CMi6160 G2

User manual

elvaco

2.2 Edition

Table of Contents

1. About this manual

This manual covers information needed to mount, install, configure, and use the product. It is intended for installers and system integrators.

To download the latest version of this User manual, visit the Elvaco website, https://www.elvaco.com. There you will also find information about Elvaco's other products and services.

1.1. Symbols

The following symbols are used throughout the manual to emphasize important information and useful tips:

Warning
Indicates a potentially dangerous situation that could result in severe injuries or serious equipment damage.

Caution
Indicates a potentially dangerous situation that could result in minor injuries or equipment damage.

Note
Indicates information that is important to take into consideration for safety reasons or to assure correct operation of the product.

Tip
Indicates information intended to help you get the most out of your product. It can for example be used to highlight a possible customization option related to the current section.

1.2. Terms and abbreviations

Abbreviation	Description
CBOR	Concise Binary Object Representation
COSE	CBOR Object Signing and Encryption
DevEUI	Device Extended Unique Identifier
DM	Device Management
DNS	Domain Name Server
DTLS	Datagram Transport Layer Security
IP	Internet Protocol
LPWAN	Low Power Wide Area Network
LwM2M	Lightweight Machine to Machine
MCM	Meter Connectivity Module
MD	Meter Data
MQTT	MQ Telemetry Transport
MQTT-SN	MQTT for Sensor Networks
NB-IoT	Narrowband Internet of Things
OSCORE	Object Security Constrained RESTful Environments
OTC	One-Touch Commissioning
PAK	Product Access Key
PSK	Pre-Shared Key
PSM	Power Save Mode
PSU	Power Supply Unit
SenML	Sensor Measurement List
TLS	Transport Layer Security

Abbreviation	Description
UDP	User Datagram Protocol
URI	Universal Resource Identifier

Number representation

- Decimal numbers are represented as normal number, i.e. 10 (ten)
- Hexadecimal numbers are represented with prefix 0x, i.e. 0x0A (ten)
- Binary numbers are represented with the prefix 0b, i.e. 0b00001010 (ten)

European standards

M-Bus standard EN 13757-3:2013: Communication systems for and remote reading of meters – Part 3: Dedicated application layer.

2. Safety

The following safety precautions must be observed during all phases of the operation, usage, service, or repair of the product. Users of the product are advised to convey the following safety information to users and operating personnel and to incorporate these guidelines into all manuals supplied with the product. Failure to comply with these precautions violates safety standards of design, manufacture and intended use of the product. Elvaco AB assumes no liability for customer's failure to comply with these precautions.

Note
The product receives and transmits radio frequency energy while switched on. Remember that interference can occur if the product is used close to TV sets, radios, computers or inadequately shielded equipment. Follow any special regulations and always switch off the product wherever forbidden, or when you suspect that it may cause interference or danger.

Note
To use the product's NFC TAG, follow the instructions issued by the manufacturer of the NFC reader for safe and efficient operation.

Note
For guaranteed lifetime of a battery-operated product, configuration and settings must be approved by Elvaco and not changed during the lifetime of the device.

Note
Waste electrical products should not be disposed of with household waste. Please recycle where facilities exist. Contact your Local Authority for recycling advise.

Caution
The product or antenna of the product must not be mounted closer than 0.5 m from areas where people are staying permanently in order not to risk exposing people to RF fields.

Caution
Ensure that any power supply and/or battery-unit connected to the product fulfils EN 62368-1 or equivalent safety standard.

Caution
This is an electrostatic-sensitive product. Observe the necessary ESD protective measures when installing the product.

Warning
Ensure all instructions and technical specifications in this manual are read and understood before the product is installed and used.

3. Product overview

3.1. Application description

CMi6160 is a meter connectivity module (MCM) that can be mounted inside a Diehl SHARKY 775 heating/cooling meter or a Diehl SCYLAR heating/cooling energy calculator. As soon as the module has been mounted and activated, it will start to deliver meter data to a receiving system via the NB-IoT network. The module is easily configured using the Elvaco OTC App or via a device management (DM) system. CMi6160 comes with several different message formats to choose from, developed to suit your specific metering application. It is ideal for applications where long range is required. The CMi6160 can be battery operated for up to 13 years. Meter data delivery and device management utilizes standard protocols, such as MQTT-SN and LwM2M for easy integration to existing systems.

