0000 0100 1101 = 0x0100 0000 004D

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdout/setdata",
  "data": {
    "newvalue": "01000000004D"
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1
}
```

Example: Reading the digital input value (operating mode "DI")

Task: The digital input value of port 2 (pin 4) is to be read. The operating mode of the IO-Link port is "Digital Input (DI)".

Solution: Using the service getdata, read the data point pdin. The process value is returned as a hexadecimal value with a length of 1 byte ("00" = LOW level, "01" = HIGH level).

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdin/getdata"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "data": {
    "value": "00"
  }
}
```

Example: Writing the digital output value (operating mode "DO")

Task: The digital output value of port 2 (pin 4) should be switched on (HIGH level). The operating mode of the IO-Link port is "Digital Output (DO)".

Solution: Use the setdata service to write the process value of the data point pdout . The value must be transferred as a hexadecimal value with a length of 1 byte (LOW level = "00", HIGH level = "01").

Request:

```
{
  "code": "request",
  "cid": 10,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdout/setdata",
  "data": {
    "newvalue": "01"
  }
}
```

Response:

```
{
  "cid": 10,
  "code": 200
}
```

9.3.11.5 Reading / writing process data of the ports

Available data points:

Name	Description	Value range	Access
../iqin	Digital input - pin 2 (I/Q) ¹	0: LOW level 1: HIGH level	ro ²
../iqout	Digital output - pin 2 (I/Q) ³	0: LOW level 1: HIGH level	rw ⁴

¹ Parameters only available for IO-Link ports (Class A)

² Read only

³ Parameters only available for IO-Link ports (Class B)

⁴ Read and write; parameter can only be changed if no connection to fieldbus controller is established

9.3.11.6 Accessing the data memory

The user has read and write access to the port data memory.

Substructure: /iolinkmaster/port[x] (x: 1...8)

Available data points:

Name	Description	Value range	Access
../datastorage	Content of the port's data memory (value as Base64-encoded STRING)	E.g. AAAABgE2AAABcwAAAAwBAgAE-ABgBAyoqK-gA8AQICWAA9AQEAAD0DAgASAEg-BAgJYAEgCAgH0AGQBAQE=	rw ¹
../datastorage/size	Size of the data in the data memory	e.g. 59: 59 bytes	ro ²
../datastorage/maxsize	Max. size of the data memory (value in bytes)	e.g. 2062: 2062 bytes	ro ²

¹ Read and write

² Read only

Example: Clone data storage

Task: The data memory of port 2 is to be cloned to port 1.

Solution: Read the data point /iolinkmaster/port[2]/datastorage and then write the read value to data point /iolinkmaster/port[1]/datastorage.

► Read data memory port 1

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/iolinkmaster/port[2]/datastorage/getdata"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/iolinkmaster/port[2]/datastorage/getdata"
  "data": {
    "value": "AAAABgE2AAABcwAAAAwBAGAEABgBAyoqKgA8AQICWAA9AQEAAD0DAgASAEgBAGJYAEgCAGH0AGQBAQE=",
    "timestamp": 1651193795034
  }
}
```

► Write data memory port 2

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/iolinkmaster/port[1]/datastorage/setdata",
  "data": {
    "newvalue": "AAAABgE2AAABcwAAAAwBAGAEABgBAyoqKgA8AQICWAA9AQEAAD0DAgASAEgBAGJYAEgCAGH0AGQBAQE="
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 2,
  "adr": "/iolinkmaster/port[8]/datastorage/setdata"
}
```

The user can then set the mode for device validation and data storage of port 1 to the value "Type compatible V1.1 with Restore" in order to restore the parameter values to the connected, identical IO-Link device.

9.3.11.7 Reading the port type

Substructure: /iolinkmaster/port[x] (x: 1...8)

Available data points:

Name	Description	Value range	Access
../type	Port type	1: CLASS_A_WITH_PORT_POWER_OFF_ON: IO-Link port (Class A) with deactivation / activation of the supply voltage 2: CLASS_B: IO-Link port (class B)	ro ¹

¹ Read only

9.3.11.8 Reading Port events

A port event contains information about events on the IO-Link port (e.g. IO-Link device connected or disconnected, change of the operating mode of the port).

Available data points:

Name	Description	Value range	Access
../portevent	Port event Struktur: 0xYYZZ00 • 0xYY: Connection Status– Status of the connection • 0xZZ: – Device Status– Status of the IO-Link device – Error Code– Error code in case of an error	Connection Status: • 0x00: No IO-Link device connected • 0x40: Incorrect IO-Link device connected • 0x80: IO-Link device connected and in state PREOPERATE • 0xFF: IO-Link device connected and in state OPERATE Device Status: • 0x00: Port deactivated • 0x01: Port activated, but no device connected • 0x02: Port activated and in operating mode "IO-Link" • 0x03: Port activated and in operating mode "DI" • 0x04: Port activated and in operating mode "DO" Error Code: • 0x02: CRC error received • 0x10: Incoming: IO-Link device disconnected; Outgoing: IO-Link device in state OPERATE • 0x1A: Incorrect IO-Link device detected (unspecified error) • 0x1B: Repetitions detected • 0x1E: Short circuit detected on pin 4 • 0x1F: Error in sensor supply • 0x20: Error in actuator supply • 0x21: Error in IO-Link master supply • 0x22: Port has been reset • 0x23: Fallback successful, IO-Link device in SIO mode • 0x24: IO-Link device reached state PREOPERATE • 0x28: Data storage completed, but invalid CRC • 0x29: IO-Link device does not match the stored parameter set • 0x2A: Parameter set of the IO-Link device too large for data storage • 0x2B: Error uploading parameter set to data storage • 0x2C: Error downloading parameter set from data storage • 0x2F: Error in data storage function because IO-Link device is locked • 0x32: Download of parameter set completed • 0x33: Upload of parameter set completed • 0x40: Length of process data (inputs) does not match settings • 0x41: Length of process data (outputs) does not match settings • 0x42: Incorrect revision of the IO-Link device • 0x43: Incorrect Vendor ID (V1.1 revision) • 0x44: Incorrect Device ID (V1.1 revision) • 0x45: Incorrect Vendor ID (V1.0 revision) • 0x46: Incorrect Device ID (V1.0 revision) • 0x47: Incorrect serial number • 0x48: Incorrect cycle time	ro ¹

¹ Read only

9.3.12 IO-Link devices

9.3.12.1 Accessing data objects of the IO-Link device

The ifm IoT Core supports access to data objects (parameters, process data, identification information and system commands) of the connected IO-Link devices via ISDU (Index Service Data Unit). Each data object can be addressed via its ISDU index and ISDU subindex (→ IODD of the IO-Link device). If access to the IO-Link device is faulty, the ISDU error code will also be output in the response (→ IODD of the IO-Link device).

Substructure: /iolinkmaster/port[x]/iolinkdevice (x: 1...8)

Applicable services:

Name	Description
../iolreadacyclic	Read parameters of an IO-Link device acyclically
../iolwriteacyclic	Write parameters of an IO-Link device acyclically

When using the validation level "Type compatible V1.1 Device with Backup + Restore":

After changing a parameter via ISDU write access, the user must end the parameter setting process with the "ParamDownloadStore" system command and activate the data storage mechanism on the IO-Link device.

- ▶ Set the SystemCommand object (ISDU index: 0x0002) to the value 0x05 (command "ParamDownloadStore") via acyclic ISDU write access.
- ▷ The parameter setting process is finished.
- ▷ The data storage mechanism on IO-Link device will be activated.
- ▷ The IO-Link device synchronises changed parameter values with data storage of the IO-Link master.

Example: Reading a parameter value of an IO-Link device

Task: Reading the serial number of the ifm temperature sensor TN2531 at port 2.

Solution: Read the serial number with the iolreadacyclic service from the IO-Link device. Access to the parameter via index 21, subindex 0 (→ IODD description of the sensor).

Request:

```
{
  "code": "request",
  "cid": 4711,
  "adr": "/iolinkmaster/port[2]/iolinkdevice/iolreadacyclic",
  "data": {
    "index": 21,
    "subindex": 0
  }
}
```

Response:

```
{
  "cid": 4711,
  "data": {"value": "4730323134323830373130"},
  "code": 200
}
```

The returned value is displayed in hexadecimal format. Converting the HEX value to a STRING value results in: G0214280710.

Example: Writing a parameter value of an IO-Link device

Task: Setting the output configuration OUT1 of the ifm temperature sensor TN2531 at port 2 to the value "Hnc / hysteresis function, normally closed". The validation level "Type compatible V1.1 Device Backup + Restore" is active.

Solution: Set the parameter [ou1] of the sensor with the service iolwriteacyclic to the value 4. The parameter can be accessed via IO-Link Index 580, Subindex 0 (→ IODD description of the sensor). The value must be passed in hexadecimal format. Converting the STRING value to a HEX value results in: 34.

Request:

```
{
  "code":"request",
  "cid":1,
  "adr":"/iolinkmaster/port[2]/iolinkdevice/iolwriteacyclic",
  "data":{
    "index":580,
    "subindex":0,
    "value":"34"
  }
}
```

Response:

```
{
  "cid":1,
  "code":200,
  "adr":"/iolinkmaster/port[2]/iolinkdevice/iolwriteacyclic"
}
```

The end of the change must then be finalised with the "ParamDownloadStore" system command in order to trigger the data storage mechanism in the IO-Link master (Index: 2, subindex: 0, parameter value: 5).

Request:

```
{
  "code":"request",
  "cid":2,
  "adr":"/iolinkmaster/port[2]/iolinkdevice/iolwriteacyclic",
  "data":{
    "index":2,
    "subindex":0,
    "value":"5"
  }
}
```

Response:

```
{
  "cid":2,
  "code":200,
  "adr":"/iolinkmaster/port[2]/iolinkdevice/iolwriteacyclic"
}
```

9.3.12.2 Reading / writing process data of the IO-Link devices

Substructure: /iolinkmaster/port[x]/iolinkdevice (x: 1...8)

Available data points:

Name	Description	Value range	Access
../pdin	Input data - pin 4 (C/Q)	- Operating mode DI: 0x00: LOW level 0x01: HIGH level - Operating mode IO-Link: - process value as byte sequence	ro ¹
../pdout	Output data - pin 4 (C/Q)	- Operating mode DO: 0x00: LOW level 0x01: HIGH level - Operating mode IO-Link: - process value as byte sequence	rw ¹

¹ Read and write; parameter can only be changed if no connection to fieldbus controller is established

Example: Reading an IO-Link process value (operating mode "IO-Link")

Task: Read the current temperature value of the ifm IO-Link device TN2531 at port 2.

Solution: Use the getdata service to read the data point pdin and then extract the temperature value from the read process value.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/iolinkmaster/port[2]/iolinkdevice/pdin/getdata"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "data": {
    "value": "03C9"
  },
  "adr": "/iolinkmaster/port[2]/iolinkdevice/pdin/getdata"
}
```

The read process value is displayed in hexadecimal format. The process value is given in hexadecimal format. Besides the temperature value the return value comprises additional information (→ IO Device Description (IODD) of the sensor). The temperature value is shown in bits 2 to 15.

0x03C9 = 0b1111001001

Temperature value: 0b11110010 = 242

This means: The current temperature value is 24.2 °C.

Example: Reading an IO-Link process value (operating mode "IO-Link")

Task: The buzzer of the DV2500 at port 2 is to be switched on. The DV2500 operates in On/Off mode.

Solution: Use the setdata service to write the value of the data point pdout. The IODD of the DV2500 shows the structure of the process value (e.g. LED activity). The buzzer is switched via bit 40 of the process value (OFF = 0, ON = 1).

Course:

1. Read the current process value: [Example: Reading an IO-Link process value \(operating mode "IO-Link"\)](#) (→ [109](#)).
2. Set bit 40 of the read value to 1.
3. Write new process value.

Example:

Read process value:

0x0000 0000 004D = 0b0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0100 1101

New process value:

0b0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0100 1101 = 0x0100 0000 004D

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdout/setdata ",
  "data": {
    "newvalue": "01000000004D"
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdout/setdata"
}
```

Example: Reading the digital input value (operating mode "DI")

Task: The digital input value of port 2 (pin 4) is to be read. The operating mode of the IO-Link port is "Digital Input (DI)".

Solution: Using the service getdata, read the data point pdin. The process value is returned as a hexadecimal value with a length of 1 byte ("00" = LOW level, "01" = HIGH level).

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdin/getdata"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "data": {
    "value": "00"
  }
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdin/getdata"
}
```

Example: Writing the digital output value (operating mode "DO")

Task: The digital output value of port 2 (pin 4) should be switched on (HIGH level). The operating mode of the IO-Link port is "Digital Output (DO)".

Solution: Use the setdata service to write the process value of the data point pdout. The value must be transferred as a hexadecimal value with a length of 1 byte (LOW level = "00", HIGH level = "01").

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdout/setdata",
  "data": {
    "newvalue": "01"
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "iolinkmaster/port[2]/iolinkdevice/pdout/setdata"
}
```

9.3.12.3 Accessing interpreted data from the IO-Link device

When the IODD extender function is enabled, the data from the connected IO-Link devices is interpreted using the appropriate IODD. The IODD extender provides the interpreted data in dynamically generated substructures of the ifm IoT Core tree.

The user can read or write to the interpreted data according to its properties.

Substructure: /iolinkmaster/port[x]/iolinkdevice (x: 1...8)

Available substructures:

Name	Description
../identification_interpreted	Interpreted identification information
../parameter_interpreted	Interpreted parameters
../observation_interpreted	Interpreted observation data
../pdin_interpreted	Interpreted input data
../pdout_interpreted	Interpreted output data



Before the first write access to the output data (pdout_interpreted), all output process data must first be written.

9.3.12.4 Reading / writing identification information of the IO-Link device

Substructure: /iolinkmaster/port[x]/iolinkdevice (x: 1...8)

Available data points:

Name	Description	Value range	Access
../status	Status of the connected IO-Link device	0: State not connected - State not connected 1: State preoperate - State PREOPERATE 2: State operate - State OPERATE 3: State communication error - State communication error	ro ¹
../vendorid	ID of the manufacturer	e.g. 310: ifm electronic	ro ¹
../deviceid	ID of the IO-Link device	e.g. 712: ifm pressure sensor, PV7003	ro ¹

Name	Description	Value range	Access
../productname	Product name of the IO-Link device	e.g. PV7003	ro ¹
../serial	Serial number of the IO-Link device	e.g. 000008500706	ro ¹
../revision	Supported IO-Link revision	0: 1.0 – IO-Link revision 1.0 1: 1.1 – IO-Link revision 1.1	ro ¹
../applicationspecifictag	Application specific identification (application tag)	E.g. welding_process	rw ²
../locationtag	Location identifier	E.g. plant1_machine3	rw ²
../functiontag	Function identifier	E.g. pressure1	rw ²

¹ Read only² Read and write

9.3.12.5 Reading IO-Link events

The device supports IO-Link events. IO-Link events are event and error messages. IO-Link events can be generated in the IO-Link master and in the connected IO-Link devices. IO-Link events generated in the IO-Link devices are forwarded to the IO-Link master and stored there.

Substructure: /iolinkmaster/port[x]/iolinkdevice (x: 1...8)

Available data points:

Name	Description	Value range	Access
../iolinkevent	Active IO-Link event Structure of an IO-Link event (7 bytes): 0xIIMMTTVVSSCCCC 0xII: Instance (IO-Link event qualifier - event trigger) 0xMM: Mode (IO-Link event qualifier - event mode) 0xTT: Type (IO-Link event qualifier - event type) 0xVV: Validity (IO-Link event qualifier - validity) 0xSS: Source (IO-Link event qualifier - event source) 0xCCCC: Event Code (IO-Link Event Code)	Instance: 0x00: unknown 0x01: PL (Physical Layer) 0x02: DL (Data Layer) 0x03: AL (Application Layer) 0x04: APPL (Application) Mode: 0x40: One-time event or warning (single shot) 0xC0: Error disappeared 0x80: Error appeared Type: 0x10: Notification 0x20: Warning 0x30: Error Validity: 0x00: valid 0x40: invalid Source: 0x00: IO-Link device 0xFF: IO-Link master Event Code: → IO-Link specification	ro ^{1 2}

¹ Read only² Parameter only available if operating mode pin 4 (US) = IO-Link and IO-Link device is connected to the port

9.3.13 Monitor

9.3.13.1 Using IIoT monitoring

The device offers the option of monitoring data points of the IoT Core and automatically sending notifications when an event occurs.

Substructure: /monitor

Applicable services:

Name	Description
../add	Create monitoring
../remove	Remove monitoring
../getitems	Output information about active monitoring

Example: Add monitoring task

Task: A new monitoring task is to be created with the following properties: The IO-Link input data of the IO-Link device at port 2 should be recorded every 10 s and bundled and sent to the URL 192.168.10.100/datadump every 100 s. All data values should be recorded. The monitoring task should remain active even after the device is restarted.

Solution: Execute the add service with the corresponding parameters on the /monitor structure.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/monitor/add",
  "data": {
    "id": 5,
    "recipient": "http://192.168.10.100/datadump",
    "data_points": ["/iolinkmaster/port[2]/iolinkdevice/pdin"],
    "sample_interval": 10000,
    "notify_interval": 100000,
    "data_changed": false,
    "persist": true
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/monitor/add",
  "data": {
    "id": 5
  }
}
```

Example: Output information about active monitoring

Task: Information on all active monitoring is to be output.

Solution: Execute the getitems service on the monitor structure.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/monitor/getitems"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/monitor/getitems",
  "data": {
    "items": [
      {
        "id": 5,
        "recipient": "http://192.168.10.100/datadump",
        "data_points": [
          "/iolinkmaster/port[2]/iolinkdevice/pdin"
        ],
        "sample_interval": 10000,
        "notify_interval": 100000,
        "data_changed": false,
        "persist": true
      }
    ]
  }
}
```

Example: Remove monitoring

Task: The created monitoring with the identifier 5 is to be deleted again.

Solution: Execute the remove service on the monitor structure.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/monitor/remove",
  "data": {
    "id": 5
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/monitor/remove"
}
```

9.3.14 IODD manager

9.3.14.1 Using IODD manager

Substructure: /ioddextender

Available data points and structures:

Name	Description	Value range	Access
../iodd_folder	Path to folder for saving the IODDs in the device's file system	/etc/ifm/iotcore/iodd (default)	ro ¹
../iodd_files	List of IODDs stored in the iodd_folder directory		ro ¹
../autodownload_iodd	Behaviour of the service when an IO-Link device is connected to one of the ports	0: FALSE - IODD is not downloaded automatically 1: TRUE - IODD is downloaded automatically	rw ²

¹ Read only

² Read and write

Applicable services:

Name	Description
../copy_iodd	Copy IODD (filename, value, crc32)
../download_iodd	Download IODD (vendorid, deviceid)
../delete_iodd	Delete IODD from directory (filename)

Example: Copy IODD

Task: The IODD of the ifm sensor KQ6005 should be saved on the device.

Solution: Use the copy_iodd service to save the IODD of the IO-Link device on the device. The IODD is available on the product detail page of the KQ6005, documentation.ifm.com. The ZIP file with the IODD must be Base64-encoded for processing by the ifm IoT Core, e.g. via www.base64encode.org.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/ioddextender/copy_iodd",
  "data": {
    "filename": "IODD_KQ_IO-Link-11.zip",
    "value": "****base64-encoded string value of IODD file****"
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/ioddextender/copy_iodd",
}
```

Example: Downloading the IODD

Task: The IODD of the ifm sensor KQ6005 should be downloaded from the Internet.

Solution: Use the `iodd_download` service to set the IODD of the IO-Link device (vendor ID: 310, device ID: 371) from the IODDfinder portal. Requirements: Device has access to the Internet and the firewall is configured accordingly.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/ioddextender/download_iodd",
  "data": {
    "vendorid": 310,
    "deviceid": 371
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "ioddextender/iodd_download",
}
```

Example: Deleting the IODD

Task: The IODD of the ifm sensor KQ6005 is to be deleted from the device.

Solution: Use the `iodd_delete` service to delete the IODD of the IO-Link device "IODD_KQ_IO-Link-11.zip" from the device.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/ioddextender/iodd_delete",
  "data": {
    "filename": "IODD_KQ_IO-Link-11.zip"
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/ioddextender/delete_iodd"
}
```

9.3.14.2 Using IODD extender

Substructure: `/ioddextender`

Available data points and structures:

Name	Description	Value range	Access
<code>../status</code>	Status of the IODD extender service	0: stopped - IODD extender stopped 1: running - IODD Extender running (default)	ro ¹

Name	Description	Value range	Access
../settings	Settings of the IODD extender Parameters: poll_interval (value in s)	poll_interval: 1: 1 s ... 30: 30 s ... 3600: 3600 s	rw ²

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/start	Start IODD Extender service (background process)
../status/stop	Stop IODD Extender service
../extend	Execute IODD Extender service once (blocking)

Example: Configure IODD Extender

Task: The interval should be increased from 30 s to 3 min.

Solution: Use the setdata service to write the new interval 180 s (= 3 min) to the data point ioddextender/settings. The IODD Extender service must then be restarted for the change to take effect.

► Stop the IODD Extender service

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "ioddextender/status/stop"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "ioddextender/status/stop"
}
```

► Write the new interval value.

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "ioddextender/settings/setdata",
  "data": {
    "poll_interval": 180
  }
}
```

Response:

```
{
  "code":200,
  "cid":2,
  "adr":"ioddextender/settings/setdata"
}
```

► Restart the IODD Extender service.

Request:

```
{
  "code":"request",
  "cid":3,
  "adr":"ioddextender/status/start"
}
```

Response:

```
{
  "code":200,
  "cid":3,
  "adr":"ioddextender/status/start"
}
```

9.3.15 Device control

9.3.15.1 Controlling the device

Substructure: devicecontrol

Applicable services:

Name	Description
../factoryreset	Reset the device to factory settings ¹
../reboot	Restart the device ¹
../signal	Trigger the flashing of the status LEDs
../gotobootloader	Starting the device in the recovery mode ¹

¹ Service can only be executed if the connection to the fieldbus PLC is interrupted

Example: Set the device to recovery mode

Task: The device should be set to recovery mode

Solution: Execute the gotobootloader service.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/devicecontrol/gotobootloader"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/devicecontrol/gotobootloader"
}
```



Available functions in recovery mode: [Resetting the device via recovery mode \(→ 158\)](#)

9.3.16 Device management

9.3.16.1 Manage external devices

Substructure: /device_management

Applicable services:

Name	Description
../mirror	Mirror external device in the IoT Core of the device
../unmirror	Remove external device from the IoT Core of the device
../adddevice	Add external device to the IoT Core of the device (substructure: /remote)
../removedevice	Remove external device from the IoT Core of the device
../supported_device	Display supported device type
../getdevicelist	Output list of external devices that are stored in the IoT Core of the device

Example: Mirror external device

Task: The IO-Link master with the IP address 192.168.10.200 is to be mirrored in the tree structure of the ifm IoT Core of the device under the alias designation IOLM_01. The mirrored structure should also be available after restarting the device.

Solution: Use the mirror service to integrate the IO-Link master into the tree structure of the ifm IoT Core of the device.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/device_management/mirror",
  "data": {
    "uri": "http://192.168.10.200:80",
    "alias": "IOLM_01",
    "persist": true
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/device_management/mirror"
}
```

Once the service has been successfully executed, the ifm IoT Core of the IO-Link master is available in the /remote substructure.

Application	Devicestatus	Devicecontrol	Devicesettings	Network	Software	Remote IOLM_01	Device_management	Timer[1]	Timer[2]	Processdatamaster	Fieldbussetup	Iolinkmaster
							Deviceinfo	Monitor	Security	Logcollection	Ioddextender	Connections
^ remote												COPY URL
^ IOLM_01												COPY URL
GetIdentity		ifm-AL1590-000250500051/remote/IOLM_01/GetIdentity					type: service profiles: undefined					COPY URL
gettree		ifm-AL1590-000250500051/remote/IOLM_01/gettree					type: service profiles: undefined					COPY URL
querytree		ifm-AL1590-000250500051/remote/IOLM_01/querytree					type: service profiles: undefined					COPY

The user can access process data and parameters via read and write commands. Services of the IO-Link master can also be executed.

Example: Output mirrored devices

Task: An overview of the devices currently mirrored in the ifm IoT Core should be displayed.

Solution: Run the service getdevicelist at the substructure /device_management.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/device_management/getdevicelist"
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/device_management/getdevicelist",
  "data": [
    {
      "adr": "/remote/IOLM_01",
      "uri": "http://192.168.10.200:80",
      "type": "IoT-Core",
      "persist": true
    }
  ]
}
```

Example: Remove external device

Task: The mirrored structure of the IO-Link master should be removed from the ifm IoT Core of the device.

Solution: Run the service unmirror at the substructure /device_management.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/device_management/unmirror",
  "data": {
    "alias": "IOLM_01"
  }
}
```

Response:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/device_management/unmirror"
}
```

9.3.17 Applications

9.3.17.1 Using NTP

The ifm IoT Core of the device offers an NTP application for automatically setting the system time.

Substructure: /application/ntp

Available data points:

Name	Description	Value range	Access
../status	Status of the NTP application	- stopped: Application stopped - running: Application running	ro ¹
../version	Version of the NTP application	E.g. 250	ro ¹
../settings/serverlist	List of NTP servers for automatic setting of the system time	Array with URL of the supported NTP servers	rw ²

¹ Read only

² Read and write

Applicable services:

Name	Description
../status/stop	Stop NTP application
../status/start	Start NTP application

Example: Add NTP server

Task: The NTP server ptbtime1.ptb.de should be added to the list of usable NTP servers.

Solution: Use the setdata service to write the extended server list to the serverlist data point

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/application/ntp/settings/serverlist/setdata",
  "data": {
    "newvalue": [
      "0.pool.ntp.org",
      "1.pool.ntp.org",
      "2.pool.ntp.org",
      "ptbtime1.ptb.de"
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/application/ntp/settings/serverlist/setdata"
}
```

Example: Delete NTP server

Task: The NTP servers 0.pool.ntp.org and ptbtime1.ptb.de should be removed from the list of usable NTP servers.

Solution: Use the setdata service to write the new server list to the serverlist data point.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/application/ntp/settings/serverlist/setdata",
  "data": {
    "newvalue": [
      "1.pool.ntp.org",
      "2.pool.ntp.org"
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/application/ntp/settings/serverlist/setdata"
}
```

9.3.17.2 Using ecologConnect

The ecologConnectelectronic application is installed on the device. The ecologConnectapplication is enabled in the factory settings.

ecologConnect enables access to the device using the ecologOne software from ifm electronic. With ecologOne, the user can make basic settings on the device and save and restore configurations.

Substructure: /application/ecologconnect

Available data points:

Name	Description	Value range	Access
../status	Status of the ecologConnect application	- stopped: - running: (Default)	ro ¹
../version	Version of the ecologConnect application	E.g. 1.0.1.4	ro ¹
../apiversion	Version of the ecologConnect API	E.g. V1.2.0.0	ro ¹

¹ Read only

Applicable services:

Name	Description
../status/stop	Stop ecologConnect application
../status/start	Start the ecologConnect application

Accessing the device with ecologOne



► Note the ecologOne user information!

Requirements:

- ✓ ecologConnect application has been launched.
- ecologOne and install it.
- Start the ecologOne.
- Connect to the device.

9.3.17.3 Using Node-RED

The Node-RED environment is pre-installed on the device. Node-RED is disabled in the factory settings.

Node-RED can be used for programming event-driven applications.

Further information about Node-RED: nodered.org

Substructure: /application/node-red

Available data points:

Name	Description	Value range	Access
../status	Status of the Node-RED application	- stopped: application has been stopped - running: application has been started	ro ¹
../version	Version of the Node-RED environment	E.g. v2.1.6	ro ¹

¹ Read only

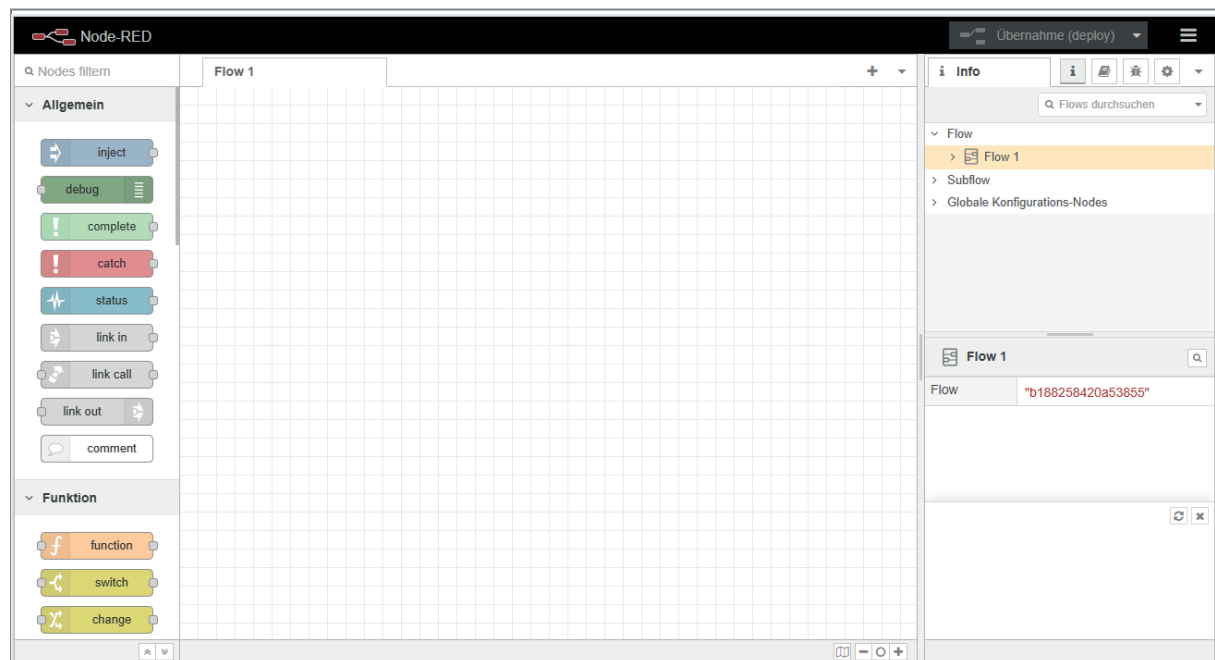
Applicable services:

Name	Description
../status/start	Start Node-RED
../status/stop	Stop Node-RED

Accessing the Node-RED environment

Requirements:

- ✓ The connection between PC / laptop and device is established.
- ✓ Node-RED environment is started.
- ▶ Start web browser.
- ▶ Go to the following URL: `http://<ip-address>:1880`, e.g. `http://192.168.10.43:1880`
- ▷ Graphical user interface of Node-RED is displayed.



9.3.17.4 Using Avahi

Substructure: /application/avahi

Available data points:

Name	Description	Value range	Access
../status	Status of the Avahi application	- stopped: Application stopped - running: Application running	ro ¹
../version	Version of the Avahi application	E.g. 0.8	ro ¹

¹ Read only

Applicable services:

Name	Description
../status/stop	Stop Avahi application
../status/start	Start the Avahi application

9.3.18 Diagnostic and status information

9.3.18.1 Reading status and diagnostic information

Substructure: /processdatamaster

Available data points:

Name	Description	Value range	Access
../voltage	Current voltage of the power supply US (value in mV)	20000: 20000 mV ... 30000: 30000 mV	ro ¹
../voltage/unit	Unit	mV	
../current	Present current intensity of the voltage supply US (value in mA)	0: 0 mA ... 3600: 3600 mA	ro ¹
../current/unit	Unit	mA	
../supervisionstatus	Status of the supply voltage US	0: OK - error-free 1: WARNING: undervoltage - warning: Undervoltage 2: WARNING: overvoltage - warning: Overvoltage 3: FAULT - error	ro ¹
../supervisionstatus_ua	Status supply voltage UA	0: OK - error-free 1: WARNING: undervoltage - warning: Undervoltage 2: WARNING: overvoltage - warning: Overvoltage 3: FAULT - error	ro ¹

¹ Read only

9.3.18.2 Downloading diagnostic data

Substructure: /logcollection

Available data points:

Name	Description	Value range	Access
../journal_log	Log file - Binary Large Object (BLOB)		
../journal_log/chunksize	Size of a data segment of the BLOB (value in bytes)	1048576: 1048576 bytes	ro ¹

¹ Read only

Applicable services:

Name	Description
../journal_log/start_stream_get	Receive Binary Large Object (BLOB)
../journal_log/stream_get	Receive part of a Binary Large Object (BLOB)

9.3.19 Identification information

9.3.19.1 Read identification information of the device

Substructure: /deviceinfo

Available data points:

Name	Description	Value range	Access
../serialnumber	Serial number	E.g. 000194610104	ro ¹
../productname	Product name	IO-LINK MASTER SOL MP 8P IP69K	ro ¹
../productcode	Product designation	AL1591	ro ¹
../ordernumber	Article number	AL1591	ro ¹
../productiondate	Production date (format: DD.MM.YYYY, hh:mm:ss)	E.g. 15.07.2024, 13:16.36	ro ¹
../hwrevision	Hardware version	E.g. AA	ro ¹
../swrevision	Firmware version	E.g. 1.0.3.6	ro ¹
../vendor	Vendor	ifm electronic gmbh	ro ¹
../swversion	Software version	E.g. 1.0.3.6	ro ¹
../hwversion	Hardware version	E.g. 1.2.1.1	ro ¹
../fieldbustype	Fieldbus type:	- Profinet: PROFINET - EtherNetIP: EtherNet/IP	ro ¹

¹ Read only

9.3.19.2 Read identification information of the IO-Link master

Substructure: /iolinkmaster

Available data points:

Name	Description	Value range	Access
../vendorid	IO-Link ID of the manufacturer of the IO-Link master	310: ifm electronic	ro ¹
../masterid	IO-Link ID of the IO-Link masters	1819 : AL1591	ro ¹

¹ Read only

9.3.20 Using notifications

The ifm IoT Core offers the option of sending automated notifications to defined target points. A notification can contain the current values of individual or multiple data points as user data.

Notifications can be triggered on a time-controlled or event-controlled basis.

- [Time-triggered notifications](#) (→ [137](#))
- [Event-triggered notifications](#) (→ [137](#))

Notifications are sent to a destination address. The destination address is specified in URL format: Communication protocol, IP address / domain, port and path (example: `http://192.168.100.254:80/myTopic`).

Supported communication protocols:

- HTTP (Hypertext Transfer Protocol)
- HTTPS (Hypertext Transfer Protocol Secure)
- MQTT (Message Queuing Telemetry Protocol)
- MQTTS (Message Queuing Telemetry Protocol Secure)
- WS (WebSocket)
- TCP (Transmission Control Protocol)

Notifications can be transmitted in the following formats:

- JSON format
- CSV format

The activity duration of a configured notification task can be set:

- Order is only active until the next restart of the device
- Order is permanently active

9.3.20.1 Event-triggered notifications

All elements of the ifm IoT Core with a sub-element of the type event can be used as a trigger for event-triggered notifications.

Available event types:

Name	Description
../datachanged	Node for displaying value changes of the parent data node
../treechanged	Node for displaying structural changes to the ifm IoT Core

Applicable services:

Name	Description
../subscribe	Create notification order
../unsubscribe	Delete notification order

9.3.20.2 Time-triggered notifications

The ifm IoT Core offers 2 timers that can be used as triggers for time-triggered notifications.

The interval time of the timers can be set. Each counting pulse triggers a notification.

Name	Description
../timer[1]/counter/datachanged	Counter of timer 1
../timer[2]/counter/datachanged	Counter of timer 2

Applicable services:

9.3.20.3 Example: Create notification job (JSON format)

Task: The current values of the following parameters are to be sent every 5 seconds via http protocol to a network server with the address 192.168.0.4 to the path /datadump:

- Cyclical input data of the IO-Link device at port 2
- Operating temperature of the device's CPU

The notification request should only be active until the device is next restarted.

Solution:

- Use the setdata service to set the interval of timer 1 to 5 s.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[1]/interval/setdata",
  "data": {
    "newvalue": 5000
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/timer[1]/interval/setdata",
}
```

- Start timer 1 with the start service.

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/timer[1]/status/start",
}
```

Response:

```
{
  "code": 200,
  "cid": 2,
  "adr": "/timer[1]/status/start",
}
```

- Create the notification job with the subscribe service.

Request:

```
{
  "code": "request",
  "cid": 3,
  "adr": "/timer[1]/counter/datachanged/subscribe",
  "data": {
    "callback": "http://192.168.0.4:80/datadump",
    "persist": false,
    "datatosend": [
      "/iolinkmaster/port[2]/iolinkdevice/pdin",
      "/devicestatus/temperature/cpu"
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 3,
  "adr": "/timer[1]/counter/datachanged/subscribe"
}
```

Notification (format: JSON)

```
{
  "code": "event",
  "cid": 3,
  "adr": "",
  "data": {
    "eventno": "6317",
    "srcurl": "/timer[1]/counter/datachanged",
    "payload": {
      "/timer[1]/counter": {"code": 200, "data": 1},
      "/devicestatus/temperature/cpu": {"code": 200, "data": 39},
      "/iolinkmaster/port[2]/iolinkdevice/pdin": {"code": 200, "data": "03B0"}
    }
  }
}
```

9.3.20.4 Example: Change notification order

Task: With the existing notification job, the current operating voltage of the device should be transmitted instead of the temperature of the CPU.

Solution: The existing notification order must be overwritten. To do this, the parameter values for cid and callback in the request must be the same as those of the existing notification request.

Request:

```
{
  "code": "request",
  "cid": 3,
  "adr": "/timer[1]/counter/datachanged/subscribe",
  "data": {
    "callback": "http://192.168.0.4:80/datadump",
    "datatosend": [
      "/iolinkmaster/port[2]/iolinkdevice/pdin",
      "/processdatamaster/voltage"
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 3,
  "adr": "/timer[1]/counter/datachanged/subscribe"
}
```

9.3.20.5 Example: Create notification order (CSV format)

Task: The current values of the following parameters are to be sent to a network server with the IP address 192.168.0.4 to the path /csvdatadump every 10 seconds:

- Cyclical input data of the IO-Link device at port 8
- Operating temperature of the device's CPU

The notification should be transmitted in CSV format, the values should be separated by commas.

Solution:

► Use the setdata service to set the interval of timer 2 to 10 s.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
  "data": {
    "newvalue": 10000
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
}
```

► Start the timer 2 with the start service.

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/timer[2]/status/start",
}
```

Response:

```
{
  "code": 200,
  "cid": 2,
  "adr": "/timer[2]/status/start",
}
```

► Use the subscribe service to create the notification job with the CSV message format.

Request:

```
{
  "code": "request",
  "cid": 10,
  "adr": "/timer[1]/counter/datachanged/subscribe",
  "codec": "csv0",
  "data": {
    "callback": "http://192.168.0.4:80/csvdatadump",
    "datatosend": [
      "/iolinkmaster/port[8]/iolinkdevice/pdin",
      "/devicestatus/temperature/cpu"
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 10,
  "adr": "/timer[1]/counter/datachanged/subscribe",
}
```

Structure of the message:

```
/timer[2]/counter/datachanged,6318,200,1,200,39,200,03B0
```

9.3.20.6 Example: Create notification job (WebSocket)

Task: The current values of the following parameters are to be sent to the data sink /myWsTopic every 60 seconds via an existing WebSocket connection (ws://192.168.0.15:80/mywebsocket):

- Product name of the IO-Link device at IO-Link port 2
- cyclic IO-Link input data of the IO-Link device at IO-Link Port 2

Solution:

► Use the setdata service to set the interval of timer 2 to 60 s.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
  "data": {
    "newvalue": 60000
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/timer[2]/interval/setdata",
}
```

► Start the timer 2 with the start service.

Request:

```
{
  "code": "request",
  "cid": 2,
  "adr": "/timer[2]/status/start",
}
```

Response:

```
{
  "code": 200,
  "cid": 2,
  "adr": "/timer[2]/status/start",
}
```

► Create the notification job with the subscribe service.

Request:

```
{
  "code": "request",
  "cid": 10,
  "adr": "/timer[2]/counter/datachanged/subscribe",
  "data": {
    "callback": "ws:///myWsTopic",
    "datatosend": [
      "/iolinkmaster/port[2]/iolinkdevice/productname",
      "/iolinkmaster/port[2]/iolinkdevice/pdin",
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 10,
  "adr": "/timer[2]/counter/datachanged/subscribe",
}
```

If the notifications are to be transmitted via the existing WebSocket connection, but without a special data sink, the callback parameter is not required.

Request:

```
{
  "code": "request",
  "cid": 11,
  "adr": "/timer[1]/counter/datachanged/subscribe",
  "data": {
    "datatosend": [
      "/iolinkmaster/port[2]/iolinkdevice/productname",
      "/iolinkmaster/port[2]/iolinkdevice/pdin"
    ]
  }
}
```

Response:

```
{
  "code": 200,
  "cid": 11,
  "adr": "/timer[2]/counter/datachanged/subscribe",
}
```

9.3.20.7 Example: Output all active notification orders

Task: Output information on all active notification jobs.

Solution: Use the getsubscriberlist service to output a list of all active notification orders.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/getsubscriberlist",
}
```

Response:

```
{
  "code": 200,
  "cid": 1,
  "adr": "/getsubscriberlist",
  "data": [
    {
      "adr": "/timer[1]/counter/datachanged",
      "callbackurl": "http://192.168.10.1:80/datadump",
      "datatosend": [
        "/iolinkmaster/port[2]/iolinkdevice/pdin",
        "/devicestatus/temperature/cpu"
      ],
      "persist": true,
      "subscribeid": 3,
      "cid": 3
    }
  ]
}
```

9.3.20.8 Example: Delete notification order

Task: The existing notification order is to be deleted.

Solution: Delete the notification job with the unsubscribe service. The notification order can be addressed via the parameters adr and callback.

Request:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[1]/counter/datachanged/unsubscribe",
  "data": {
    "callback": "http://192.168.0.4:80/datadump"
  }
}
```

Response:

```
{
  "code": "request",
  "cid": 1,
  "adr": "/timer[1]/counter/datachanged/unsubscribe",
}
```

9.4 PROFINET

9.4.1 Note: Startup package

ifm electronic provides Startup Packages for integrating the device into fieldbus applications with controllers from various manufacturers.

The startup packages can be downloaded free of charge from the ifm website: documentation.ifm.com

9.4.2 Install the GSD file

To map the device in a PROFINET projection software, ifm provides an GSD file. The device description file contains identification information, supported parameters and process data. The user can download the GSD file via documentation.ifm.com.

To add the device to the hardware catalogue of the PROFINET projection software:

- ▶ Download the GSD file of the device.
- ▶ Launch the PROFINET projection software.
- ▶ Install the GSD file.
- ▷ The device is added to the hardware catalogue of the PROFINET projection software.
- ▷ The PROFINET projection software can access the device functions and data.

9.4.3 Integrating the device into a project

Using the installed device description in the hardware catalogue, the device can be added to a PROFINET project.

Requirements:

- ▶ Create a new project or open an existing one.
- ▶ Open the [Devices & networks] view.
- ▶ Add the necessary components to the network (e.g. PROFINET controller).
- ▶ Select the device in the hardware catalogue and add it to the network using drag & drop.
- ▶ Establish a logical PROFINET IO connection between the device and the PROFINET controller.
- ▶ Set the IP configuration of the device's PROFINET interface.
- ▷ The device has been integrated into the project.

9.4.3.1 Use S2 redundancy

The device supports the S2 redundancy. S2 redundancy provides a solution to build a redundant system for implementation of fail-safe systems. The following constraints apply:

- Both Application Relations (SR-AR) use the same configuration (submodule settings, connection parameters).
- Only one AR can act as a primary SR-AR.
- Diagnostic messages and alarms are only reported to the primary SR-AR.
- Data records can be read via any SR-AR.
- Data records can only be written via the following SR-AR:
 - primary SR-AR
 - SR-AR that first accesses the device while establishing the connection

9.4.3.2 Use Configuration-in-Run

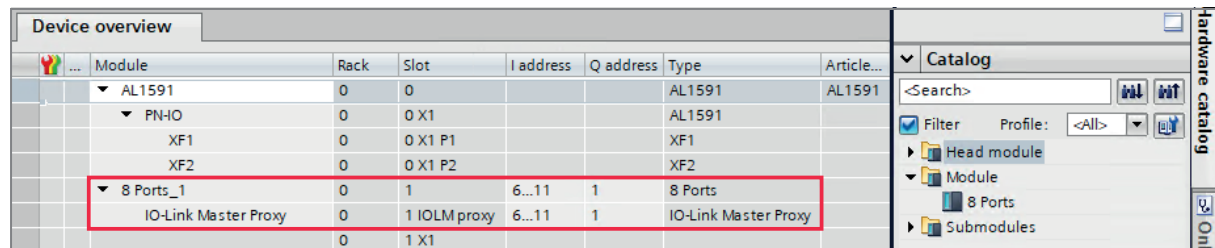
The device supports the Configuration-in-Run (CiR) function. CiR enables the user to load changes and extensions of the hardware configuration in the PROFINET configuration software to the PLC without needing to put the PLC into the “stop” state first.