3.2. Features

CMi6160 comes with a wide feature set, with the possibility to adapt its function based on your specific metering project. Some of the key features include:

- **IoT-ready:** As soon as the meter connectivity module has been mounted and activated, it will automatically initiate transmission of meter data over a NB-IoT network, without any manual steps needed. The CMi6160 is prepared for seamless integration with all leading IoT platforms.
- **Mains or battery powered:** Thanks to its efficient power usage, CMi6160 can be battery operated for up to 13 years with daily transmission of meter data. If preferred, the module can also be mains-powered.
- **One-Touch Commissioning:** The Elvaco OTC App enables fast, secure configuration and deployment of SHARKY and SCYLAR meters. Enter your desired settings in the Elvaco OTC App and hold your phone to the upper left corner of the meter. Settings are applied instantly via NFC. On firmware ≥ 1.3.2, the module can also be set to activate automatically when water flow is detected, removing the need for any in-field configuration.
- **Flexible message scheme:** CMi6160 has different message formats to choose from, making it easy to setup the device for your specific metering project. If using Transparent mode, the customer defined telegram within the meter can be forwarded, allowing a customized way of setting up the data being sent from the module.
- **Push of alarms:** Always stay up to date with the status of your meter park. The Meter Alarm Monitor sends alerts from the field, so you can quickly address issues and maintain optimal performance.

3.3. Compatibility

CMi6160 is compatible with Diehl SHARKY 775 and SCYLAR 548 meters with a compatible firmware version. Once the module is mounted and starts up, a compatibility check is made to ensure that it is compatible with the meter firmware. If the module is mounted in an incompatible meter, this is indicated in the Elvaco OTC App.

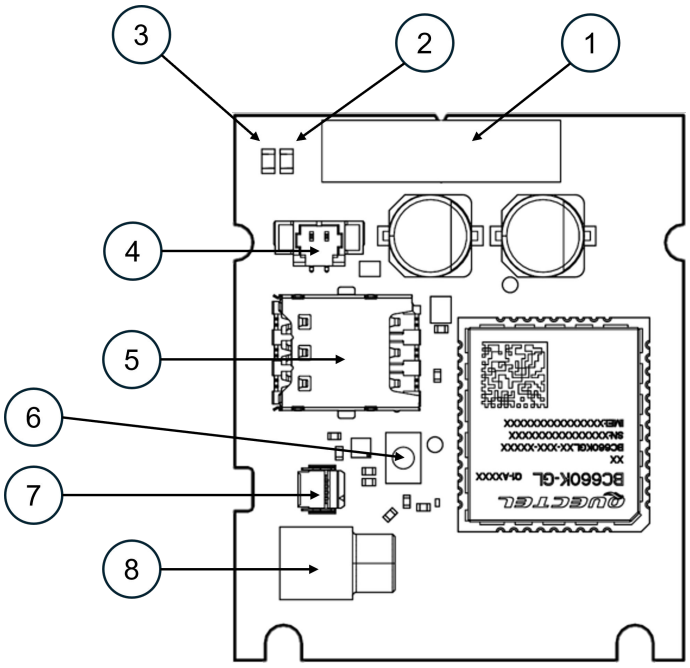
The power for CMi6160 is supplied via an external battery pack, unrelated to the power supply of the meter.

Below table shows the minimum meter firmware required for compatibility with CMi6160.

Table 1. CMi6160 - Meter firmware compatibility

Meter	Minimum required meter FW
Diehl SHARKY 775	FW version FW02
Diehl SCYLAR 548	FW version FW02

3.4. Parts overview



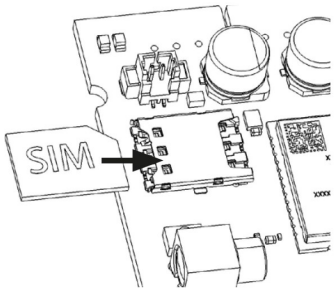
1. Meter Interface
2. LED - Green
3. LED - Red
4. Power Connector
5. SIM (Nano)
6. Push Button
7. NFC antenna connector
8. Antenna connector (MCX)

4. Installation

4.1. Mounting the meter connectivity module

Before being mount into a Diehl SHARKY 775 or a Diehl SCYLAR 548 meter, it is easiest to start off by mounting the SIM card.