9.4.4 Configuring the IO-Link master

The IOLM proxy module provides access to the basic settings of the IO-Link master.

The [8 Ports] module is permanently assigned to slot 1.

The [IO-Link Master Proxy] submodule is permanently assigned to the [1 IOLM proxy] slot.



Module	Rack	Slot	I address	Q address	Type	Article...
AL1591	0	0			AL1591	AL1591
PN-IO	0	0 X1			AL1591	
XF1	0	0 X1 P1			XF1	
XF2	0	0 X1 P2			XF2	
8 Ports_1	0	1	6...11	1	8 Ports	
IO-Link Master Proxy	0	1 IOLM proxy	6...11	1	IO-Link Master Proxy	
	0	1 X1				

Available parameters:

- [Parameters: 8 ports \(→ 226\)](#)

Requirements:

- ✓ Device is integrated in PROFINET project.
- Open the [Device view].
 - ▷ The [Device overview] tab displays the device structure.
- In column [Module] click on [IO-Link Master Proxy] in slot [1 IOLM proxy].
 - ▷ The [Properties] view appears.
 - ▷ The [General] tab displays the properties of the module.
- Set the parameters.
- ▷ The device has been configured.

9.4.5 Setting the function of the ports

The function of the device ports is set via the IOLD proxy submodules.

The selected IOLD proxy submodule determines the following properties of the port:

- Operating mode pin 4 (US)
- Type and length of the process data
- Adjustable parameters

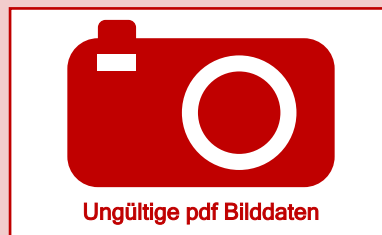
Available IOLD proxy submodules:

- [IOLD Proxy Submodule: IO-Link master proxy \(→ 225\)](#)

Each port of the device can be addressed via a separate slot.

Port:	X1	X2	X3	X4	X5	X6	X7	X8
Slot	1 X1	1 X2	1 X3	1 X4	1 X5	1 X6	1 X7	1 X8

Example configuration:



-  A specific IOLD proxy module must be assigned to each slot.
- ▶ Assign the IOLD proxy submodule [Deactivated] to unused ports.

To set the IO-Link function of the ports:

Requirements:

- ✓ Hardware catalogue is open.
- ✓ The [Device view] is open.
- ▶ In the hardware catalogue: drag and drop the required IOLD proxy module from the [Submodules] folder to the slot of the port.
 - ▷ The IOLD proxy module will be loaded with the factory settings.
 - ▷ IEC addresses for process data of the port will be generated.
- ▶ Link the IEC addresses of the process data with the variables.
- ▶ Repeat the process for further ports.
- ▶ Save the project.
- ▷ Function of the IO-Link port is set.
- ▷ The linked variables can be used to access the process data from the application.

9.4.6 Parameter setting of the ports

The ports are configured via the IOLD proxy submodules. The following port properties are set during parameter setting:

The function set for the port determines which parameters are available.

Available parameters:

- [Parameters: Deactivated \(→ !\[\]\(7d7564dcec72f0f4ed2bfde74f648bcf_img.jpg\) 227\)](#)
- [Parameters: Digital Input \(→ !\[\]\(0777cd2863d4fdd306066f04b18abcab_img.jpg\) 228\)](#)
- [Parameters: Digital Output \(→ !\[\]\(2d3466463ed0af25c9d94b47abdd45e3_img.jpg\) 228\)](#)
- [Parameters: IO-Link - Input \(→ !\[\]\(de6e64b9a420b3ed2c17c03d3964a21d_img.jpg\) 228\)](#)
- [Parameters: IO-Link - Output \(→ !\[\]\(0354cb083a388dbeeeb47706579a559f_img.jpg\) 230\)](#)
- [Parameters: IO-Link - Input / Output \(→ !\[\]\(a2232bc10b4d64cfd08b50fd34ed2706_img.jpg\) 232\)](#)

To configure the ports of the device:

Requirements:

- ✓ The [Device view] is open.
- ✓ IOLD proxy submodules are assigned to the slots of the ports.
- ▶ Click on the slot assigned to the port.
 - ▷ The [Properties] view appears.

- ▶ Select the [Module parameters] area.
 - ▷ The [General] tab displays the properties of the module.
- ▶ Set the parameters.
- ▶ Repeat the process to configure further ports.
- ▶ Save the project.
 - ▷ The parameters of the ports are set.

9.4.7 Use I&M data records

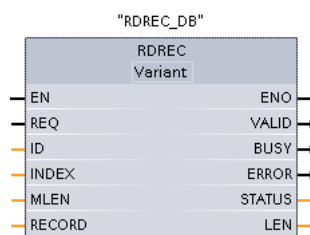
The unit supports the I&M data records I&M0 to I&M3 (→ [I&M data](#) □ 241).

- The data record I&M0 contains ID information. The I&M0 data record is available for each module / submodule of the device.
- In the data records I&M1...I&M3, the user can store application and device-specific information.

I&M data records can be accessed acyclically. Access to the I&M data records takes place index-based.

Note: Read data record

- ▶ Read the data record with the function block RDREC.

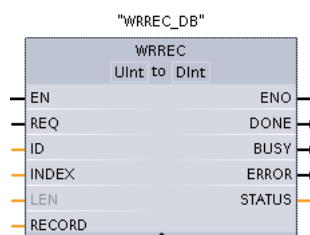


Specific parameters:

- Input ID: HW_ID of the subslot (project-specific); properties of the subslot
- Input INDEX: index of the data record

Note: Write data record

- ▶ Write the data record with the function block WRREC.
- ▶ Observe read and write permissions!



Specific parameters:

- Input ID: HW-ID of the slot (project-specific); properties of the slot
- Input INDEX: index of the data record

9.5 EtherNet/IP

9.5.1 Note: Startup package

ifm electronic provides Startup Packages for integrating the device into fieldbus applications with controllers from various manufacturers.

The startup packages can be downloaded free of charge from the ifm website: documentation.ifm.com

9.5.2 Integrating the device into a project

9.5.2.1 Top-down mode

In configuration mode Top-Down, the device parameters are set in the EtherNet/IP configuration software. The created configuration is transferred to the device and enabled when the connection to the EtherNet/IP controller is established.

Adding a device to the project

The device is integrated into the project using the installed EDS file.

Requirements:

- ✓ The EDS file is installed.
- ✓ A project with Ethernet/IP controller and Ethernet communication module has been created.
- ▶ In the Controller Organizer: Right-click on node [Ethernet] of the EtherNet/IP controller.
 - ▷ The context menu appears.
- ▶ Select [New module...] in the context menu.
 - ▷ A dialogue window appears.
- ▶ Select AL1591 and click on [Create] .
 - ▷ The [New module] window appears.
- ▶ Configure the IP settings of the device.
- ▶ Save the project.

Setting the connection type

The connection type determines which object instances of Configuration Assembly (parameters), Input Assembly (input data) and Output Assembly (output data) are used in the configuration.

Available connection types:

Connection type	Description	Configuration Assembly	Input Assembly	Output Assembly
Exclusive Owner IO	Process data (inputs, outputs) incl. PQI	Instance 199	Instance 100	Instance 150
Exclusive Owner IO-Acyc-Diag-EnMo	- Process data (inputs, outputs) incl. PQI - Acyclic command channel - Diagnostic information, IO-Link events - Energy monitoring	Instance 199	Instance 101	Instance 151
Input Only IO	Process data (inputs) incl. PQI	Instance 199	Instance 100	Instance 193

Connection type	Description	Configuration Assembly	Input Assembly	Output Assembly
Input Only IO-Acyc-Diag-En-Mo	• Process data (inputs) incl. PQI • Acyclic command channel • Diagnostic information, IO-Link events • Energy monitoring	Instance 199	Instance 101	Instance 193
Listen Only IO	• Process data (inputs) incl. PQI	Instance 199	Instance 100	Instance 192
Listen Only IO-Acyc-Diag-En-Mo	• Process data (inputs) incl. PQI • Acyclic command channel • Diagnostic information, IO-Link events • Energy monitoring	Instance 199	Instance 101	Instance 192



The device only supports an active connection of types Exclusive Owner and Input. Connections of type Listen only can be selected as the second active connection.

To set the connections:

Requirements:

- ✓ The project is open.
- ✓ The device has been added to the project using the EDS.
- In the [Controller Organizer]: Double-click on the device node.
 - ▷ A dialogue window appears.
- Click on [Change...].
 - ▷ The [Module Definition] window appears.
- Select the desired connection in the [Connections] list (type: Exclusive Owner or Input).
- If necessary, set the data type to SINT.
- Optional: Select a second connection (type: Listen only).
- Click [OK] to adopt the changes.
 - ▷ The connection type is set.
 - ▷ RSLogix 5000 generates the [Controller Tags] of the device (Configuration, Input, Output) based on the selected connection type.

Device configuration

The device configuration includes the configuration of the IO-Link master and the IO-Link ports.

Available parameters:

- Length of the process data
- Arrangement of the bytes in the process data
- Ports X1...X8: Port configuration
 - Operating mode pin 4 (C/Q)
 - Cycle time of the data transmission of the IO-Link ports
 - Device validation and data storage mode
 - Device ID and manufacturer ID for device validation
 - Fallback mode for output data Pin 4 (C/Q)
 - Fallback values for digital output data Pin 4 (C/Q)
- Ports X1...X4: Operating mode Pin 2 (I/Q)

- Ports X1...X4: Fallback value for digital output data Pin 2 (I/Q)

The parameters of the device are defined in the following object:

- [Configuration assembly \(instance 199\) \(→ □ 176\)](#)

The controller tags provide access to the parameters.

Requirements:

- ✓ The device is integrated into the project.
- ▶ In the [Controller Organizer]: Double-click the [Controller Tags] node.
 - ▷ The [Controller Tags] window appears.
 - ▷ The structure [AL159x.C] > [AL159x.C:Data] contains the parameters.
- ▶ Set the parameters.
- ▶ Save the project.
- ▷ The device configuration has been created.

Transferring the configuration to the device

To activate the configuration on the device, the device configuration must be loaded onto the PLC.

Requirements:

- ✓ The project is open.
- ✓ The device has been added to the project using the EDS.
- ✓ The device has been configured.
- ▶ Select the [Communications] > [Download] menu.
 - ▷ Project is loaded onto the controller.
 - ▷ The configuration is saved on the device and activated.
 - ▷ The control changes to Run Mode.

To test the connection to the device:

- ▶ In the [Controller Organizer]: double-click on the [Controller Tags] node.
 - ▷ The [Controller Tags] window appears.
- ▷ Variable [AL159x:I] > [AL159x:I.ConnectionFaulted] shows the status of the connection (0: OK, 1: Error)

9.5.2.2 Independent mode

In configuration mode Independent Mode, the device is configured using the parameter setting software or the IoT Core. In the fieldbus configuration software, the device is integrated into the project as a generic device. The length of the input data and output data must then be set according to the device configuration. The settings of the configuration assembly object are ignored when the project is transferred to the controller.

Device configuration

The device can be configured locally with the following options:

- ifm moneo: [Setting the EtherNet/IP configuration mode \(→ □ 34\)](#)
- ifm IoT Core REST API: [EtherNet/IP Setting the configuration mode \(→ □ 100\)](#)
- ifm IoT Core Visualizer: [EtherNet/IP Setting the configuration mode \(→ □ 63\)](#)

The user must make the following parameter settings:

Fieldbus settings

Parameter	Description	Parameter value
Independent mode	Enable independent mode	On
Process data length	Length of the process data	According to the requirements of the application
Swap	Arrangement of the bytes in the process data	According to the requirements of the application

Port settings

Parameter	Description	Parameter value
Mode Pin2 UA	Operating mode Pin 2 (I/Q)	According to the requirements of the application
Mode Pin4 US	Operating mode Pin 4 (C/Q)	IO-Link
Fail-safe digital out	Fallback value digital output Pin 4 (C/Q)	According to the requirements of the application
Fail-safe DO/UA	Fallback value digital output Pin 2 (I/Q)	According to the requirements of the application

To configure the device:

- ▶ Access the device with the desired parameter setting option:
- ▶ Set the parameters.
- ▶ Write the changed values to the device.
- ▷ The device has been configured.

Adding a device to the project

The device is integrated into the project as a generic Ethernet module.

The connection settings must be entered manually. The connection settings define the structure and scope of the cyclic input data (Input Assembly) and output data (Output Assembly). The size of the input assembly and the output assembly depends on the process data length set on the device.



The connection settings must match the set process data length.

Connection type	Input Assembly	Set process data length				
	Output Assembly	2 bytes	4 bytes	8 bytes	16 bytes	32 bytes
	Configuration Assembly					
Exclusive Owner IO	100	Length: 36 bytes	Length: 52 bytes	Length: 84 bytes	Length: 148 bytes	Length: 276 bytes
	150	Length: 18 bytes	Length: 34 bytes	Length: 66 bytes	Length: 130 bytes	Length: 258 bytes
	199	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
Exclusiv Owner IO-Acyc-Diag-EnMo	101	Length: 210 bytes	Length: 226 bytes	Length: 258 bytes	Length: 322 bytes	Length: 450 bytes
	151	Length: 62 bytes	Length: 78 bytes	Length: 110 bytes	Length: 174 bytes	Length: 302 bytes
	199	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
Input Only IO	100	Length: 36 bytes	Length: 52 bytes	Length: 84 bytes	Length: 148 bytes	Length: 276 bytes
	193	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
	199	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
Input Only IO-Acyc-Diag-En-Mo	101	Length: 210 bytes	Length: 226 bytes	Length: 258 bytes	Length: 322 bytes	Length: 450 bytes
	193	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes

Connection type	Input Assembly	Set process data length				
	Output Assembly	2 bytes	4 bytes	8 bytes	16 bytes	32 bytes
	Configuration Assembly					
Input Only IO-Acyc-Diag-En-Mo	199	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
Listen Only IO	100	Length: 36 bytes	Length: 52 bytes	Length: 84 bytes	Length: 148 bytes	Length: 276 bytes
	192	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
	199	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
Listen Only IO-Acyc-Diag-En-Mo	101	Length: 210 bytes	Length: 226 bytes	Length: 258 bytes	Length: 322 bytes	Length: 450 bytes
	192	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes
	199	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes	Length: 0 bytes

Requirements:

- ✓ A project with Ethernet/IP controller and Ethernet communication module has been created.
- ✓ The project is open.
- ▶ In the [Controller Organizer]: right-click on the node [Ethernet] (Ethernet communication module).
 - ▷ The context menu appears.
- ▶ Select [New module...] in the context menu.
 - ▷ A dialogue window appears.
- ▶ Select the device type [ETHERNET-MODULE] (Generic Ethernet module).
- ▶ Click on [Create].
 - ▷ The [New module] window appears.
- ▶ Enter the name of the device, e.g. AL159x.
- ▶ In the field [Comm Format], select [Data – SINT].
- ▶ Configure the IP settings of the device.
- ▶ Enter the length of the input data and output data in the [Connection Parameters] area (→ table).
- ▶ Click [OK] to adopt the changes.
 - ▷ The device node is added to the project tree.
 - ▷ The device is configured for operation in independent mode.

Transferring the configuration to the device

To transfer the configuration to the device:

Requirements:

- ✓ The project is open.
- ✓ The device has been added to the project using the EDS.
- ✓ The device has been configured.
- ▶ Select the [Communications] > [Download] menu.
 - ▷ Project is loaded onto the controller.
 - ▷ The configuration is saved on the device and activated.
 - ▷ The control changes to Run Mode.

9.5.3 Accessing the process data

The cyclical process data of the device is transmitted via the Controller Tags.

The type, scope and position of the process data within the assembly instances depend on the selected connection type or the set process data length.

Available data:

- Ports X5...X8: Digital input data (pin 2 + pin 4)
- Ports X5...X8: Digital output data (pin 4)
- Ports X1...X4: Digital input data (pin 4)
- Ports X1...X4: Digital output data (pin 2 + pin 4)
- Ports X1...X8: IO-Link input data + output data

Cyclical input data is part of the following input assembly instances

- [Input assembly \(instance 100\) \(→ □ 178\)](#)
- [Input Assembly \(Instance 101\) \(→ □ 182\)](#)

Cyclical output data is part of the following output assembly instances.

- [Output Assembly \(Instance 150\) \(→ □ 188\)](#)
- [Output Assembly \(Instance 151\) \(→ □ 190\)](#)



The Port Qualifier Information (PQI) is available in the diagnostic and status information to check the validity of the cyclical process data.

- ▶ To check the validity of the process data, analyse the Port Qualifier Information (PQI).



If the EtherNet/IP connection is interrupted, the Port Qualifier Information (PQI) indicates that the process data is valid.

- ▷ Unintended effects on the control process
- ▶ Take suitable measures to detect the interruption of the EtherNet/IP connection.

Requirements:

- ✓ The project is open.
- ✓ The device is integrated into the project.
- ✓ The device has been configured.
- ▶ In the [Controller Organizer]: double-click on the [Controller Tags] node.
 - ▷ The [Controller Tags] window appears.
 - ▷ The structure [AL159x.I] > [AL159x.I.Data] contains the cyclic input data.
 - ▷ The structure [AL159x.O] > [AL159x.O.Data] contains the cyclic output data.
- ▶ If required: Link the cyclic input data and output data with the application variables.
- ▶ Save the project.

9.5.4 Accessing IO-Link events

The device supports the transmission of up to 3 IO-Link events per port. The IO-Link events are mapped as part of the diagnostic information in the cyclical input data.

The IO-Link events are part of the following input assembly instances:

- [Input Assembly \(Instance 101\) \(→ □ 182\)](#)

To access the IO-Link events:

Requirements:

- ✓ Operating mode pin 4 (US) of the port is IO-Link.

- ✓ The IO-Link device is connected to the port.
- ✓ The project is open.
- ✓ The device is integrated into the project.
- ✓ The device has been configured.
- ✓ A connection type with the required assembly instance is set.
- ▶ In the [Controller Organizer]: double-click on the [Controller Tags] node.
 - ▷ The [Controller Tags] window appears.
 - ▷ Structure [AL159x.I] > [AL159x.I:Data] contains the IO-Link events of the IO-Link ports.
- ▶ If required: link the IO-Link events with the application variables.
- ▶ Save the project.

9.5.5 Accessing diagnostic and status information

Diagnostic and status information is transmitted in the cyclical input data.

Available information:

- Status indication supply voltages US / UA of the device
- Short-circuit and overload indicator for the IO-Link ports
- Port Qualifier Information (PQI) of the IO-Link ports

The diagnostic and status information is part of the following input assembly instances:

- [Input assembly \(instance 100\) \(→ □ 178\)](#)
- [Input Assembly \(Instance 101\) \(→ □ 182\)](#)

Requirements:

- ✓ The project is open.
- ✓ The device is integrated into the project.
- ✓ The device has been configured.
- ▶ In the [Controller Organizer]: double-click on the [Controller Tags] node.
 - ▷ The [Controller Tags] window appears.
 - ▷ The structure [AL159x.I] > [AL159x.I:Data] contains the diagnostic and status information.
- ▶ If required: link the diagnostic and status information with the application variables.
- ▶ Save the project.

In addition, the diagnostic and status information of the CIP object classes of the device can be read acyclically, [Using acyclic services \(→ □ 156\)](#).

9.5.6 Using energy monitoring

The energy monitoring information is transmitted in the cyclical input data:

Available information:

- Voltage and current values of the supply voltages US / UA of the device
- Current values of the voltage UA of the IO-Link ports (pin 2)
- Current values of the voltage US of the IO-Link ports (pin 1+ pin 4)

Energy monitoring is part of the following input assembly instances:

- [Input Assembly \(Instance 101\) \(→ □ 182\)](#)

Requirements:

- ✓ The project is open.
- ✓ The device is integrated into the project.
- ✓ The device has been configured.
- In the [Controller Organizer]: double-click on the [Controller Tags] node.
 - ▷ The [Controller Tags] window appears.
- ▷ Structure [AL159x.I] > [AL159x.I:Data] contains the information for energy monitoring.
- If required: Link information on energy monitoring with application variables.
- Save the project.

9.5.7 Accessing IO-Link devices

The device supports access to the connected IO-Link devices from the application. The Indexed Service Data Unit (ISDU) is used to access the parameters, process data, diagnostics and status information. Each data object of an IO-Link device can be addressed via index and subindex.

The available data objects depend on the IO-Link device, → IODD description of the IO-Link device.

Available access options:

- Via the acyclic command channel within the cyclic process data, [Acyclic command channel](#) (→ [192](#))
- Via the manufacturer-specific CIP object class IO-Link Request, [IO-Link Request Object \(Class Code: 0x80\)](#) (→ [213](#))
 - Available services:

Service	Description	Reference
Read ISDU	Read ISDU object	Read ISDU (→ 214)
Write ISDU	Write ISDU object	Write ISDU (→ 215)

9.5.8 Using acyclic services

The device supports the following mechanisms for acyclic read and write access to data on the IO-Link ports and the connected IO-Link devices.

- Acyclical command channel within the cyclical process data: [Using the acyclic command channel](#) (→ [156](#))
- EtherNet/IP mechanisms for accessing objects in the object dictionary: [EtherNet/IP mechanisms for acyclic commands](#) (→ [157](#))

9.5.8.1 Using the acyclic command channel

The device features an acyclic command channel. This provides acyclical read and write access to the parameters of the device and the connected IO-Link devices. The command channel uses special sections within the cyclic input data and output data.

An acyclic command consists of a request and a response. The command request is transmitted in the Output Assembly. The command response of the IO-Link master is transmitted in the input assembly.

Structure of the acyclic command channel: [Acyclic command channel](#) (→ [192](#))

To use the acyclic command channel:

Requirements:

- ✓ The device has been integrated into the project.
- ✓ In the Top-Down Mode: a connection type is active that provides acyclical ranges in the input data and output data.
- ✓ In the Independent Mode: The device has been configured correctly locally.

- Execute the read and write commands to the acyclic areas.

Principle of the command channels

General procedure of acyclic communication:

1. Write Command Request

- In the request channel: write the required command data (except [Trigger])
- Set [Trigger] = 1.
- ▷ Change to [Trigger] = 1 signals a new command.
- ▷ In the response channel: bytes are set to 0.
- ▷ Processing of the command is started.

2. Check status

- In the response channel: check [Handshake].
 - ▷ If [Handshake] <> 0: command processing finished, continue with step 3.
 - ▷ If [Handshake] == 0: command is still being processed, repeat step 2.

3. Read Command Response

- In the response channel: read returned user data.
- In the request channel: set [Trigger] = 0.
- ▷ Command processing completed.
- ▷ New command can be executed.

9.5.8.2 Acyclic port commands

For the acyclic access to the configuration of the IO-Link port of the device, the following commands are available:

Command	Description	Reference
Set Mode	Set the operating mode of the IO-Link port	Set Mode (0x10) (→ ¶ 197)
Set Validation ID / Data Storage	Set the device validation and the data storage of the IO-Link port	Set validation ID / data storage (0x20) (→ ¶ 199)
Set Fail-safe Data Pattern	Set the behaviour of the outputs when the Ethernet/IP connection is interrupted and the corresponding fallback values of the IO-Link port	Set fail-safe data pattern (0x30) (→ ¶ 201)
Set Port Power Mode	The command controls the supply voltage US (pin 1) of the IO-Link port	Set Port Power Mode (0x40) (→ ¶ 203)

The port commands use the mechanisms of the acyclic command channel.

9.5.8.3 EtherNet/IP mechanisms for acyclic commands

The device supports the acyclic access to the objects in the object directory, [Object directory \(→ ¶ 205\)](#). The EtherNet/IP command Message (MSG) can be used for acyclic read and write access.



Detailed information on the Message (MSG) command: → User documentation RSLogix 5000

10 Maintenance, repair and disposal

The operation of the unit is maintenance-free.

It is not possible to repair the device.

- ▶ After use, dispose of the unit in an environmentally friendly way in accordance with the applicable national regulations.

10.1 Cleaning

- ▶ Clean the surface of the device when necessary.
- ▶ Disconnect the unit from the voltage supply.
- ▶ Clean the device from dirt using a soft, chemically untreated and dry cloth.
- ▶ Carry out cleaning, taking into account the protection class of the appliance and the connection technology used.

The cleaning agents must not damage the appliance materials or the appliance labelling.

- ▶ Check cleaning agents for their suitability for the appliance materials (→ Technical data) and the connection technology applied.

10.2 Firmware update

The firmware of the device can be updated using the following methods:

- ifm IoT Core Visualizer: [Firmware update](#) (→ [70](#))
- ifm moneo: [Updating the firmware](#) (→ [43](#))
- Recovery mode: [Resetting the device via recovery mode](#) (→ [158](#))

10.3 Resetting the device via recovery mode

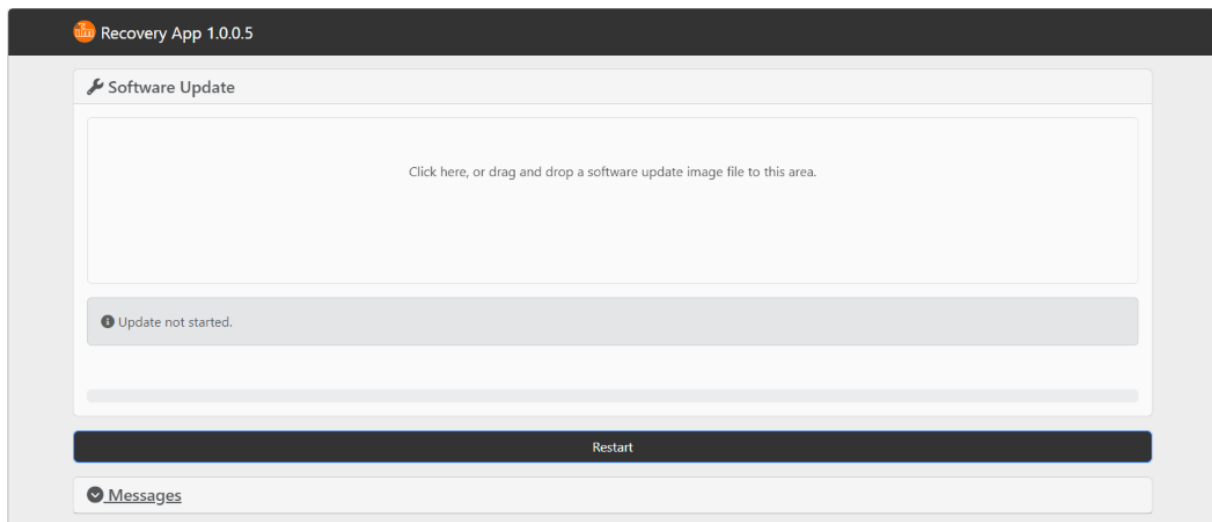
The user can reset the device to the factory settings or update the device's firmware via the recovery mode.



When the device is started in recovery mode, the user-specific data is retained. The user-specific data is only deleted when the firmware is subsequently overwritten.

Requirements:

- ✓ Firmware file is available.
- ✓ PC / laptop is connected to the IIoT interface of the device.
- ✓ PC / laptop and device are in the same IP subnet.
- ✓ IP address of the IIoT interface is known.
- ▶ Disconnect the appliance from power.
- ▶ Apply the 24 V DC voltage to pin 5 of port X8 and connect the device to the power supply at the same time.
 - ▷ The status LED RDY lights up red.
 - ▷ The device starts in the recovery mode.
- ▶ On the laptop / PC: Start web browser.
- ▶ Enter the following URL: `http://<IP address of the device>:8080`, e.g. `http://192.168.10.42:8080`
 - ▷ The Recovery Mode web interface appears.



- ▶ Drag and drop the firmware file onto the corresponding area.
 - ▷ The device recognises firmware and starts installation.
 - ▷ View shows the status of the installation process.

After successful installation:

- ▶ Disconnect pin 5 of port X8 from power.
- ▶ Click on [Restart].
 - ▷ The device reboots.
- ▷ The device is in the delivery state.

11 Appendix

11.1 ifm IoT Core

11.1.1 Profiles

Profile	Description
blob	Binary Large Object
deviceinfo	Identification information of a device
devicetag	Device-specific identification
iolinkdevice_full	IO-Link device
iolinkmaster	IO-Link master
network	Network
parameter	Parameter
processdata	Process data
service	Service
software	Software
software/uploadable-software	upgradeable software
timer	Counter

11.1.2 Types

Type	Description
data	Data point
device	Root element representing a device
event	Event that can be triggered by the firmware and sends notifications
service	Service that can be addressed from the network
structure	Structural element (e.g. a folder in the file system)

11.1.3 Services

11.1.3.1 Service: abortinstall

Name: abortinstall

Description: The service aborts a running installation process.

Request ("data" field): none

Response ("data" field): none

11.1.3.2 Service: add

Name: add

Description: The service creates a new monitoring task.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
data_points	mandatory	STRING[1...n]	List with n addresses of the elements (type: data) that are to be monitored
sample_interval	mandatory	NUMBER	Interval at which the data values are recorded (value in ms)
notify_interval	mandatory	NUMBER	Interval at which a notification is to be sent with the monitored data values (value in ms)
data_changed	Optional	BOOLEAN	Frequency of data value storage - false: Save all recorded data values - true: Only save the data value if there is a change to the last recorded data value
recipient	mandatory	STRING	URI to which the notifications are to be sent
id	mandatory	NUMBER	Identification of the monitoring task 0...2147483647 -1: Automatic allocation of the identifier
persist	Optional	BOOLEAN	Gültigkeitsdauer der Überwachungsaufgabe - false: Monitoring task active as long as device is switched on - true: Monitoring task active even after restarting the device

Response ("data" field):

Parameter	Mandatory field	Data type	Description
id	mandatory	STRING	Monitoring identifier

11.1.3.3 Service: adddevice

Name: adddevice

Description: The service adds a new external device to the IoT Core of the device.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
uri	mandatory	STRING	URI of the external device
type	Optional	STRING	Type of external device Value range: Elements of the list supported_devices (Default: "IoT Core")
callbackurl	Optional	STRING	URL for sending event messages from the external device to the device

Parameter	Mandatory field	Data type	Description
alias	Optional	STRING	Identifier of the external device
auth	Optional	AuthInfo_Type	Authentication data for access to the external device
cache_timeout	Optional	NUMBER	Max. time period after which the cache memory must be updated (default: 0)
persist	Optional	BOOLEAN	- false: Save the external device only until the next restart of the device - true: Save the external device even after the next restart of the device
check_connection	Optional	BOOLEAN	- false: Do not check the connection to external device - true: Check the connection to the external device

Response ("data" field):

Parameter	Mandatory field	Data type	Description
adr	Optional	STRING	Address of the added device (only if service was successfully executed)
identity	Optional	OBJECT	Response from the /getidentity service (only if the connection to the remote device has been successfully established)

11.1.3.4 Service: copy_iodd

Name: copy_iodd

Description: The service copies an IODD file (*.xml) or several packaged IODD files (*.zip) to the IODD directory on the device.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
filename	mandatory	STRING	File name under which the content is to be saved
value	mandatory	STRING	File content as Base64-encoded STRING
crc32	Optional	UINT32	CRC32 checksum of the file

Response ("data" field): none

11.1.3.5 Service: delete_iodd

Name: delete_iodd

Description: The service deletes an IODD file on the device.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
filename	mandatory	STRING	Name of the file to be deleted.

Response ("data" field): none

11.1.3.6 Service: download_iodd

Name: download_iodd

Description: The service loads an IODD file from the website ioddfinder.io-link.com. The file is identified via the manufacturer ID and device ID.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
vendorid	mandatory	NUMBER	Manufacturer ID of the IODD
deviceid	mandatory	NUMBER	Device ID of the IODD

Response ("data" field): none

11.1.3.7 Service: enable

Name: enable

Description: The service activates a fieldbus type.

Request ("data" field): none

Response ("data" field): none

11.1.3.8 Service: extend

Name: extend

Description: The service executes the IODD extender process.

Request ("data" field): none

Response ("data" field): none

11.1.3.9 Service: factoryreset

Name: factoryreset

Description: The service sets the parameters of the device to the factory settings.

Request ("data" field): none

Response ("data" field): none

11.1.3.10 Service: getdata

Name: getdata

Description: The service reads the value of a data point and provides it.

Request ("data" field): none

Response ("data" field):

Parameter	Mandatory field	Data type	Description
value	mandatory	STRING	Value of the data point
timestamp	mandatory	STRING	Timestamp of the response; Value as INT64 number in UTC format (milliseconds from 01.01.1970)

11.1.3.11 Service: getdatamulti

Name: getdatamulti

Description: The service sequentially reads the values of multiple data points and outputs them. The value and a diagnostic code are output for each data point.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
datatosend	mandatory	ARRAY OF STRINGS	List of data points to be requested; data points must support the getdata service ("datatosend":["url1", "url2",..., "urlx"])

Response ("data" field):

Parameter	Mandatory field	Data type	Description
url	mandatory	STRING	Data point that was queried
code	mandatory	INT	Diagnostic code of the request
data	mandatory	STRING	Value of the data point
timestamp	mandatory	STRING	Timestamp of the response; Value as INT64 number in UTC format (milliseconds from 01.01.1970)

11.1.3.12 Service: getdevicelist

Name: getdevicelist

Description: The service outputs a list of external devices that are stored in the IoT Core of the device.

Request:

Parameter	Mandatory field	Data type	Description
adr	Optional	STRING	Address of the added device; if empty, then all added devices are output

Response:

Parameter	Mandatory field	Data type	Description
--	Optional	DeviceInfo_Type[]	List of devices and their properties

- DeviceInfo_Type:

Parameter	Mandatory field	Data type	Description
adr	mandatory	STRING	Address of the added device
uri	mandatory	STRING	URI of the added device
type	mandatory	STRING	Device type of the added device
user	Optional	STRING	Parameter user of the Auth object of the added device
persist	Optional	BOOLEAN	• false: Save the external device only until the next restart of the device • true: Save the external device even after the next restart of the device
callbackurl	Optional	STRING	URL for notifications

11.1.3.13 Service: getidentity

Name: getidentity

Description: The service retrieves information about the identity of the device.

Request ("data" field): none

Response ("data" field):

Parameter	Mandatory field	Data type	Description
iot	mandatory	device	Device description as JSON object
iot.name	mandatory	STRING	Name of the device (root node)
iot.uid	Optional	STRING	Identifier
iot.version	mandatory	STRING	Version number of the ifm IoT Core

Parameter	Mandatory field	Data type	Description
iot.catalogue	Optional	ARRAY OF STRING	Name and version of the catalogue, e.g. {"name": "ifm-std", "version": "V2.0"}
iot.component	Optional	ARRAY OF OBJECTS	List of software components of the ifm IoT Core For every object: Name and version of the software component, e.g. {"name": "ifm.IoTCore.Application", "version": "1.0.1.4"}
iot.namespace	Optional	STRING	Namespace (e.g. JSON)
iot.deviceclass	Optional	ARRAY OF STRING	Device class (e.g. device)
iot.serverlist	Optional	ARRAY OF OBJECTS	List of server services of the ifm IoT Core For every object: Type, address and format of the server service, e.g. {"type": "http", "uri": "http://0.0.0.0/", "formats": ["json"]}
device	Optional		Article number
device.serialnumber	Optional	STRING	Serial number
device.hwrevision	Optional	STRING	Hardware version
device.swrevision	Optional	STRING	Software version
device.custom	Optional	STRING	
security	Optional		Security options
security.securitymode	Optional	ENUM	Status of the security mode - disabled: security mode disabled - enabled: security mode enabled
security.authscheme	Optional	ENUM	Active authentication scheme (e.g. standard)
security.ispasswdset	Optional	BOOL	shows whether the password is set for the interface used to call the service - false: password not set - true: password set
security.activeconnection	Optional	ENUM	Connection type currently in use - tcp_if: unencrypted http connection at the IoT interface, port 80 - tls_if: encrypted https connection at the IoT interface, port 443 - fb_if: unencrypted http connection at the fieldbus interface, port 80

11.1.3.14 Service: getitems

Name: getitems

Description: The service outputs the data points that are currently being monitored.

Request ("data" field): none

Request ("data" field):

Parameter	Mandatory field	Data type	Description
--	mandatory	GetItems_Type[]	List of active monitoring; The properties are output for each monitoring

- GetItems_Type:

Parameter	Mandatory field	Data type	Description
recipient	mandatory	STRING	URI to which the notifications are sent
data_points	mandatory	STRING[]	List with n addresses of the monitored elements (type: data)
id	mandatory	NUMBER	Monitoring identifier
sample_interval	mandatory	NUMBER	Sampling interval (value in ms)

Parameter	Mandatory field	Data type	Description
sample_trigger	Optional	SampleTrigger_Type	Address of a data point for event-controlled sampling; Data point is read with the sampling interval; When a value is changed, all values of the monitored elements are read
notify_interval	mandatory	NUMBER	Notification interval (value in ms)
data_changed	mandatory	BOOLEAN	- false: Data values are always saved - true: Data values are only saved when a change is made
persist	mandatory	BOOLEAN	- false: Monitoring is no longer active after restarting the device - true: Monitoring is active even after restarting the device

11.1.3.15 Service: getsubscriberlist

Name: getsubscriberlist

Description: The service provides a list of all active subscriptions.

Request ("data" field): none

Response ("data" field):

Parameter	Mandatory field	Data type	Description
adr	mandatory	STRING	Notification trigger
datatosend	mandatory	ARRAY OF STRINGS	List with URLs of the data elements; URLs are comma separated
cid	mandatory	STRING	ID of the active notification
callbackurl	mandatory	STRING	Destination address for the notifications
duration	mandatory	STRING	Activity duration

11.1.3.16 Service: gettree

Name: gettree

Description: The service reads the device description of the unit and outputs it as a JSON object. The output can be limited to a subtree of the device description.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
adr	Optional	STRING	Root element of the subtree
level	Optional	STRING	Max. level up to which the subtree is output - no entry: all levels will be displayed 0: do not display sub-elements ("subs") 1: display sub-elements 2: display sub-elements up to the 2nd level 3: display sub-elements up to the 3rd level ... 20: display sub-elements up to the 20th level

Response ("data" field):

Parameter	Mandatory field	Data type	Description
identifier	mandatory	STRING	Identifier of the root element
type	mandatory	STRING	Type of the element
format	Optional	JSON object	Format of the data content
uid	Optional	STRING	

Parameter	Mandatory field	Data type	Description
profiles	Optional	JSON array	
subs	mandatory	JSON array	Sub-elements
hash	Optional	STRING	

11.1.3.17 Service: gotobootloader

Name: gotobootloader

Description: The service activates the device's recovery mode.

Request ("data" field): none

Response ("data" field): none

11.1.3.18 Service: install

Name: install

Description: The service installs the software stored in a memory area of the unit.

Request ("data" field): none

Response ("data" field): none

11.1.3.19 Service: iolreadacyclic

Name: iolreadacyclic

Description: The service acyclically reads the parameter value of an IO-Link device. It is accessed via IO-Link index and subindex.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
index	mandatory	NUMBER	IO-Link index of the parameter
subindex	mandatory	NUMBER	IO-Link subindex of the parameter

Response ("data" field):

Parameter	Mandatory field	Data type	Description
value	mandatory	STRING	Parameter value (value in hexadecimal format)

11.1.3.20 Service: iolwriteacyclic

Name: iolwriteacyclic

Description: The service acyclically writes the parameter value of an IO-Link device. It is accessed via IO-Link index and subindex.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
index	mandatory	NUMBER	IO-Link index of the parameter
subindex	mandatory	NUMBER	IO-Link subindex of the parameter
value	mandatory	STRING	Parameter value (value in hexadecimal format)

Response ("data" field): none

11.1.3.21 Service: mirror

Name: mirror

Description: The service mirrors an external device in the IoT Core of the device.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
uri	mandatory	STRING	URI of the remote device
callbackurl	Optional	STRING	URL for forwarding notifications from the external device to the device
alias	Optional	STRING	Name of the external device under /remote
auth	Optional	AuthInfo_Type	Authentication data for outgoing requests to the external device
cache_timeout	Optional	NUMBER	Time in which the data values are buffered by the external device in order to reduce the network bandwidth (value in ms)
persist	Optional	BOOLEAN	- false: Mirror until the next start of the device - true: permanently mirror

Response ("data" field): none

11.1.3.22 Service: power_cycle

Name: power_cycle

Description: The service switches the port's supply voltage off and on again after a defined switch-off time.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
off_duration	mandatory	NUMBER	Switch-off time (value in ms) Value range: 500...65535

Response ("data" field): none

11.1.3.23 Service: power_off

Name: power_off

Description: The service switches off the supply voltage US of the port.

Request ("data" field): none

Response ("data" field): none

11.1.3.24 Service: power_on

Name: power_on

Description: The service switches on the supply voltage US of the port.

Request ("data" field): none

Response ("data" field): none

11.1.3.25 Service: querytree

Name: querytree

Description: The service searches a device tree for the criteria profile, type and identifier and outputs a list with the URLs of the elements found. At least one of the search criteria must be specified. The service can only be executed on the root node of the unit.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
profile	Optional	STRING	Profile of the requested element
type	Optional	STRING	Type of the requested element
identifier	Optional	STRING	Name of the requested element

Response ("data" field): none

Parameter	Mandatory field	Data type	Description
urllist	mandatory	ARRAY	Array with URLs of the found elements; URLs are separated by commas

11.1.3.26 Service: reboot

Name: reboot

Description: The service reboots the device.

Request ("data" field): none

Response ("data" field): none

11.1.3.27 Service: remove

Name: remove

Description: The service deletes active monitoring.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
id	mandatory	STRING	Monitoring identifier

Response ("data" field): none

11.1.3.28 Service: removedevice

Name: removedevice

Description: The service deletes an external device from the IoT Core of the device.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
adr	mandatory	STRING	Address of the added device

Response ("data" field): none

11.1.3.29 Service: reset

Name: reset

Description: The service resets a connection to the initialisation state.

Request ("data" field): none

Response ("data" field): none

11.1.3.30 Service: setblock

Name: setblock

Description: The service simultaneously sets the values of multiple data points of a structure.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
datatset	mandatory	ARRAY OF OBJECTS	List of data points and their new values; Data points must support the setdata service
consistent	Optional	BOOL	Response within a certain time

Response ("data" field): none

11.1.3.31 Service: setdata

Name: setdata

Description: The service sets the value of a data point.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
newvalue	mandatory	STRING	New value of the data point
duration	Optional	STRING	Duration of value storage - lifetime: value is saved with IoT Core; value remains valid even after restart of the device - uptime: value is saved until the next restart of the device

Response ("data" field): none

11.1.3.32 Service: setdatamulti

Name: setdatamulti

Description: The service sequentially writes the values of multiple data points. For each data point, the target value must be provided.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
datatosend	mandatory	ARRAY OF STRINGS	List of data points and associated values (coded) to be written; data points must support the setdata service ("datatosend":["url1":"value1","url2":"value2",...,"urlx":"valuex"])

Response ("data" field): none

11.1.3.33 Service: signal

Name: signal

Description: The service triggers the flashing of the status LEDs of the unit.

Request ("data" field): none

Response ("data" field): none

11.1.3.34 Service: start

Name: start

Description: The service starts a function (e.g. connection, timer).

Request ("data" field): none

Response ("data" field): none

11.1.3.35 Service: start_stream_get

Name: start_stream_get

Description: The service receives a Binary Large Object (BLOB) as a streamed transmission.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
size	Optional	NUMBER	Length of the BLOB to be received

Response ("data" field): none

11.1.3.36 Service: start_stream_set

Name: start_stream_set

Description: The service starts the sequential transmission of several data fragments.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
size	mandatory	STRING	Overall length of the data to be transmitted (number of bytes)

Response ("data" field): none

11.1.3.37 Service: stop

Name: stop

Description: The service stops a function (e.g. connection, timer).

Request ("data" field): none

Response ("data" field): none

11.1.3.38 Service: stream_get

Name: stream_get

Description: The service receives a part of a Binary Large Object (BLOB).