1. Slide a SIM card (size: Nano) in the SIM card holder.



SIM card mounting

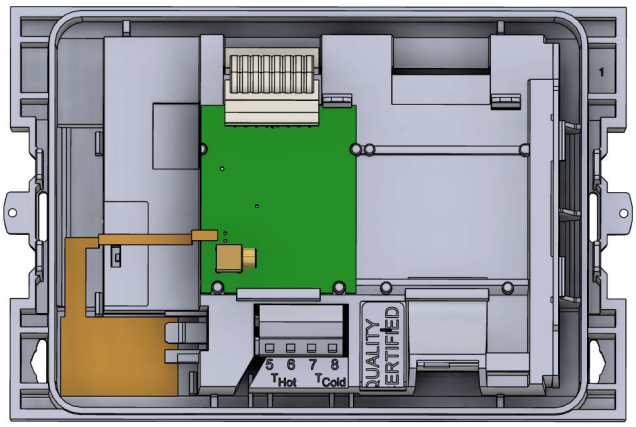
2. When the SIM card has been mounted, open the meter by folding down the side latches.

Having the SIM-card inserted in the device, mount the module in either a Diehl SHARKY 775 or a Diehl SCYLAR 548 by following the procedure below.

1. Connect the meter interface cable (white) by attaching it to the meter
2. Push the module (green) into meter slot 1 according to the following image. The guiding pins will help to get it into the right position.

> **Note**
> To be able to align the NFC antenna correctly, it is only possible to utilize meter slot 1.

3. Connect the NFC antenna (orange) to the module according to the following image. Remove the protecting film covering the adhesive on the backside of the NFC antenna. After connecting the antenna and removing the film, best practice is to firstly put the larger part in place, followed by pressing the thin part of the antenna into its position.



4. Connect the meter interface cable (white) to the module (green).
5. Attach the antenna of choice to the MCX connector, and connect the power connector before closing the lid.

> **Caution**
> This is an electrostatic-sensitive product. Observe the necessary ESD protective measures when installing the product.

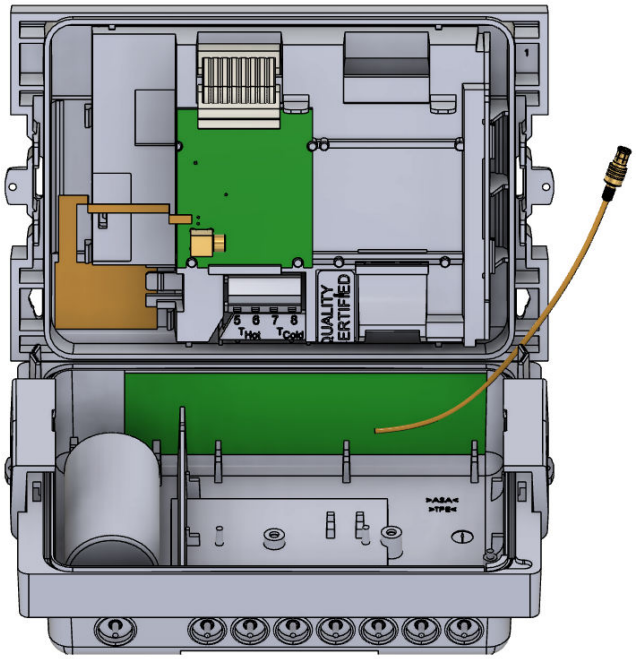
4.2. Connecting the battery

If not already assembled to the meter, see Diehl Metering's mounting instruction for external battery packs. When the battery pack has been mounted properly alongside the meter, connect the power cable from the battery to the power connector on the CMi6160.

4.3. Connection of the antenna

Using internal antenna

1. Place the antenna in the meter housing according to the following image.



2. Connect the antenna cable to the MCX antenna connector on the module.

> **Note**
> Make sure to not clamp the antenna cable when closing the lid.