Request ("data" field):

Parameter	Mandatory field	Data type	Description
length	Optional	STRING	Max. length of a BLOB part

Response ("data" field):

Parameter	Mandatory field	Data type	Description
value	mandatory	BINARY	Part of the data stream
last	complementary	BOOLEAN	- false: not the last part of the data stream - true: Last part of the data stream

11.1.3.39 Service: stream_set

Name: stream_set

Description: The service transfers a data segment.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
value	mandatory	BIN (BASE64)	Segment of the binary data (BASE64-encoded)

Response ("data" field): none

11.1.3.40 Service: subscribe

Name: subscribe

Description: The service generates a notification order.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
callback	mandatory	STRING	Destination address for notifications; URL formats: - JSON: http://ipaddress:port/path - JSON: https://ipaddress:port/path - JSON: ws:///path - MQTT: mqtt://ipaddress:port/topic - MQTTS: mqtts://ipaddress:port/topic - CSV: tcp://ipaddress:port/path
datatosend	mandatory	ARRAY OF STRINGS	List with URLs of the data elements that are to be transferred; URLs are comma-separated; Elements must support service getdata
codec	Optional	STRING	Format of the returned data - json: JSON formatted - csv: CSV with standard separator (,) - csv0: CSV formatted with comma separator (,) - csv1: CSV formatted with semicolon separator (;)
persist	Optional	BOOL	Duration of the notification activity - true: login remains permanently active, also after restarting the unit - false: login is active until the next restart of the device
subscribeid	Optional	INT	ID of the notification order -1: IoT Core selects ID independently 1...4 294 967 295

Response ("data" field): none

Parameter	Mandatory field	Data type	Description
subscribeid	Optional	INT	ID of the notification order 1...4 294 967 295

Notification format: JSON

```
{
  "code": "event",
  "cid": 4711,
  "adr": "/eventhandler",
  "data": {
    "eventno": "EventNo",
    "srcurl": "SrcURL",
    "payload": {
      "eventurl": { "code": EventStatus, "data": EventData },
      "datapointurl_1":
        { "code": DataStatus_1, "data": DataValue_1, "timestamp": TimeStamp_1 },
      "datapointurl_2":
        { "code": DataStatus_2, "data": DataValue_2, "timestamp": TimeStamp_2 }, ...
    }
  }
}
```

Notification format: CSV

SrcURL, EventNo, EventStatus, EventData, DataStatus_1, DataValue_1, TimeStamp_1, DataStatus_2, DataValue_2, TimeStamp_2, ...

- SrcURL: Source of the event (data point on which the subscribe service was listed)
- EventNo: Event number
- EventStatus: Status code of the event
- EventData: Event data
- DataStatus_1: Status code of the 1st element in datatosend list
- DataValue_1: Value of the 1st element in the datatosend list
- TimeStamp_1: Timestamp of the 1st element in the datatosend list
- DataStatus_2: Status code of the 2nd element in datatosend list
- DataValue_2: Value of the 2nd element in the datatosend list
- TimeStamp_2: Timestamp of the 1st element in the datatosend list

11.1.3.41 Service: supported_devicetypes

Name: supported_devicetypes

Description: The service displays the supported device types.

Request ("data" field): none

Response ("data" field):

Parameter	Mandatory field	Data type	Description
--	mandatory	Supported_Device_Type[]	List of supported device types

- Supported_Devices_Type:

Parameter	Mandatory field	Data type	Description
type	mandatory	STRING	List of supported device types

11.1.3.42 Service: swinfo

Name: swinfo

Description: The service provides status information on the software components of the device.

Request ("data" field): none

Response ("data" field):

Parameter	Mandatory field	Data type	Description
--	mandatory	[]	List of software components

Parameter	Mandatory field	Data type	Description
name	mandatory	STRING	Name of the software subsystem E.g. "Recovery system"
version	mandatory	STRING	Version of the software subsystem • e.g. "1.2.0.1"
active	mandatory	BOOLEAN	• false: Software component inactive • true: Software component active

11.1.3.43 Service: unmirror

Name: unmirror

Description: The service removes an external device from the IoT Core of the device. All corresponding subscriptions will be cancelled. The /treechanged service is finally executed.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
uri	complementary	STRING	URI of the external device
alias	complementary	STRING	Designation of the external device

Response ("data" field): none

11.1.3.44 Service: unsubscribe

Name: unsubscribe

Description: The service deletes an active notification request. The values transferred for cid and callback must be the same as when the notification order (Service: subscribe) was created. If the value DELETE is transferred as callback, all active notifications are deleted.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
callback	mandatory	STRING	Destination address for notifications; URL formats: • JSON: http://ipaddress:port/path • JSON: ws://path • MQTT: mqtt://ipaddress:port/topic • MQTTS: mqttps://ipaddress:port/topic • CSV: tcp://ipaddress:port/path • DELETE: all active notifications are deleted

Response ("data" field): none

11.1.3.45 Service: upload_certificate

Name: upload_certificate

Description: The service loads a certificate and a private key onto the device.

Request ("data" field):

Parameter	Mandatory field	Data type	Description
private_key_value	mandatory	STRING (Base64)	Private key
private_key_crc32	mandatory	STRING (UINT32)	CRC32 checksum of the private key
certificate_value	mandatory	STRING (Base64)	Certificate
certificate_crc32	mandatory	STRING (UINT32)	CRC32 checksum of the certificate

Response ("data" field): none

11.1.3.46 Service: validation_useconnecteddevice

Name: validation_useconnecteddevice

Description: The service checks whether the device ID and the vendor ID of the connected IO-Link device match the data points /validation_vendorid and /validation_deviceid.

Request ("data" field): none

Response ("data" field): none

11.2 EtherNet/IP

11.2.1 Parameters

11.2.1.1 Configuration assembly (instance 199)

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
0	Port Process Data Swap							
1	Port Process Data Size							
2...13	Port X1: Port Configuration							
14...25	Port X2: Port Configuration							
26...37	Port X3: Port Configuration							
38...49	Port X4: Port Configuration							
50...61	Port X5: Port Configuration							
62...73	Port X6: Port Configuration							
74...85	Port X7: Port Configuration							
86...97	Port X8: Port Configuration							
98	Port X1: Mode I/Q (UA)							
99	Port X2: Mode I/Q (UA)							
100	Port X3: Mode I/Q (UA)							
101	Port X4: Mode I/Q (UA)							
102	Port X1: Fail Safe Mode DO I/Q (UA)							
103	Port X2: Fail Safe Mode DO I/Q (UA)							
104	Port X3: Fail Safe Mode DO I/Q (UA)							
105	Port X4: Fail Safe Mode DO I/Q (UA)							

Legend:

- Port Process Data Swap**
 Exchange arrangement of bytes in IO-Link process data **NOTE! EtherNet/IP uses Little Endian format (Intel), IO-Link uses Big Endian format (Motorola)**
 UINT8 / rw
 - 0x00: Enabled
 - 0x01: Disabled (default)
- Port Process Data Size**
 Length of the IO-Link process data (input data and output data)
 UINT8 / rw
 - 0x00: 2 bytes
 - 0x01: 4 byte
 - 0x02: 8 byte
 - 0x03: 16 byte
 - 0x04: 32 Bytes (Default)
- Port Configuration**
 Configuration of the port (pin 4 - C/Q)
 [Mapping: Port Configuration \(→ 177\)](#)
- Mode I/Q (UA)**
 Operating mode pin 2 (I/Q) of the port
 UINT8 / rw
 - 0x00: Off (Supply Type A)
 - 0x01: On (Supply Type B) (Default)
 - 0x02: Digital Output (DO)
- Fail Safe Mode DO I/Q (UA)**
 Fallback value for output data of the port if the EtherNet/IP connection is interrupted (only valid for Mode I/Q (UA): Digital Output (DO))
 UINT8 / rw
 - 0x00: Failsafe Reset Value
 - 0x01: Failsafe Old Value (Default)
 - 0x02: Failsafe Set Value

Mapping: Port Configuration

Byte (Offset)	Contents
n	Port Mode
n+1	Port Cycle Time
n+2	reserved
n+3	Validation / Data Storage
n+4	Vendor ID (LSB)
n+5	Vendor ID (MSB)
n+6	Device ID (LSB)
n+7	Device ID
n+8	Device ID (MSB)
n+9	reserved
n+10	Failsafe Mode IO-Link (C/Q)
n+11	Failsafe Mode DO (C/Q)

Legend:

- Port Mode

Operating mode pin 4 (C/Q) of the port

UINT8 / rw

- 0x00: Disabled
 - 0x01: Digital input (C/Q)
 - 0x02: Digital output (C/Q)
 - 0x03: IO-Link (C/Q) (default)
- Port Cycle Time

Cycle time of the data transmission between IO-Link master and IO-Link device

UINT8 / rw

- 0x00: As fast as possible (default)
 - 0x01: 2 ms
 - 0x02: 4 ms
 - 0x03: 8 ms
 - 0x04: 16 ms
 - 0x05: 32 ms
 - 0x06: 64 ms
 - 0x07: 128 ms
- Validation / Data Storage

Validation mode / data storage mode

UINT8 / rw

- 0x00: No device check and clear (default)
 - 0x01: Type compatible V1.0 device
 - 0x02: Type compatible V1.1 device
 - 0x03: Type compatible V1.1 device with Backup + Restore
 - 0x04: Type compatible V1.1 device with Restore
- Vendor ID

Vendor ID of the manufacturer of the IO-Link device

Vendor ID = 0x1234

 - Vendor ID (MSB): 0x12
 - Vendor ID (LSB): 0x34

UINT16 / rw

- 0x0000 (default)
 - ...
 - 0x7FFF
- Device ID

Device ID of the IO-Link device

Device ID = 0x123456

 - Device ID (MSB): 0x12
 - Device ID 0x34
 - Device ID (LSB): 0x56

UINT32 / rw

- 0x000000 (default)
 - ...
 - 0xFFFFFFFF
- Failsafe Mode IO-Link (C/Q)

Fail-safe mode for output data of the port if the EtherNet/IP connection is interrupted (only valid for port mode: IO-Link)

UINT8 / rw

- 0x00: No Failsafe (Default)
 - 0x01: Failsafe Reset Value
 - 0x02: Failsafe Old Value
 - 0x03: Failsafe with Pattern
- Failsafe Mode DO (C/Q)

Fail-safe value for output data of the port if the EtherNet/IP connection is interrupted (only valid for port mode: Digital Output)

UINT8 / rw

- 0x00: Failsafe Reset Value
 - 0x01: Failsafe Old Value (Default)
 - 0x02: Failsafe Set Value

11.2.2 Cyclic data

11.2.2.1 Input assembly (instance 100)

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
0...1	Digital Input							
2...3	Status Information							
4...5	Port X1: Diagnostic information							
6...7	Port X2: Diagnostic information							
8...9	Port X3: Diagnostic information							
10...11	Port X4: Diagnostic information							
12...13	Port X5: Diagnostic information							
14...15	Port X6: Diagnostic information							
16...17	Port X7: Diagnostic information							
18...19	Port X8: Diagnostic information							
20	Port X1: Input Data IO-Link (n Bytes) ¹							
20+n	Port X2: Input Data IO-Link (n Bytes) ¹							
20+2n	Port X3: Input Data IO-Link (n Bytes) ¹							
20+3n	Port X4: Input Data IO-Link (n Bytes) ¹							
20+4n	Port X5: Input Data IO-Link (n Bytes) ¹							
20+5n	Port X6: Input Data IO-Link (n Bytes) ¹							
20+6n	Port X7: Input Data IO-Link (n Bytes) ¹							
20+7n	Port X8: Input Data IO-Link (n Bytes) ¹							

¹ n: {2, 4, 8, 16, 32}; n is determined by the parameter [Port Process Data Size] in the configuration assembly

Legend:

- Digital Input Digital input data of the ports [Mapping: Digital Input \(→ □ 179\)](#)
- Status Information Status information of the device [Mapping: Status Information \(→ □ 180\)](#)
- Diagnosis Information Diagnostic information of the port [Mapping: Diagnostic information \(→ □ 181\)](#)
- Input Data IO-Link IO-Link input data (n bytes) Per byte:
 - 0x00...0xFF

Mapping: Digital Input

Byte (Off-set)	Bit							
	7	6	5	4	3	2	1	0
n	Port X8: C/Q	Port X7: C/Q	Port X6: C/Q	Port X5: C/Q	Port X4: C/Q	Port X3: C/Q	Port X2: C/Q	Port X1: C/Q
n+1	Port X8: I/Q	Port X7: I/Q	Port X6: I/Q	Port X5: I/Q	reserved	reserved	reserved	reserved

Legend:

- C/Q Signal status of the digital input on 1 BIT • 0x0: LOW
 Pin 4 of the port • 0x1: HIGH
- I/Q Signal status of the digital input on 1 BIT • 0x0: LOW
 Pin 2 of the port • 0x1: HIGH

Mapping: Status Information

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port X8: SC / OL	Port X7: SC / OL	Port X6: SC / OL	Port X5: SC / OL	Port X4: SC / OL	Port X3: SC / OL	Port X2: SC / OL	Port X1: SC / OL
n+1	UA Under Voltage Warning	UA Over Voltage Warning	UA Fault	US Under Voltage Warning	US Over Voltage Warning	US Fault	AUX PWR	SENS PWR

Legend:

- C/Q SC / OL Short circuit / overload display on C/Q (pin 4) of the port 1 BIT
 - Short circuit L+ and C/Q
 - Short circuit L- and C/Q
 - Overload on C/Q
- SENS PWR Status of the supply voltage US 1 BIT
 - 0x0: No error
 - 0x1: Short circuit / overload on pin 4 (US)
- AUX PWR Status supply voltage UA 1 BIT
 - 0x0: No warning (operating range US)
 - 0x1: Warning to US (undervoltage, overvoltage, error)
- US Fault Supply voltage US error indication 1 BIT
 - 0x0: No warning (operating range UA)
 - 0x1: Overcurrent, no voltage applied
- US Overvoltage Warning Overvoltage warning on US 1 BIT
 - 0x0: No error
 - 0x1: Fault (overcurrent on US)
- US Undervoltage Warning Undervoltage warning on US 1 BIT
 - 0x0: No error
 - 0x1: Overvoltage on US
- UA Fault Supply voltage UA error indication 1 BIT
 - 0x0: No error
 - 0x1: Undervoltage on US
- UA Overvoltage Warning Overvoltage warning on UA 1 BIT
 - 0x0: No error
 - 0x1: Fault (overcurrent at UA, no voltage applied)
- UA Undervoltage Warning Undervoltage warning on UA 1 BIT
 - 0x0: No error
 - 0x1: Overvoltage at UA
- UA Undervoltage Warning Undervoltage warning on UA 1 BIT
 - 0x0: No error
 - 0x1: Undervoltage at UA

Mapping: Diagnostic information

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Diagnosis present	Wrong PD Output Len	Wrong PD Input Len	Wrong cycle Time	Wrong VID/DID	Data Invalid	Device Connected	IOL Mode
n+1	PQ	DevErr	DevCom	reserved	reserved	reserved	SENS PWR	reserved

Legend:

- IOL Mode Operating mode status Pin 4 (US) 1 BIT
 - 0x0: disabled, DI, DO
 - 0x1: IO-Link
- Device Connected Connection status to IO-Link device on the port 1 BIT
 - 0x0: No IO-Link device connected
 - 0x1: IO-Link device connected:
- Data Invalid Status of the process data 1 BIT
 - 0x0: valid
 - 0x1: invalid
- Wrong VID/DID Match between configured and recognised Vendor ID / Device ID 1 BIT
 - 0x0: OK
 - 0x1: no match
- Wrong cycle Time Conformity of configured and detected cycle time 1 BIT
 - 0x0: OK
 - 0x1: no match
- Wrong PD Input Len Match between configured and detected process data length (inputs) 1 BIT
 - 0x0: OK
 - 0x1: configured process data length too small
- Wrong PD Output Len Match between configured and detected process data length (outputs) 1 BIT
 - 0x0: OK
 - 0x1: configured process data length too small
- Diagnosis present Indication of new IO-Link events (Coming Event, Single Shot Event) 1 BIT
 - 0x0: no IO-Link event
 - 0x1: new IO-Link event detected
 - Coming Events disappears when the corresponding Disappearing Event occurs
 - Single Shot Events disappear automatically
- SENS PWR Sensor Power: Status supply voltage US of the port (pin 1 and pin 3) 1 BIT
 - 0x0: No error
 - 0x1: Short circuit
- PQ Port qualifier Validity display of the process data of the connected device 1 BIT
 - 0x0: No error
 - 0x1:
- DevErr Device Error: Error display for port or connected device 1 BIT
 - 0x0: no fault / no warning
 - 0x1: Error / warning occurred
- DevCom Device Communication: 1 BIT
 - 0x0: No device available
 - 0x1: device detected Device is in PREOPERATE or OPERATE state

11.2.2.2 Input Assembly (Instance 101)

Byte (Off-set)	Bit							
	7	6	5	4	3	2	1	0
0...1	Digital Input							
2...3	Status Information							
4...45	Acyclic Command Channel: Response							
46...63	Port X1: Diagnostic information + IO-Link events							
64...81	Port X2: Diagnostic information + IO-Link events							
82...99	Port X3: Diagnostic information + IO-Link events							
100...117	Port X4: Diagnostic information + IO-Link events							
118...135	Port X5: Diagnostic information + IO-Link events							
136...153	Port X6: Diagnostic information + IO-Link events							
154...171	Port X7: Diagnostic information + IO-Link events							
172...189	Port X8: Diagnostic information + IO-Link events							
190	Port X1: Input Data IO-Link (n Bytes) ¹							
190+n	Port X2: Input Data IO-Link (n Bytes) ¹							
190+2n	Port X3: Input Data IO-Link (n Bytes) ¹							
190+3n	Port X4: Input Data IO-Link (n Bytes) ¹							
190+4n	Port X5: Input Data IO-Link (n Bytes) ¹							
190+5n	Port X6: Input Data IO-Link (n Bytes) ¹							
190+6n	Port X7: Input Data IO-Link (n Bytes) ¹							
190+7n	Port X8: Input Data IO-Link (n Bytes) ¹							
190+8n...190+8n+3	Energy Monitoring							

¹ n: {2, 4, 8, 16, 32}; n is determined by the parameter [Port Process Data Size] in the configuration assembly

Legend:

- Digital Input Digital input data of the ports [Mapping: Digital Input \(→ □ 183\)](#)
- Status Information Status information of the device [Mapping: Status Information \(→ □ 184\)](#)
- Acyclic Command Channel: Response Acyclic command channel - answer [Response channel \(→ □ 194\)](#)
- Diagnosis Information + IO-Link Events Diagnostic information and IO-Link events of the port [Mapping: Diagnostic information + IO-Link events \(→ □ 185\)](#)
- Input Data IO-Link IO-Link input data (n bytes) Per byte:
 - 0x00...0xFF
- Energy Monitoring Energy monitoring [Mapping: Energy monitoring \(→ □ 187\)](#)

Mapping: Digital Input

Byte (Off-set)	Bit							
	7	6	5	4	3	2	1	0
n	Port X8: C/Q	Port X7: C/Q	Port X6: C/Q	Port X5: C/Q	Port X4: C/Q	Port X3: C/Q	Port X2: C/Q	Port X1: C/Q
n+1	Port X8: I/Q	Port X7: I/Q	Port X6: I/Q	Port X5: I/Q	reserved	reserved	reserved	reserved

Legend:

- C/Q Signal status of the digital input on 1 BIT • 0x0: LOW
 Pin 4 of the port • 0x1: HIGH
- I/Q Signal status of the digital input on 1 BIT • 0x0: LOW
 Pin 2 of the port • 0x1: HIGH

Mapping: Status Information

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port X8: SC / OL	Port X7: SC / OL	Port X6: SC / OL	Port X5: SC / OL	Port X4: SC / OL	Port X3: SC / OL	Port X2: SC / OL	Port X1: SC / OL
n+1	UA Under Voltage Warning	UA Over Voltage Warning	UA Fault	US Under Voltage Warning	US Over Voltage Warning	US Fault	AUX PWR	SENS PWR

Legend:

- C/Q SC / OL Short circuit / overload display on C/Q (pin 4) of the port 1 BIT
 - Short circuit L+ and C/Q
 - Short circuit L- and C/Q
 - Overload on C/Q
- SENS PWR Status of the supply voltage US 1 BIT
 - 0x0: No error
 - 0x1: Short circuit / overload on pin 4 (US)
- AUX PWR Status supply voltage UA 1 BIT
 - 0x0: No warning (operating range US)
 - 0x1: Warning to US (undervoltage, overvoltage, error)
- US Fault Supply voltage US error indication 1 BIT
 - 0x0: No warning (operating range UA)
 - 0x1: Overcurrent, no voltage applied
- US Overvoltage Warning Overvoltage warning on US 1 BIT
 - 0x0: No error
 - 0x1: Fault (overcurrent on US)
- US Undervoltage Warning Undervoltage warning on US 1 BIT
 - 0x0: No error
 - 0x1: Overvoltage on US
- UA Fault Supply voltage UA error indication 1 BIT
 - 0x0: No error
 - 0x1: Undervoltage on US
- UA Overvoltage Warning Overvoltage warning on UA 1 BIT
 - 0x0: No error
 - 0x1: Fault (overcurrent at UA, no voltage applied)
- UA Undervoltage Warning Undervoltage warning on UA 1 BIT
 - 0x0: No error
 - 0x1: Overvoltage at UA
- UA Undervoltage Warning Undervoltage warning on UA 1 BIT
 - 0x0: No error
 - 0x1: Undervoltage at UA

Mapping: Diagnostic information + IO-Link events

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Diagnosis present	Wrong PD Output Len	Wrong PD Input Len	Wrong cycle Time	Wrong VID/DID	Data Invalid	Device Connected	IOL Mode
n+1	PQ	DevErr	DevCom	reserved	reserved	reserved	SENS PWR	reserved
n+2	Vendor ID (LSB)							
n+3	Vendor ID (MSB)							
n+4	Device ID (LSB)							
n+5	Device ID							
n+6	Device ID (MSB)							
n+7	reserved							
n+8	Event 1: Mode		Event 1: Type		Event 1: Src	Event 1: Instance		
n+9	Event Code 1 (LSB)							
n+10	Event Code 1 (MSB)							
n+11	Event 2: Mode		Event 2: Type		Event 2: Src	Event 2: Instance		
n+12	Event 2 (LSB)							
n+13	Event 2 (MSB)							
n+14	Event 3: Mode		Event 3: Type		Event 3: Src	Event 3: Instance		
n+15	Event 3 (LSB)							
n+16	Event 3 (MSB)							
n+17	reserved							

Legend:

- IOL Mode Operating mode status Pin 4 (US) 1 BIT
 - 0x0: disabled, DI, DO
 - 0x1: IO-Link
- Device Connected Connection status to IO-Link device on the port 1 BIT
 - 0x0: No IO-Link device connected
 - 0x1: IO-Link device connected:
- Data Invalid Status of the process data 1 BIT
 - 0x0: valid
 - 0x1: invalid
- Wrong VID/DID Match between configured and recognised Vendor ID / Device ID 1 BIT
 - 0x0: OK
 - 0x1: no match
- Wrong cycle Time Conformity of configured and detected cycle time 1 BIT
 - 0x0: OK
 - 0x1: no match
- Wrong PD Input Len Match between configured and detected process data length (inputs) 1 BIT
 - 0x0: OK
 - 0x1: configured process data length too small
- Wrong PD Output Len Match between configured and detected process data length (outputs) 1 BIT
 - 0x0: OK
 - 0x1: configured process data length too small
- Diagnosis present Indication of new IO-Link events (Coming Event, Single Shot Event) 1 BIT
 - 0x0: no IO-Link event
 - 0x1: new IO-Link event detected
 - Coming Events disappears when the corresponding Disappearing Event occurs
 - Single Shot Events disappear automatically
- SENS PWR Sensor Power: Status supply voltage US of the port (pin 1 and pin 3) 1 BIT
 - 0x0: No error
 - 0x1: Short circuit
- PQ Port qualifier Validity display of the process data of the connected device 1 BIT
 - 0x0: No error
 - 0x1:

• DevErr	Device Error: Error display for port or connected device	1 BIT	• 0x0: no fault / no warning • 0x1: Error / warning occurred
• DevCom	Device Communication:	1 BIT	• 0x0: No device available • 0x1: device detected Device is in PREOPERATE or OPERATE state

Mapping: Energy monitoring

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Supply US - Voltage Level (LSB)							
n+1	Supply US - Voltage Level (MSB)							
n+2	Supply US - Current Level (LSB)							
n+3	Supply US - Current Level (MSB)							

Legend:

- Supply US - Voltage Level**
 Voltage value of the supply voltage US (value in mV)
 Supply US - Voltage Level = 0x1234
 • Supply US - Voltage Level (MSB) = 0x12
 • Supply US - Voltage Level (LSB) = 0x34
 UINT16 / ro
 • 0x4E20: 20000 mV
 ...
 • 0x7530: 30000 mV
- Supply US - Current Level**
 Current intensity value of the supply voltage US (value in mA)
 Supply US - Current Level = 0x1234
 • Supply US - Current Level (MSB) = 0x12
 • Supply US - Current Level (LSB) = 0x34
 UINT16 / ro
 • 0x0000: 0 mA
 ...
 • 0x0E10: 3600 mA

11.2.2.3 Output Assembly (Instance 150)

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
0...1	Digital Output							
2	Port X1: Output Data IO-Link (n Bytes) ¹							
2+n	Port X2: Output Data IO-Link (n Bytes) ¹							
2+2n	Port X3: Output Data IO-Link (n Bytes) ¹							
2+3n	Port X4: Output Data IO-Link (n Bytes) ¹							
2+4n	Port X5: Output Data IO-Link (n Bytes) ¹							
2+5n	Port X6: Output Data IO-Link (n Bytes) ¹							
2+6n	Port X7: Output Data IO-Link (n Bytes) ¹							
2+7n	Port X8: Output Data IO-Link (n Bytes) ¹							

¹ n: {2, 4, 8, 16, 32}; n is determined by the parameter [Port Process Data Size] in the configuration assembly

Legend:

- Digital Output Digital output data of the ports
- Output Data IO-Link IO-Link output data (n bytes)

[Mapping: Digital output \(→ 189\)](#)

Per byte:

- 0x00...0xFF

Mapping: Digital output

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port X8: C/Q	Port X7: C/Q	Port X6: C/Q	Port X5: C/Q	Port X4: C/Q	Port X3: C/Q	Port X2: C/Q	Port X1: C/Q
n+1	reserved	reserved	reserved	reserved	Port X4: I/Q	Port X3: I/Q	Port X2: I/Q	Port X1: I/Q

Legend:

- | | | | |
|-------|---|-------|---|
| • C/Q | Digital output signal status on pin 4 of the port | 1 BIT | <ul style="list-style-type: none"> • 0x0: LOW • 0x1: HIGH |
| • I/Q | Digital output signal status on pin 2 of the port | 1 BIT | <ul style="list-style-type: none"> • 0x0: LOW • 0x1: HIGH |

11.2.2.4 Output Assembly (Instance 151)

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
0...1	Digital Output							
2...3	reserved							
4...45	Acyclic Command Channel: Request							
46	Port X1: Output Data IO-Link (n Bytes) ¹							
46+n	Port X2: Output Data IO-Link (n Bytes) ¹							
46+2n	Port X3: Output Data IO-Link (n Bytes) ¹							
46+3n	Port X4: Output Data IO-Link (n Bytes) ¹							
46+4n	Port X5: Output Data IO-Link (n Bytes) ¹							
46+5n	Port X6: Output Data IO-Link (n Bytes) ¹							
46+6n	Port X7: Output Data IO-Link (n Bytes) ¹							
46+7n	Port X8: Output Data IO-Link (n Bytes) ¹							

¹ n: {2, 4, 8, 16, 32}; n is determined by the parameter [Port Process Data Size] in the configuration assembly

Legend:

- Digital Output Digital output data of the ports [Mapping: Digital output \(→ 191\)](#)
- Acyclic Command Channel: Request Acyclic command channel - request rw [Request channel \(→ 193\)](#)
- Output Data IO-Link IO-Link output data (n bytes) Per byte:
 - 0x00...0xFF

Mapping: Digital output

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port X8: C/Q	Port X7: C/Q	Port X6: C/Q	Port X5: C/Q	Port X4: C/Q	Port X3: C/Q	Port X2: C/Q	Port X1: C/Q
n+1	reserved	reserved	reserved	reserved	Port X4: I/Q	Port X3: I/Q	Port X2: I/Q	Port X1: I/Q

Legend:

- | | | | |
|-------|---|-------|---|
| • C/Q | Digital output signal status on pin 4 of the port | 1 BIT | <ul style="list-style-type: none"> • 0x0: LOW • 0x1: HIGH |
| • I/Q | Digital output signal status on pin 2 of the port | 1 BIT | <ul style="list-style-type: none"> • 0x0: LOW • 0x1: HIGH |

11.2.3 Acyclical data

11.2.3.1 Acyclic command channel

In the cyclic process data, command channels for the transmission of acyclic data is available.

Assembly Objects	Contents	Byte	Access
Output Assembly (Instance 151)	Request channel (EtherNet/IP controller → IO-Link master) Request channel (→ □ 193)	4...45	rw ¹
Input Assembly (Instance 101)	Response channel (IO-Link master → EtherNet/IP controller) Response channel (→ □ 194)	4...45	ro ²

¹ Read and write

² Read only

Request channel

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	Index (LSB)							
n+3	Index (MSB)							
n+4	Subindex (LSB)							
n+5	Subindex (MSB)							
n+6	Trigger	Command ID						
n+7	Data Length							
n+8	Data (Byte 0)							
n+9	Data (Byte 1)							
...	...							
n+39	Data (Byte 31)							
n+40	reserved							
n+41	reserved							

Legend:

- Port No. IO-Link Port Unsigned16 • 0x0001: Port 1
 Port No. 0x1234 • 0x0002: Port 2
 • Port No. (MSB): 0x12 • 0x0003: Port 3
 • Port No. (LSB): 0x34 • 0x0004: Port 4
 • 0x0005: Port 5
 • 0x0006: Port 6
 • 0x0007: Port 7
 • 0x0008: Port 8
- Index Index of the ISDU object Unsigned16 0x0000...0xFFFF
 Index: 0x1234
 • Index (MSB): 0x12
 • Index (LSB): 0x34
- Subindex Subindex of the ISDU object Unsigned16 0x0000...0xFFFF
 Subindex 0x1234
 • Subindex (MSB): 0x12
 • Subindex (LSB): 0x34
- Trigger Control of the command execution 1 BIT • 0x0: no action
 • 0x1: Execute command
- Command ID Command number 7 BIT • 0x01: Get
 • 0x02: Write
- Data Length Number of bytes that contain relevant user data Unsigned8 • 0x00: 0 bytes
 ...
 • 0x20: 32 bytes
- Data (Byte n) User data (byte n) Unsigned8 Per byte:
 • 0x00...0xFF

Response channel

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	Index (LSB)							
n+3	Index (MSB)							
n+4	Subindex (LSB)							
n+5	Subindex (MSB)							
n+6	Handshake	Command ID						
n+7	Result							
n+8	Data Length							
n+9	Data (Byte 0) Error Code							
n+10	Data (Byte 1) Additional Code							
	...							
n+40	Data (Byte 31)							
n+41	reserved							

Legend:

• Port No.	IO-Link Port Port No. 0x1234 • Port No. (MSB): 0x12 • Port No. (LSB): 0x34	Unsigned16	• 0x0001: Port 1 • 0x0002: Port 2 • 0x0003: Port 3 • 0x0004: Port 4 • 0x0005: Port 5 • 0x0006: Port 6 • 0x0007: Port 7 • 0x0008: Port 8
• Index	Index of the ISDU object Index: 0x1234 • Index (MSB): 0x12 • Index (LSB): 0x34	Unsigned16	0x0000...0xFFFF
• Subindex	Subindex of the ISDU object Subindex 0x1234 • Subindex (MSB): 0x12 • Subindex (LSB): 0x34	Unsigned16	0x0000...0xFFFF
• Handshake	Validity of the IO-Link response data	1 BIT	• 0x0: data invalid • 0x1: data valid
• Command ID	Command number	7 BIT	• 0x01: Get • 0x02: Write
• Result	Status of the command processing	Unsigned8	• 0x00: OK • 0x0F: OK, data user read > 32 bytes • 0xFF: Error
• Data Length	Number of bytes that contain relevant user data	Unsigned8	• 0x00: 0 bytes ... • 0x20: 32 bytes
• Data (Byte 0) Error Code	User data (byte 0) or error codes	Unsigned8	• User data: 0x00...0xFF • Error Code: Error Code (→ □ 196)
• Data (Byte 1) Additional Code	User data (byte 1) or additional error codes	Unsigned8	• User data: 0x00...0xFF • Additional Code: Additional Code (→ □ 196)

- Data (byte n) User data (byte n) Unsigned8 Per byte:
0x00...0xFF

Error Code

Error code	Description
0x71	Service not available (unknown command has been sent to the IO-Link port)
0x72	Port blocked (another cyclic process accesses the IO-Link port)
0x73	Forbidden (access rights prohibit command execution)
0x74	Invalid data (wrong parameter has been sent in the command)
0x76	Wrong port (wrong port number)
0x77	Wrong port function (wrong port function or wrong parameter has been sent to the device)
0x78	Invalid length (set length is > 0x20)
0x80	Error in the device application Consider additional code (Additional Code (→ □ 196))

Additional Code



Additional Code only valid if Error Code = 0x80 (→ [Error Code □ 196](#)).

Code	Name	Description
0x00	APP_DEV	Error in the device application - no details
0x11	IDX_NOTAVAIL	Index not available
0x12	SUBIDX_NOTAVAIL	Subindex not available
0x20	SERV_NOTAVAIL	Service temporarily unavailable
0x21	SERV_NOTAVAIL_LOCCTRL	Service temporarily unavailable - local
0x22	SERV_NOTAVAIL_DEVCTRL	Service temporarily unavailable - device
0x23	IDX_NOT_WRITEABLE	Access denied
0x30	PAR_VALOUTOFRNG	Parameter value outside the valid range
0x31	PAR_VALGTLM	Parameter value above the threshold
0x32	PAR_VALLTLM	Parameter value below the threshold
0x33	VAL_LENVERRUN	Parameter length exceeded
0x34	VAL_LENUNDRUN	Parameter length not reached
0x35	FUNC_NOTAVAIL	Function not available
0x36	FUNC_UNAVAILTEMP	Function temporarily unavailable
0x40	PAR_SETINVALID	Invalid parameter set
0x41	PAR_SETINCONSIST	Inconsistent parameter set
0x82	APP_DEVNOTRDY	Application not ready

11.2.3.2 Acyclic commands

Available commands:

- [Set Mode \(0x10\)](#) (→ [197](#))
- [Set validation ID / data storage \(0x20\)](#) (→ [199](#))
- [Set fail-safe data pattern \(0x30\)](#) (→ [201](#))
- [Set Port Power Mode \(0x40\)](#) (→ [203](#))

Set Mode (0x10)

The command changes the operating mode of an IO-Link port of the device.

Command request

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Trigger	0x10						
n+7	Target Mode							
n+8	reserved							
...	...							
n+41	reserved							

Legend:

- | | | | |
|---------------|--|------------|--|
| • Port No. | IO-Link Port
Port No. 0x1234 | Unsigned16 | <ul style="list-style-type: none"> • 0x0001: Port 1 • 0x0002: Port 2 • 0x0003: Port 3 • 0x0004: Port 4 • 0x0005: Port 5 • 0x0006: Port 6 • 0x0007: Port 7 • 0x0008: Port 8 |
| | <ul style="list-style-type: none"> • Port No. (MSB): 0x12 • Port No. (LSB): 0x34 | | |
| • Trigger | Control of the command execution | 1 BIT | <ul style="list-style-type: none"> • 0x0: no action • 0x1: Execute command |
| • Target Mode | Operating mode of the IO-Link port | 1 BYTE | <ul style="list-style-type: none"> • 0x00: deactivated • 0x01: Digital input (DI) • 0x02: Digital output (DO) • 0x03: IO-Link |

Command response

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Handshake	0x10						
n+7	Result							
n+8	Target Mode							
n+9	reserved							
...	...							
n+41	reserved							

Legend:

- Port No. IO-Link Port Unsigned16 • 0x0001: Port 1
 Port No. 0x1234 • 0x0002: Port 2
 • Port No. (MSB): 0x12 • 0x0003: Port 3
 • Port No. (LSB): 0x34 • 0x0004: Port 4
 • 0x0005: Port 5
 • 0x0006: Port 6
 • 0x0007: Port 7
 • 0x0008: Port 8
- Handshake Validity of the IO-Link response data 1 BIT • 0x0: data invalid
 • 0x1: data valid
- Result Status of the command processing Unsigned8 • 0x00: OK
 • 0x0F: OK, data user read > 32 bytes
 • 0xFF: Error
- Target Mode Operating mode of the IO-Link port 1 BYTE • 0x00: deactivated
 • 0x01: Digital input (DI)
 • 0x02: Digital output (DO)
 • 0x03: IO-Link

Set validation ID / data storage (0x20)

The command sets the behaviour of the IO-Link master when connecting a new IO-Link device to an IO-Link port of the device.

Command request

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Trigger	0x20						
n+7	Validation ID							
n+8	reserved							
...	...							
n+41	reserved							

Legend:

- Port No. IO-Link Port Unsigned16 • 0x0001: Port 1
 - Port No. 0x1234
 - Port No. (MSB): 0x12
 - Port No. (LSB): 0x34
 - 0x0002: Port 2
 - 0x0003: Port 3
 - 0x0004: Port 4
 - 0x0005: Port 5
 - 0x0006: Port 6
 - 0x0007: Port 7
 - 0x0008: Port 8
- Trigger Control of the command execution 1 BIT • 0x0: no action
 - 0x1: Execute command
- Validation ID Validation type and data storage behaviour of the port 1 BYTE • 0x00: No check
 - 0x01: Type compatible V1.0 device
 - 0x02: Type compatible V1.1 device
 - 0x03: Type compatible V1.1 device with Backup + Restore
 - 0x04: Type compatible V1.1 device with Restore

Command response

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Handshake	0x20						
n+7	Result							
n+8	Validation ID							
n+9	reserved							
...	...							
n+41	reserved							

Legend:

- Port No. IO-Link Port Unsigned16
 - Port No. 0x1234
 - Port No. (MSB): 0x12
 - Port No. (LSB): 0x34
- Handshake Validity of the IO-Link response data 1 BIT
 - 0x0001: Port 1
 - 0x0002: Port 2
 - 0x0003: Port 3
 - 0x0004: Port 4
 - 0x0005: Port 5
 - 0x0006: Port 6
 - 0x0007: Port 7
 - 0x0008: Port 8
 - 0x0: data invalid
 - 0x1: data valid
- Result Status of the command processing Unsigned8
 - 0x00: OK
 - 0x0F: OK, data user read > 32 bytes
 - 0xFF: Error
- Validation ID Validation type and data storage behaviour of the port 1 BYTE
 - 0x00: No check
 - 0x01: Type compatible V1.0 device
 - 0x02: Type compatible V1.1 device
 - 0x03: Type compatible V1.1 device with Backup + Restore
 - 0x04: Type compatible V1.1 device with Restore

Set fail-safe data pattern (0x30)

The command sets the behaviour of the outputs when the EtherNet/IP connection and the corresponding fail-safe values are interrupted.

Command request

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Trigger	0x30						
n+7	Fail-safe mode							
n+8	Data length							
n+9	reserved / fail-safe data (byte 0)							
...	...							
n+40	reserved / fail-safe data (byte 31)							
n+41	reserved							

Legend:

• Port No.	IO-Link Port Port No. 0x1234 • Port No. (MSB): 0x12 • Port No. (LSB): 0x34	Unsigned16	• 0x0001: Port 1 • 0x0002: Port 2 • 0x0003: Port 3 • 0x0004: Port 4 • 0x0005: Port 5 • 0x0006: Port 6 • 0x0007: Port 7 • 0x0008: Port 8
• Trigger	Control of the command execution	1 BIT	• 0x0: no action • 0x1: Execute command
• Fail-safe mode	Fail-safe mode for the outputs of the IO-Link ports in case of an interruption of the EtherNet/IP connection	1 BYTE	• 0x00: No Fail-safe • 0x01: Fail-safe - Reset Value • 0x02: Fail-safe - Old Value • 0x03: Fail-safe - Pattern
• Data Length	Number of the bytes containing fail-safe values	1 BYTE	• 0x02: 0 byte ... • 0x20: 32 bytes
• Fail-safe data (byte n)	Fail-safe value byte n (n: 0...31) Only with fail-safe mode = 0x0003	1 BYTE	Per byte: • 0x00...0xFF

Command response

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Handshake	0x30						
n+7	Result							
n+8	Fail-safe Mode							
n+9	reserved							
...	...							
n+41	reserved							

Legend

• Port No.	Port number	1 WORD	• 0x0001: Port X01 • 0x0002: Port X02 • 0x0003: Port X03 • 0x0004: Port X04 • 0x0005: Port X05 • 0x0006: Port X06 • 0x0007: Port X07 • 0x0008: Port X08
• Handshake	Validity of the IO-Link response data	1 BIT	• 0x0: data invalid • 0x1: data valid
• Result	Status of the command processing	Unsigned8	• 0x00: OK • 0x0F: OK, data user read > 32 bytes • 0xFF: Error
• Fail-safe mode	Fail-safe mode for the outputs of the IO-Link ports in case of an interruption of the EtherNet/IP connection	1 BYTE	• 0x00: No Fail-safe • 0x01: Fail-safe - Reset Value • 0x02: Fail-safe - Old Value • 0x03: Fail-safe - Pattern

Set Port Power Mode (0x40)

The command controls the supply voltage US (pin 1) of the port.

Command request

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Trigger	0x40						
n+7	Port Power Mode							
n+8	Power Off Time (LSB)							
n+9	Power Off Time (MSB)							
...	...							
n+41	reserved							

Legend:

- | | | | |
|-------------------|---|------------|--|
| • Port No. | IO-Link Port
Port No. 0x1234 <ul style="list-style-type: none"> • Port No. (MSB): 0x12 • Port No. (LSB): 0x34 | Unsigned16 | <ul style="list-style-type: none"> • 0x0001: Port 1 • 0x0002: Port 2 • 0x0003: Port 3 • 0x0004: Port 4 • 0x0005: Port 5 • 0x0006: Port 6 • 0x0007: Port 7 • 0x0008: Port 8 |
| • Trigger | Control of the command execution | 1 BIT | <ul style="list-style-type: none"> • 0x0: no action • 0x1: Execute command |
| • Port Power Mode | Behaviour of pin 2 of the port | Unsigned8 | <ul style="list-style-type: none"> • 0x00: One-time switch-off for the set Power Off Time • 0x01: Switch off (OFF) • 0x02: Switch on (ON) |
| • Power Off Time | Duration of switch-off (value in ms)
Power Off Time: 0x01F4 <ul style="list-style-type: none"> • Power Off Time (MSB): 0x01 • Power Off Time (LSB): 0xF4 | Unsigned16 | <ul style="list-style-type: none"> • 0x01F4: 500 ms • ... • 0xFFFF: 65535 ms |

Command response

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n	Port No. (LSB)							
n+1	Port No. (MSB)							
n+2	reserved							
n+3	reserved							
n+4	reserved							
n+5	reserved							
n+6	Handshake	0x40						

Byte (Offset)	Bit							
	7	6	5	4	3	2	1	0
n+7	Result							
n+8	Port Power Mode							
n+9	Power Off Time (LSB)							
n+10	Power Off Time (MSB)							
...	...							
n+41	reserved							

Legend:

- Port No. IO-Link Port Unsigned16 • 0x0001: Port 1
 Port No. 0x1234 • 0x0002: Port 2
 • Port No. (MSB): 0x12 • 0x0003: Port 3
 • Port No. (LSB): 0x34 • 0x0004: Port 4
 • 0x0005: Port 5
 • 0x0006: Port 6
 • 0x0007: Port 7
 • 0x0008: Port 8

- Handshake Validity of the IO-Link response data 1 BIT • 0x0: data invalid
 • 0x1: data valid

- Result Status of the command processing Unsigned8 • 0x00: OK
 • 0x0F: OK, data user read > 32 bytes
 • 0xFF: Error

- Port Power Mode Behaviour of pin 2 of the port Unsigned8 • 0x00: One-time switch-off for the set
 Power Off Time
 • 0x01: Switch off (OFF)
 • 0x02: Switch on (ON)

- Power Off Time Duration of switch-off (value in ms) Unsigned16 • 0x01F4: 500 ms
 Power Off Time: 0x01F4 • ...
 • Power Off Time (MSB): 0x01 • 0xFFFF: 65535 ms
 • Power Off Time (LSB): 0xF4

11.2.4 Object directory

11.2.4.1 CIP object classes

Supported object classes:

Class Code	Name	Description
0x01	Identity Object	Identity Object (Class Code: 0x01) (→ □ 206)
0x02	Message Router Object	Message Router Object (Class Code: 0x02) (→ □ 208)
0x04	Assembly Object	Assembly Object (Class Code: 0x04) (→ □ 209)
0x06	Connection Manager Object	Connection Manager Object (Class Code: 0x06) (→ □ 210)
0x47	Device Level Ring Object	Device Level Ring Object (Class Code: 0x47) (→ □ 211)
0x48	Quality Of Service Object	Quality Of Service Object (Class Code: 0x48) (→ □ 212)
0x80	IO-Link Request Object	IO-Link Request Object (Class Code: 0x80) (→ □ 213)
0xF5	TCP/IP Object	TCP/IP Object (Class Code: 0xF5) (→ □ 219)
0xF6	Ethernet Link Object	Ethernet Link Object (Class Code: 0xF6) (→ □ 221)
0x109	LLDP Management Object	LLDP Management Object (Class Code: 0x109) (→ □ 223)

11.2.4.2 CIP class and instance services

Supported class and instance services:

Service code	Name	Description
0x01	Get Attribute All	Read all attributes
0x02	Set Attribute All	Write all attributes
0x05	Reset	Reset
0x09	Delete	Delete
0x0E	Get Attribute Single	Read single attribute
0x10	Set Attribute Single	Write single attribute
0x4B	Read ISDU	Read ISDU
0x4C	Write ISDU	Write ISDU
0x4D	Write Failsafe Pattern	Write fail-safe values
0x4E	Forward Close	Close connection
0x54	Forward Open	Open new connection

11.2.4.3 Identity Object (Class Code: 0x01)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	1.1
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances of the object	1
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	19

Instance attributes

Attr. ID	Access	Name	Data type	Description	Default
1	Get	Vendor ID	UINT	Manufacturer ID	322
2	Get	Device Type	UINT	Device type	12
3	Get	Product Code	UINT	Product code of the device	1591
4	Get	Revision	STRUCT	Revision of the device	1.1
		• Major Revision	USINT	Major revision (1...127)	1
		• Minor Revision	USINT	Minor revision (3 digits)	001
5	Get	Status	WORD	Overall status of the device	
6	Get	Serial Number	UDINT	Serial number of the device	--
7	Get	Product Name	SHORT STRING	Product name of the device	IO-LINK MASTER SOL MP 8P IP69K
8	Get	State	USINT	State of the device (State machine) • 0: Nonexistent • 1: Device Self Testing • 2: Standby • 3: Operational • 4: Major Recoverable Fault • 5: Major Unrecoverable Fault • 6-254: Reserved • 255: Default for "Get_Attributes_All" service	
19	Get	Protection Mode	UINT	Current protection mode of the device	0

Services

Code	Service	Class	Instance	Description
0x01	Get Attribute All	Yes	Yes	Read all attributes
0x05	Reset	Yes	Yes	Reset
0x4B	Flash LEDs	No	Yes	Flash device LEDs (identification)
0x0E	Get Attribute Single	Yes	Yes	Read single attribute

If an Identity Object receives a reset request, it carries out the following actions:

- It checks if it supports the requested reset type.
- It responds to the request.
- It tries to execute the requested reset type.

Supported reset types:

- 0: reboot the device (mandatory for all EtherNet/IP devices).
- 1: restore factory settings and reboot the device.

11.2.4.4 Message Router Object (Class Code: 0x02)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	1
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances	1
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	0

Instance attributes

The object class has no instance attributes.

Services

Code	Service	Class	Instance	Description
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute

11.2.4.5 Assembly Object (Class Code: 0x04)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	2
2	Get	Max. Instance	UINT	Max. number of instances of the object	0x00C7
3	Get	Number of instances	UINT	Number of instances	3
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of the class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	4

Instance attributes

The following object instances are available:

- [Configuration assembly \(instance 199\) \(→ □ 176\)](#)
- [Input assembly \(instance 100\) \(→ □ 178\)](#)
- [Input Assembly \(Instance 101\) \(→ □ 182\)](#)
- [Output Assembly \(Instance 150\) \(→ □ 188\)](#)
- [Output Assembly \(Instance 151\) \(→ □ 190\)](#)

Services

Code	Service	Class	Instance	Description
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute

11.2.4.6 Connection Manager Object (Class Code: 0x06)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	1
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances	3
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	0

Instance attributes

The object class has no instance attributes.

Services

Code	Service	Class	Instance	Description
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute
0x4E	Forward Close	Yes	No	Close connection
0x54	Forward Open	Yes	No	Open new connection

11.2.4.7 Device Level Ring Object (Class Code: 0x47)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	3
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances of the object	1
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of the class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	12

Instance attributes

Attr. ID	Access	Name	Data type	Description	Default
1	Get	Network topology	USINT	Current network topology • 0: linear	0
2	Get	Network status	USINT	Current network status • 0: OK	0
10	Get	Active Supervisor	STRUCT	Identification of the supervisor	0
			• UDINT	IP address	
			• ARRAY(6) of USINT	MAC address of the active supervisor	
12	Get	Capability Flags	DWORD	DLR capability of the device • 0x82: Beacon based Ring Node, Flush Table Frame support	0x82

Services

Code	Service	Class	Instance	Description
0x01	Get Attribute All	No	Yes	Read all attributes
0x0E	Get Attribute Single	Yes	Yes	Read single attribute

11.2.4.8 Quality Of Service Object (Class Code: 0x48)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	1
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances of the object	1
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	8

Instance attributes

Attr. ID	Access	Name	Data type	Description	Default
1	Get	802.1Q Tag Enable	USINT	Current network topology	0: deactivated
2	Get / Set	DSCP PTP Event	USINT	DSCP value for PTP event frames	59
3	Get / Set	DSCP PTP General	USINT	DSCP value for PTP general frames	47
4	Get / Set	DSCP PTP Urgent	USINT	DSCP value for implicit messages with "urgent" priority	55
5	Get / Set	DSCP Scheduled	USINT	DSCP value for implicit messages with "scheduled" priority	47
6	Get / Set	DSCP High	USINT	DSCP value for implicit messages with "high" priority	43
7	Get / Set	DSCP Low	USINT	DSCP value for implicit messages with "low" priority	31
8	Get / Set	DSCP Explicit	USINT	DSCP value for explicit messages with "explicit" priority	27

Services

Code	Service	Class	Instance	Description
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute

11.2.4.9 IO-Link Request Object (Class Code: 0x80)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	4
2	Get	Max. Instance	UINT	Max. number of instances of the object	2
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	8

Instance attributes

The required IO-Link port of the device is addressed via the instance attribute.

Services

Code	Service	Class	Instance	Description
0x4B	Read ISDU (→ 214)	no	yes	Read ISDU
0x4C	Write ISDU (→ 215)	no	yes	ISDU write
0x4D	Write Failsafe Pattern (→ 216)	no	yes	Write fail-safe values
0x4E	Write Port Power Mode (→ 217)	No	yes	Write port power mode

Read ISDU

With Read_ISDU, parameters of a connected IO-Link device can be read.

Request

CIP Attributes determines the IO-Link port to which the IO-Link device is connected. The area "CIP User Specific Service Data" contains the IO-Link index and the IO-Link sub-index of the IO-Link object whose value is to be read:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4B	"Read ISDU" service
CIP User Specific Service Data	UINT	0x0000...0xFFFF	ISDU object index
	USINT	0x00...0xFF	ISDU object subindex

Response

Fault-free processing

If the service has been executed successfully (CIP Error Code = 0), the read data are returned bit by bit (CIP User Specific Service Data). The response has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4B	"Read ISDU" service
CIP Error Code	USINT	0x00	Fault-free processing
CIP Extended Error Code	USINT	0x00	--
CIP User Specific Service Data	USINT	0x00...0xFF	Data (byte 0)
	USINT	0x00...0xFF	Data (byte 1)
	...	...	...
	USINT	0x00...0xFF	Data (byte n)



The read data is in the IO-Link format. If necessary, the user needs to adapt the byte arrangement of the read data to the CIP format.

Fault

If an error occurs while executing the service (CIP Error Code $\neq$ 0), an extended error code will be transmitted (CIP Extended Error Code). If the CIP Error Code = 0x1E, the CIP Extended Error Code is = 0x00 and the IO-Link Error Code and the IO-Link Additional Code will be transmitted in the CIP User Specific Service Data area. The response has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4B	"Read ISDU" service
CIP Error Code	USINT	$\neq$ 0x00	Error code (see below)
CIP Extended Error Code	USINT	0x00	Extended error code
CIP User Specific Service Data	USINT	$\neq$ 0x00	IO-Link Error Code: Error Code ($\rightarrow$ 196)

CIP Format	Data type	MSG Config	Description
CIP User Specific Service Data	USINT	<> 0x00	IO-Link Additional Code: Additional Code (→ 196)

CIP Error Code:

Code	Description
0x02	Resource not available: The IO-Link port is busy processing another acyclic service.
0x05	Invalid class ID or instance ID
0x08	Wrong service ID: only service 0x4B, 0x4C or 0x4D is permitted
0x09	Wrong attribute ID: wrong port number
0x20	Invalid parameter value (e.g. invalid length)
0x1E	Embedded service error: Error occurred during an IO-Link service. Byte 0 and byte 1 of the CIP User Specific Service Data contain the IO-Link error code and an additional code returned by the IO-Link master.

Write ISDU

With Write_ISDU, the parameters of a connected IO-Link device can be changed.

Request

CIP Attributes determines the IO-Link port to which the IO-Link device is connected. The area "User Specific Service Data" contains the IO-Link index, the IO-Link sub-index of the IO-Link object whose value is to be changed. The value to be assigned to the parameter follows byte by byte.

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4C	"Write ISDU" service
CIP User Specific Service Data	UINT	0x0000...0xFFFF	ISDU object index
	USINT	0x00...0xFF	ISDU object subindex
	USINT	0x00...0xFF	ISDU data (byte 0)
	USINT	0x00...0xFF	ISDU data (byte 1)

Response

Fault-free processing

If the service was executed successfully (CIP Error Code = 0), the area "User Specific Data" will remain empty. The response has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4C	"Write ISDU" service
CIP Error Code	USINT	0x00	Fault-free processing
CIP Extended Error Code	USINT	0x00	--

Fault

If an error occurs while executing the service (CIP Error Code <> 0), an extended error code is transmitted (CIP Extended Error Code). If the CIP Error Code = 0x1E, the CIP Extended Error Code is = 0x00 and the IO-Link Error Code and the IO-Link Additional Code will be transmitted in the CIP User Specific Service Data area. The response has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4C	"Write ISDU" service
CIP Error Code	USINT	<> 0x00	Error code (see below)
CIP Extended Error Code	USINT	0x00	Extended error code
CIP User Specific Service Data	USINT	<> 0x00	IO-Link Error Code: Error Code (→ 196)
	USINT	<> 0x00	IO-Link Additional Code: Additional Code (→ 196)

CIP Error Code:

Code	Description
0x02	Resource not available: The IO-Link port is busy processing another acyclic service.
0x05	Invalid class ID or instance ID
0x08	Wrong service ID: only service 0x4B, 0x4C or 0x4D is permitted
0x09	Wrong attribute ID: wrong port number
0x20	Invalid parameter value (e.g. invalid length)
0x1E	Embedded service error: Error occurred during an IO-Link service. Byte 0 and byte 1 of the CIP User Specific Service Data contain the IO-Link error code and an additional code returned by the IO-Link master.

Write Failsafe Pattern

Write Failsafe Pattern can be used to write the failsafe value of an IO-Link port.

Request

CIP Attribute determines the IO-Link port. The area CIP User Specific Service Data includes the fail-safe mode and the fail-safe value (Failsafe Pattern).

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4D	"Write Failsafe Pattern" service
CIP User Specific Service Data	USINT	0x00: No Fail-safe 0x01: Fail-safe Reset Value 0x02: Fail-safe Old Value 0x03: Fail-safe with Pattern	Fail-safe Mode
	USINT	0x00...0xFF	Fail-safe Pattern (MSB)
	USINT	0x00...0xFF	Fail-safe Pattern (LSB)

Response

Fault-free processing

If the service was executed successfully (CIP Error Code = 0), the area "User Specific Data" will remain empty. The response has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number

CIP Format	Data type	MSG Config	Description
CIP Service Code ID	USINT	0x4D	"Write Failsafe Pattern" service
CIP Error Code	USINT	0x00	Fault-free processing
CIP Extended Error Code	USINT	0x00	--

Fault

If an error occurs while executing the service (CIP Error Code $\neq$ 0), an extended error code will be transmitted (CIP Extended Error Code). The response has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Error Code	USINT	$\neq$ 0x00	Error code (see below)
CIP Extended Error Code	USINT	0x00	Extended error code

CIP Error Code:

Code	Description
0x02	Resource not available: The IO-Link port is busy processing another acyclic service.
0x05	Invalid class ID or instance ID
0x08	Wrong service ID: only service 0x4B, 0x4C or 0x4D is permitted
0x09	Wrong attribute ID: wrong port number
0x20	Invalid parameter value (e.g. invalid length)
0x1E	Embedded service error: Error occurred during an IO-Link service. Byte 0 and byte 1 of the CIP User Specific Service Data contain the IO-Link error code and an additional code returned by the IO-Link master.
0x0F	Insufficient access rights

Write Port Power Mode

The Write Port Power Mode service can be used to control the power supply of a port (pin 1), e.g. switching off and on again.

Request

CIP Attribute determines the IO-Link port. The CIP User Specific Service Data area contains the desired port supply mode and the switch-off duration.

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4E	"Write Port Power Mode" service
CIP User Specific Service Data	USINT	0x00: One Time Switch Off 0x01: Switch Port Power Off 0x02: Switch Port Power On	Port Power Mode
	USINT	0x00...0x01	Power Off Time (MSB)
	USINT	0x00...0xFF	Power Off Time (LSB)

Response

Fault-free processing

If the service was executed without errors (CIP Error Code = 0), the User Specific Data area remains empty. The answer has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4E	"Write Port Power Mode" service
CIP Error Code	USINT	0x00	Fault-free processing
CIP Extended Error Code	USINT	0x00	--

Fault

If an error occurs during the execution of the service (CIP Error Code $\neq$ 0), an extended error code is transmitted (CIP Extended Error Code). The answer has the following format:

CIP Format	Data type	MSG Config	Description
CIP Class ID	UINT	0x80	IO-Link acyclic access
CIP Instance ID	UINT	0x01	IO-Link master
CIP attributes	USINT	0x01...0x08	Port number
CIP Service Code ID	USINT	0x4E	"Write Port Power Mode" service
CIP Error Code	USINT	$\neq$ 0x00	Error code (see below)
CIP Extended Error Code	USINT	0x00	Extended error code

CIP Error Code:

Code	Description
0x02	Resource not available: The IO-Link port is busy processing another acyclic service.
0x05	Invalid class ID or instance ID
0x08	Wrong service ID: only service 0x4B, 0x4C or 0x4D is permitted
0x09	Wrong attribute ID: wrong port number
0x20	Invalid parameter value (e.g. invalid length)
0x1E	Embedded service error: Error occurred during an IO-Link service. Byte 0 and byte 1 of the CIP User Specific Service Data contain the IO-Link error code and an additional code returned by the IO-Link master.
0x0F	Insufficient access rights

11.2.4.10 TCP/IP Object (Class Code: 0xF5)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	4
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances of the object	1
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attribute	UINT	Max. ID number of the instance attribute	14

Instance attributes

Attr. ID	Access	Name	Data type	Description	Default
1	Get	Status	UINT	Status of the TCP/IP interface 0: interface not configured 1: interface configured with DHCP, BOOTP or non-volatile memory	
2	Get	Configuration Capability	DWORD	Configuration options for interface - Bit 0: BOOTP Client - Bit 2: DHCP Client - Bit 4: configuration adjustable - Bit 7: ACD-capable	0x95 (BOOTP, DHCP, TCP/IP configurable, ACD capable)
3	Get / Set	Configuration Control	DWORD	Configuration control - Bit 0..3: startup configuration 0: statically assigned IP address 1: configuration via BOOTP 2: configuration via DHCP	0
4	Get	Physical Link Object Path	STRUCT of	Logical path to the physical communication interface (Ethernet link object)	
		• Path Size	UINT	Length (number of words, little endian format)	0x02 00
		• Path	Padded EPATH	Class ID: 0xF6 (Ethernet Link Object) Instance ID: 0x1	0x20 F6 2 4 01
5	Get / Set	Interface configuration	STRUCT of	TCP/IP interface configuration	
		• IP Address	UDINT	IP address	192.168.1.250
		• Network Mask	UDINT	Network mask	255.255.255.0
		• Gateway Address	UDINT	Gateway address	0.0.0.0
		• Name Server	UDINT	Primary name server	0.0.0.0
		• Name Server 2	UDINT	Secondary name server	0.0.0.0
		• Domain Name	STRING	Default domain name	0
6	Get / Set	Host Name	STRING	Host Name 0: not configured	0
8	Get	TTL Value	UINT	TTL value	1
9	Get / Set	Mcast Config	UINT	Mcast configuration	0
10	Get / Set	SelectAcd	BOOL	ACD activation / deactivation 0: deactivated 1: activated	1
11	Get / Set	Last conflict detected	STRUCT of	Last conflict detected	0

Attr. ID	Access	Name	Data type	Description	Default
11	Get / Set	• AcdActivity	USINT	Condition of ACD activity at last conflict detected • 0: Noconflict detected • 1: Probelpv4Address • 2: Ongoing Detection • 3: SemiActiveprobe	
		• Remote MAC	ARRAY(6) of USINT	MAC address of the remote node of the ARP PDU in which the conflict was detected	
		• ArpPdu	ARRAY(28) of USINT	Copy of the data of the ARP PDU in which the conflict was detected	
13	Get / Set	Encapsulation Inactivity Timeout	UINT	Inactivity before the TCP connection is deactivated (in seconds)	120

Services

Code	Service	Class	Instance	Description
0x01	Get Attribute All	No	Yes	Read all attributes
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute

11.2.4.11 Ethernet Link Object (Class Code: 0xF6)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	4
2	Get	Max. Instance	UINT	Max. number of instances of the object	2
3	Get	Number of instances	UINT	Number of instances	2
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attribute	UINT	Max. ID number of the instance attribute	768

Instance attributes

Attr. ID	Access	Name	Data type	Description	Default
1	Get	Interface Speed	UDINT	Current transmission rate 10: 10 Mbit/s 100: 100 Mbit/s	100
2	Get	Interface Status Flag	DWORD	Interface status flags - Bit 0: link status - Bit 1: half/full duplex - Bit 2...4: negotiation status - Bit 5: manual change requires reset - Bit 6: local hardware error - Bit 7...31: reserved	0x20
3	Get	Physical Address	ARRAY(6) of USINT	MAC address	
4	Get	Interface Counters	STRUCT(11) of UDINT	Interface-specific counter	
5	Get	Media Counters	STRUCT(12) of UDINT	Medium-specific counter	
6	Get / Set	Interface Control	STRUCT of	Interface control	
		• Interface Settings	DWORD	Settings - Bit 0: 0: auto-negotiation on 1: auto-negotiation off - Bit 1: 0: half duplex 1: full duplex	0
		• Interface Speed	UINT	Transmission rate 10: 10 Mbit/s 100: 100 Mbit/s	
7	Get	Interface type	USINT	0: unknown 1: internal interface 2: twisted pair 3: optical fibre	2
8	Get	Interface State	USINT	Current status of the interface 0: unknown 1: active; ready for transmission and reception 2: not active 3: test mode	0

Attr. ID	Access	Name	Data type	Description	Default
9	Get / Set	Admin State	USINT	Control of the access to the interface • 1: activate • 2: deactivate	1
10	Get	Interface Label	SHORT_STRING	Interface identifier	• XF1 (instance 1) • XF2 (instance 2)
11	Get	Interface Capability	STRUCT of	Interface capability	
		• Interface Speed	DWORD	Transmission rate • 10: 10 Mbit/s • 100: 100 Mbit/s	
		• Interface Duplex Mode	DWORD	Duplex mode • HD: half duplex • FD: full duplex	
768	Get / Set	MDIX	USINT	MDIX configuration • 1: MDIX_AUTO • 2: MDIX_MDI • 3: MDIX_MDIX	1

Services

Code	Service	Class	Instance	Description
0x01	Get Attribute All	No	Yes	Read all attributes
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute

11.2.4.12 LLDP Management Object (Class Code: 0x109)

Class attributes

Attr. ID	Access	Name	Data type	Description	Value
1	Get	Revision	UINT	Revision of the object	1
2	Get	Max. Instance	UINT	Max. number of instances of the object	1
3	Get	Number of instances	UINT	Number of instances of the object	1
6	Get	Max. ID Number Class Attributes	UINT	Max. ID number of a class attribute	7
7	Get	Max. ID Number Instance Attributes	UINT	Max. ID number of the instance attribute	5

Instance attributes

Attr. ID	Access	Name	Data type	Description	Default
1	Get / Set	LLDP Enable	UINT	Enable / disable LLDP • 0: disabled • 1: enabled for all ports	1
2	Get / Set	msgTxInterval	UINT	Interval for transmission of LLDP frames (in seconds)	30
3	Get / Set	msgTxHold	USINT	Interval time multiplier, factor for determining the hold time for transmission to adjacent devices • e.g. 4: 4x interval time	4
4	Get	LLDP Datastore	UINT	Supported methods for querying the LLDP database • 0x02: SNMP	0x02
5	Get / Set	Last Change	DWORD	Time since last change in local LLDP database (in seconds)	0

Services

Code	Service	Class	Instance	Description
0x0E	Get Attribute Single	Yes	Yes	Read single attribute
0x10	Set Attribute Single	No	Yes	Write single attribute

11.3 PROFINET

11.3.1 Modules

11.3.1.1 IOLM Proxy Module

Modules	IOLM Proxy	Description
8 Ports	IO-Link Master Proxy	IO-Link master with 8 ports and energy monitoring

11.3.2 Submodules

11.3.2.1 IOLM Proxy Submodule: IO-Link master proxy

Name	Subslots	Description
IO-Link master proxy	1	IO-Link master proxy with energy monitoring

11.3.2.2 IOLD Proxy Submodule: IO-Link master proxy

IO-Link Input + Output + PQI

Submodule	Port Mode	Description
IO-Link 1 I / 1 O Byte + PQI	IO-Link	1 byte IO-Link input data and 1 byte IO-Link output data with PQI
IO-Link 4 I / 1 O Byte + PQI	IO-Link	4 bytes of IO-Link input data and 1 byte of IO-Link output data with PQI
IO-Link 8 I / 1 O Byte + PQI	IO-Link	8 bytes of IO-Link input data and 1 byte of IO-Link output data with PQI
IO-Link 16 I / 1 O Byte + PQI	IO-Link	16 bytes of IO-Link input data and 1 byte of IO-Link output data with PQI
IO-Link 32 I / 1 O Byte + PQI	IO-Link	32 bytes of IO-Link input data and 1 byte of IO-Link output data with PQI
IO-Link 1 I / 1 O Byte + PQI	IO-Link	1 byte IO-Link input data and 1 byte IO-Link output data with PQI
IO-Link 1 I / 15 O Byte + PQI	IO-Link	1 byte IO-Link input data and 15 bytes IO-Link output data with PQI
IO-Link 2 I / 2 O Byte + PQI	IO-Link	2 bytes of IO-Link input data and 2 bytes of IO-Link output data with PQI
IO-Link 4 I / 4 O Byte + PQI	IO-Link	4 bytes of IO-Link input data and 4 bytes of IO-Link output data with PQI
IO-Link 8 I / 8 O Byte + PQI	IO-Link	8 bytes of IO-Link input data and 8 bytes of IO-Link output data with PQI
IO-Link 16 I / 16 O Byte + PQI	IO-Link	16 bytes of IO-Link input data and 16 bytes of IO-Link output data with PQI
IO-Link 32 I / 32 O Byte + PQI	IO-Link	32 bytes of IO-Link input data and 32 bytes of IO-Link output data with PQI

IO-Link Input + PQI

Submodule	Port Mode	Description
IO-Link 1 I Byte + PQI	IO-Link	1 byte of IO-Link input data with PQI
IO-Link 2 I Byte + PQI	IO-Link	2 byte of IO-Link input data with PQI
IO-Link 4 I Byte + PQI	IO-Link	4 byte of IO-Link input data with PQI
IO-Link 8 I Byte + PQI	IO-Link	8 byte of IO-Link input data with PQI
IO-Link 16 I Byte + PQI	IO-Link	16 byte of IO-Link input data with PQI
IO-Link 32 I Byte + PQI	IO-Link	32 byte of IO-Link input data with PQI

IO-Link Output + PQI

Submodule	Port Mode	Description
IO-Link 1 O Byte + PQI	IO-Link	1 byte of IO-Link output data with PQI
IO-Link 2 O Byte + PQI	IO-Link	2 byte of IO-Link output data with PQI
IO-Link 4 O Byte + PQI	IO-Link	4 byte of IO-Link output data with PQI
IO-Link 8 O Byte + PQI	IO-Link	8 byte of IO-Link output data with PQI
IO-Link 16 O Byte + PQI	IO-Link	16 byte of IO-Link output data with PQI

Submodule	Port Mode	Description
IO-Link 32 O Byte + PQI	IO-Link	32 byte of IO-Link output data with PQI

Digital Input / Output

Submodule	Port Mode	Description
Digital Input	DI	Digital input
Digital Output	DO	Digital output

IO-Link Deactivated

Submodule	Port Mode	Description
Deactivated	Deactivated	Port deactivated

11.3.3 Parameters

11.3.3.1 IOLM Proxy Module

Parameters: 8 ports

Parameter	Description	Value range
Enable US Diagnosis	Enable / disable diagnostic data of the supply voltage US	☐: deactivated ☒: enabled (default)
Enable UA Diagnosis	Enable / disable diagnostic data of the supply voltage UA	☐: deactivated ☒: enabled (default)
Port Configuration without Tool	Enable the configuration of the IO-Link ports via IODD and the configuration tool	☐: deactivated ☒: enabled (default)
Mode I/Q (Actuator Supply UA)	Operating mode of pin 2 (UA) of the IO-Link port X1...X4	- OFF (IO-Link Type A Supply): UA disabled -- operation as IO-Link port (Class A) (default) - ON (IO-Link Type B Supply): UA enabled -- operation as IO-Link port (Class B) - Digital Output: operation as digital output
Fail Safe DO mode I/Q (Actuator Supply UA)	Behaviour of the additional digital output on pin 2 of ports X1...X4 if the PROFINET connection is interrupted	- Fail Safe Reset Value: Set output to LOW - Fail Safe Old Value: Output retains the last active process value (default) - Fail Safe Set Value: Set output to HIGH

11.3.3.2 IOLD Proxy Submodule

Parameters: Deactivated

Parameter	Description	Value range
Enable Port Diagnosis	Enable / disable diagnostic messages of the port	☐: deactivated ☒: activated (default)
Enable Process Alarm	Enable / disable process alarms	☐: deactivated ☒: activated (default)

Parameters: Digital Input

Parameter	Description	Value range
Enable Port Diagnosis	Enable / disable diagnostic messages of the port	☐: deactivated ☒: activated (default)
Enable Process Alarm	Enable / disable process alarms	☐: deactivated ☒: activated (default)

Parameters: Digital Output

Parameter	Description	Value range
Enable Port Diagnosis	Enable / disable diagnostic messages of the port	☐: deactivated ☒: activated (default)
Enable Process Alarm	Enable / disable process alarms	☐: deactivated ☒: activated (default)
Fail safe mode	Behaviour of the output if the PROFINET connection is interrupted	- Fail Safe Reset Value: Fail-safe mode enabled - set outputs to LOW - Fail Safe Old Value: Fail-safe mode enabled - outputs retain the last set process value (default) - Fail Safe Set Value: Fail-safe mode enabled - set outputs to HIGH

Parameters: IO-Link - Input

Parameter	Description	Value range
Enable Port Diagnosis	Enable / disable diagnostic messages of the port	☐: deactivated ☒: activated (default)
Enable Process Alarm	Enable / disable process alarms	☐: deactivated ☒: activated (default)
Port Configuration without Tool	Enable configuration of the IO-Link ports via IODD and the parameterization tool	- Disabled: deactivated - Enabled: activated (default)
Enable Pull/Plug	Enable / disable pull/plug alarms	- Disabled: deactivated - Enabled: activated (default)
Port Mode	Configuration mode of the ports during start-up	- Apply port configuration: Use values projected in PROFINET - Autostart: Use the set values of the IO-Link device

Parameter	Description	Value range
Validation / Data Storage	Supported IO-Link standard and behaviour of the port when connecting new IO-Link devices to the IO-Link port	- no Device check and clear: - No device validation - No data storage (default) - type compatible Device (V1.0): - IO-Link device is compatible with the V1.0 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - No data storage - type compatible Device (V1.1): - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - No data storage - type compatible Device (V1.1) with Backup + Restore: - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - The IO-Link master saves the parameter values of the connected IO-Link device; The changes of the parameter values are saved as well - When connecting an IO-Link device with factory settings, the parameter values stored in the IO-Link master are restored automatically on the IO-Link device. - type compatible Device (V1.1) with Restore: - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - The IO-Link master stores the parameter values of the connected IO-Link device once if the data memory of the device is empty. - When connecting an IO-Link device with factory settings, the parameter values stored in the IO-Link master are restored automatically on the IO-Link device.
Port cycle time	Cycle time of the data transmission on the IO-Link port	- Device automatically sets the fastest possible cycle time (default) 2.0 ms: 2 milliseconds 4.0 ms: 4 milliseconds 8.0 ms: 8 milliseconds 16.0 ms: 16 milliseconds 32.0 ms: 32 milliseconds 64.0 ms: 64 milliseconds 128.0 ms: 128 milliseconds
Vendor ID	ID of the manufacturer that is to be validated	0 (default)...65535
Device ID	ID of the IO-Link device that is to be validated	0 (default)...16777215

Parameters: IO-Link - Output

Parameter	Description	Value range
Enable Port Diagnosis	Enable / disable diagnostic messages of the port	☐: deactivated ☒: activated (default)
Enable Process Alarm	Enable / disable process alarms	☐: deactivated ☒: activated (default)
Port Configuration without Tool	Enable configuration of the IO-Link ports via IODD and the parameterization tool	- Disabled: deactivated - Enabled: activated (default)
Enable Pull/Plug	Enable / disable pull/plug alarms	- Disabled: deactivated - Enabled: activated (default)
Port Mode	Configuration mode of the ports during start-up	- Apply port configuration: Use values projected in PROFINET - Autostart: Use the set values of the IO-Link device
Validation / Data Storage	Supported IO-Link standard and behaviour of the port when connecting new IO-Link devices to the IO-Link port	- no Device check and clear: - No device validation - No data storage (default) - type compatible Device (V1.0): - IO-Link device is compatible with the V1.0 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - No data storage - type compatible Device (V1.1): - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - No data storage - type compatible Device (V1.1) with Backup + Restore: - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - The IO-Link master saves the parameter values of the connected IO-Link device; The changes of the parameter values are saved as well - When connecting an IO-Link device with factory settings, the parameter values stored in the IO-Link master are restored automatically on the IO-Link device. - type compatible Device (V1.1) with Restore: - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - The IO-Link master stores the parameter values of the connected IO-Link device once if the data memory of the device is empty. - When connecting an IO-Link device with factory settings, the parameter values stored in the IO-Link master are restored automatically on the IO-Link device.

Parameter	Description	Value range
Port cycle time	Cycle time of the data transmission on the IO-Link port	• Device automatically sets the fastest possible cycle time (default) • 2.0 ms: 2 milliseconds • 4.0 ms: 4 milliseconds • 8.0 ms: 8 milliseconds • 16.0 ms: 16 milliseconds • 32.0 ms: 32 milliseconds • 64.0 ms: 64 milliseconds • 128.0 ms: 128 milliseconds
Vendor ID	ID of the manufacturer that is to be validated	0 (default)...65535
Device ID	ID of the IO-Link device that is to be validated	0 (default)...16777215
Fail safe mode	Behaviour of the output if the PROFINET connection is interrupted (only valid for "DO" and "IO-Link" operating modes)	• No Fail Safe: Fail-safe mode disabled (default) • Fail Safe Reset Value: Fail-safe mode enabled - set outputs to LOW • Fail Safe Old Value: Fail-safe mode enabled - outputs retain the last set process value • Fail Safe Set Value: Fail-safe mode enabled - set outputs to HIGH • Failsafe with Pattern: Fail-safe mode enabled - set outputs to user-defined values
Pattern Value	Process data values in case of interruption of the PROFINET connection (value as hexadecimal value) Pattern depending on the scope of the selected PROFINET module	Per byte (outputs): • 0x00 (default) • ... • 0xFF

Parameters: IO-Link - Input / Output

Parameter	Description	Value range
Enable Port Diagnosis	Enable / disable diagnostic messages of the port	☐: deactivated ☒: activated (default)
Enable Process Alarm	Enable / disable process alarms	☐: deactivated ☒: activated (default)
Port Configuration without Tool	Enable configuration of the IO-Link ports via IODD and the parameterization tool	- Disabled: deactivated - Enabled: activated (default)
Enable Pull/Plug	Enable / disable pull/plug alarms	- Disabled: deactivated - Enabled: activated (default)
Port Mode	Configuration mode of the ports during start-up	- Apply port configuration: Use values projected in PROFINET - Autostart: Use the set values of the IO-Link device
Validation / Data Storage	Supported IO-Link standard and behaviour of the port when connecting new IO-Link devices to the IO-Link port	- no Device check and clear: - No device validation - No data storage (default) - type compatible Device (V1.0): - IO-Link device is compatible with the V1.0 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - No data storage - type compatible Device (V1.1): - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - No data storage - type compatible Device (V1.1) with Backup + Restore: - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - The IO-Link master saves the parameter values of the connected IO-Link device; The changes of the parameter values are saved as well - When connecting an IO-Link device with factory settings, the parameter values stored in the IO-Link master are restored automatically on the IO-Link device. - type compatible Device (V1.1) with Restore: - IO-Link device is compatible with the V1.1 IO-Link standard - Verification whether it is an IO-Link device of the same type (validation via vendor ID and device ID) - The IO-Link master stores the parameter values of the connected IO-Link device once if the data memory of the device is empty. - When connecting an IO-Link device with factory settings, the parameter values stored in the IO-Link master are restored automatically on the IO-Link device.

Parameter	Description	Value range
Port cycle time	Cycle time of the data transmission on the IO-Link port	• Device automatically sets the fastest possible cycle time (default) • 2.0 ms: 2 milliseconds • 4.0 ms: 4 milliseconds • 8.0 ms: 8 milliseconds • 16.0 ms: 16 milliseconds • 32.0 ms: 32 milliseconds • 64.0 ms: 64 milliseconds • 128.0 ms: 128 milliseconds
Vendor ID	ID of the manufacturer that is to be validated	0 (default)...65535
Device ID	ID of the IO-Link device that is to be validated	0 (default)...16777215
Fail safe mode	Behaviour of the output if the PROFINET connection is interrupted (only valid for "DO" and "IO-Link" operating modes)	• No Fail Safe: Fail-safe mode disabled (default) • Fail Safe Reset Value: Fail-safe mode enabled - set outputs to LOW • Fail Safe Old Value: Fail-safe mode enabled - outputs retain the last set process value • Fail Safe Set Value: Fail-safe mode enabled - set outputs to HIGH • Failsafe with Pattern: Fail-safe mode enabled - set outputs to user-defined values
Pattern Value	Process data values in case of interruption of the PROFINET connection (value as hexadecimal value) Pattern depending on the scope of the selected PROFINET module	Per byte (outputs): • 0x00 (default) • ... • 0xFF

11.3.4 Cyclic data

11.3.4.1 IOLM proxy: IO-Link master proxy

Input data: 6 byte

Byte	Contents							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	I/Q Status							
1	Supply Status							
2...3	Voltage Sensor Supply US [mV]							
4...5	Current Sensor Supply US [mA]							

Legend:

- I/Q Status Switching states of the digital inputs Unsigned8 • [Mapping: I/Q Status \(→ □ 235\)](#)
- Supply Status Status of the voltage supplies US and UA Unsigned8 • [Mapping: Supply Status \(→ □ 235\)](#)
- Voltage Sensor Supply US [mV] Current voltage of the power supply US (value in mV) Unsigned16 • 0x4E20: 20000 mV
...
• 0x7530: 30000 mV
- Current Sensor Supply US [mA] Present current intensity of the voltage supply US (value in mA) Unsigned16 • 0x0000: 0 mA
...
• 0x0E10: 3600 mA

Output data: 1 byte

Byte	Contents							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	I/Q Output							

- I/Q Output Switching states of the digital outputs Unsigned8 • [Mapping: I/Q Output \(→ □ 235\)](#)

Mapping: I/Q Status

Byte	Contents							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
n	Port X8: DI (Pin 2)	Port X7: DI (Pin 2)	Port X6: DI (Pin 2)	Port X5: DI (Pin 2)	res.	res.	res.	res.

Legend:

- DI (Pin 2) Switching status of the digital input of the port (pin 2) 1 BIT
 - 0x0: LOW
 - 0x1: HIGH

Mapping: Supply Status

Byte	Contents							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
n	res.	res.	res.	res.	res.	res.	Status Sen- sor Supply US	Status Actua- tor Supply UA

Legend:

- Status Actuator Supply UA Status of the UA voltage supply 1 BIT
 - 0x0: UA is connected and no error
 - 0x1: Error
- Status Sensor Supply US Status of the US voltage supply 1 BIT
 - 0x0: No error
 - 0x1: Error

Mapping: I/Q Output

Byte	Contents							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
n	res.	res.	res.	res.	Port X4: DO (Pin 2)	Port X3: DO (Pin 2)	Port X2: DO (Pin 2)	Port X1: DO (Pin 2)

Legend:

- DO (Pin 2) Switching status of the digital output of the port (pin 2) 1 BIT
 - 0x0: LOW
 - 0x1: HIGH

11.3.4.2 IOLD Proxy: IO-Link n I / m O + PQI

Input data: n+1 bytes

Byte	Contents
0...n-1	Input data n Bytes
n	Port Qualifier Information (PQI)

Legend:

- Input data n Bytes IO-Link input data (pin 4) OctetString Per octet:
 - 0x00...0xFF
- Port Qualifier Information (PQI) IO-Link output data (pin 4) Unsigned8 • [Mapping: Port Qualifier Information \(PQI\)](#) (→ [236](#))

Output data: m Bytes

Byte	Contents
0	Output data m Bytes

Legend:

- Output data m Bytes IO-Link output data (pin 4) OctetString Per octet:
 - 0x00...0xFF

Mapping: Port Qualifier Information (PQI)

Byte	Bit							
	7	6	5	4	3	2	1	0
n	PQ	DevErr	DevCom	PortActive	SubstDev	NewPar	res.	res.

Legend:

- NewPar Indication of a data storage update 1 BIT
 - 0x0: no update
 - 0x1: update detected; The IO-Link master has updated Data Storage, and a new IO-Link Device Backup object is available
- SubstDev Indication of an exchange of the IO-Link device (comparison of the serial numbers) 1 BIT
 - 0x0: No exchange detected
 - 0x1: Exchange detected
- PortActive Indication of the port status 1 BIT
 - 0x0: Port disabled via the port function
 - 0x1: Port enabled
- DevCom Indication of the availability of an IO-Link device 1 BIT
 - 0x0: IO-Link device not available
 - 0x1: IO-Link device available; IO-Link device in PREOPERATE or OPERATE state
- DevErr Indication of errors / warnings from the IO-Link device or port 1 BIT
 - 0x0: no fault / no warning
 - 0x1: error / warning
- PQ Status of the process input data on the IO-Link port 1 BIT
 - 0x0: invalid data
 - 0x1: valid data

11.3.4.3 IOLD Proxy: IO-Link n I + PQI

Input data: n+1 bytes

Byte	Contents
0...n-1	Input data n Bytes
n	Port Qualifier Information (PQI)

Legend:

- Input data n Bytes IO-Link input data (pin 4) OctetString Per octet:
 - 0x00...0xFF
- Port Qualifier Information (PQI) IO-Link output data (pin 4) Unsigned8
 - [Mapping: Port Qualifier Information \(PQI\)](#) (→ [□ 237](#))

Output data: --

Mapping: Port Qualifier Information (PQI)

Byte	Bit							
	7	6	5	4	3	2	1	0
n	PQ	DevErr	DevCom	PortActive	SubstDev	NewPar	res.	res.

Legend:

- NewPar Indication of a data storage update 1 BIT
 - 0x0: no update
 - 0x1: update detected; The IO-Link master has updated Data Storage, and a new IO-Link Device Backup object is available
- SubstDev Indication of an exchange of the IO-Link device (comparison of the serial numbers) 1 BIT
 - 0x0: No exchange detected
 - 0x1: Exchange detected
- PortActive Indication of the port status 1 BIT
 - 0x0: Port disabled via the port function
 - 0x1: Port enabled
- DevCom Indication of the availability of an IO-Link device 1 BIT
 - 0x0: IO-Link device not available
 - 0x1: IO-Link device available; IO-Link device in PREOPERATE or OPERATE state
- DevErr Indication of errors / warnings from the IO-Link device or port 1 BIT
 - 0x0: no fault / no warning
 - 0x1: error / warning
- PQ Status of the process input data on the IO-Link port 1 BIT
 - 0x0: invalid data
 - 0x1: valid data

11.3.4.4 IOLD Proxy: IO-Link m O + PQI

Input data: 1 byte

Byte	Contents
0	Port Qualifier Information (PQI)

Legend:

- Port Qualifier Information (PQI) IO-Link output data (pin 4) Unsigned8 • [Mapping: Port Qualifier Information \(PQI\) \(→ 238\)](#)

Output data: m Bytes

Byte	Contents
0	Output data m Bytes

Legend:

- Output data m Bytes IO-Link output data (pin 4) OctetString Per octet:
 - 0x00...0xFF

Mapping: Port Qualifier Information (PQI)

Byte	Bit							
	7	6	5	4	3	2	1	0
n	PQ	DevErr	DevCom	PortActive	SubstDev	NewPar	res.	res.

Legend:

- NewPar Indication of a data storage update 1 BIT
 - 0x0: no update
 - 0x1: update detected; The IO-Link master has updated Data Storage, and a new IO-Link Device Backup object is available
- SubstDev Indication of an exchange of the IO-Link device (comparison of the serial numbers) 1 BIT
 - 0x0: No exchange detected
 - 0x1: Exchange detected
- PortActive Indication of the port status 1 BIT
 - 0x0: Port disabled via the port function
 - 0x1: Port enabled
- DevCom Indication of the availability of an IO-Link device 1 BIT
 - 0x0: IO-Link device not available
 - 0x1: IO-Link device available; IO-Link device in PREOPERATE or OPERATE state
- DevErr Indication of errors / warnings from the IO-Link device or port 1 BIT
 - 0x0: no fault / no warning
 - 0x1: error / warning
- PQ Status of the process input data on the IO-Link port 1 BIT
 - 0x0: invalid data
 - 0x1: valid data

11.3.4.5 IOLD Proxy: Digital Input

Input data: 1 byte

Byte	Contents
0	Digital I/O data 1 Byte

Legend:

- Digital I/O data 1 Byte Switching state of the digital input (pin Unsigned8 • 0x00: LOW
4) • 0x01: HIGH

11.3.4.6 IOLD Proxy: Digital Output

Output data: 1 byte

Byte	Contents
0	Digital I/O data 1 Byte

Legend:

- Digital I/O data 1 Byte Switching state of the digital output (pin 4) Unsigned8
 - 0x00: LOW
 - 0x01...0xFF: HIGH

11.3.5 Acyclical data

11.3.5.1 I&M data

I&M1 (Slot 0)

Index: 0xAFF1

Variable	Description	Value	Bytes
TAG_FUNCTION	Identifier for function of the submodule • 0x20: empty	0x20	32
TAG_LOCATION	Identifier for location of the submodule • 0x20: empty	0x20	22

I&M2 (Slot 0)

Index: 0xAFF2

Variable	Description	Value	Bytes
INSTALLATION_DATE	Installation date of the submodule (ASCII, separated by spaces) • 0x20: empty	0x20	16
RESERVED	reserved	0x00	38

I&M3 (Slot 0)

Index: 0xAFF3

Variable	Description	Value	Bytes
DESCRIPTOR	Description of the submodule (ASCII, separated by spaces) • 0x20: empty	0x20	54

I&M0 (Slot 1)

Index: 0xAFF4

Variable	Description	Value	Bytes
API	API of the submodule		4
SLOT	Slot of the submodule		2
SUBSLOT	Subslot of the submodule		2
FLAGS	Flags: • 0x01: submodule has own I&M data • 0x02: I&M data of the submodule represent I&M data of the module • 0x04: I&M data of the submodule represent I&M data of the device		4