Using external antenna

1. Make a hole in the rubber gromets/sealing and push the MCX connector through the gromets/sealing. Make sure that the thicker part of the antenna cable is in the gromet/sealing.
2. Connect the antenna cable to the MCX antenna connector on the module.

4.4. Activating the module

Upon delivery, the module is set to inactive mode, which means no messages will be transmitted from the module. The module can be activated in one of the following ways:

Via the Activation at flow feature

1. Make sure the module is configured with the Automatic activation at flow enabled. Utilize the Elvaco OTC App to check that the correct configuration is applied. To have the function enabled, **Power mode** should be set to **Standby**.

2. Once the meter detects flow of water and has registered sufficient water flow, the module will be activated and start to deliver meter data. The module checks once every hour if the meter has registered sufficient water flow.

> **Note**
> Sufficient water flow is when the meter has registered permanent water flow for 3 hours.

Via the module push button

1. Press down the push button of the module for at least five seconds.
2. Release the button when the green LED lights up.
3. Wait for up to 10 seconds.

The module indicates start-up by flashing red and green LEDs for one second.

Via Elvaco OTC app

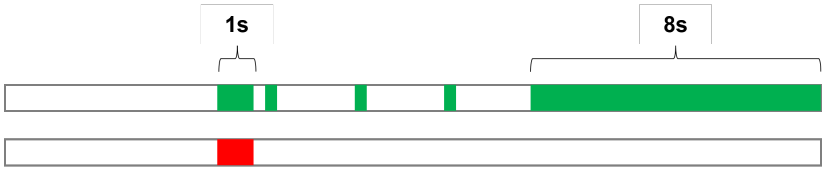
1. Open Elvaco OTC app (available for Android and iOS).
2. Scan the module (make sure NFC is activated on the phone).
3. Go to **Apply mode**.
4. Set the power mode to **Active**.
5. Select **Apply settings**.
6. Scan the module to apply new settings.
   To verify that the module has been activated, go to **Inspect**, scan the module, and make sure that power mode is set to **Active**.

> **Tip**
> Having the meter closed and the meter display facing you, the NFC antenna is located in the upper left corner of the meter.

4.5. Connecting to the network

When activated, the module will attempt to connect to the NB-IoT network. This phase is indicated by the green and red LED lights up for 1 second, followed by short flashes on the green LED until the module has joined the network. When the module succeeds in connecting to the network, the green LED will be lit for 8 seconds, as illustrated in the image below. Upon successful connection, a meter readout will automatically be sent to the receiving system.

If the module fails to join the mobile network, it will perform retries until it succeeds. The time between each attempt will increase for every attempt until it is performed once every day. A new join attempt cycle can be manually started anytime by using the push button to reboot the module or by deactivating and activating the module using the Elvaco OTC App.



LED indications, network connection

5. Operation

5.1. Meter data readout and transmission

The core settings of the module, what meter data is sent and how often, determine its overall behavior. Depending on the use case and requirements, such as battery life or data granularity, the module settings can be adjusted accordingly. The table below summarizes the essential settings that control the sending pattern and define which data points are included in the transmission.

Table 2. Essential settings to control meter data readouts and transmissions

LwM2M resource	Setting	Description
33906/./17	Meter Report Interval	Interval in minutes. Specifies how often the module reads and stores values from the meter. A short report interval means the meter is read often, meaning a higher data granularity can be achieved. The meter data are persistently stored in non-volatile memory and can be deleted by doing a factory reset. In case the memory is full, the oldest data will be replaced by the newest.
33906/./18	Meter Transmit Interval	Interval in minutes. Specifies how often the module sends data to the receiving system. A short transmit interval means the module will send data more frequent, allowing the system to get data from more recent meter readouts.
33906/./68	Measurement mode	A setting that determines whether the module operates in meter-controlled mode or module-controlled mode. In module-controlled mode, the module sends preset data according to the selected message format. In meter-controlled mode, it transparently transmits the customer-defined telegram, as specified in the meter, without modification. **Note** When using meter-controlled mode and transparent forwarding, message encoding is automatically set to SenML/CBOR to be able to accommodate larger customer-defined telegrams.
33906/./1	Report data encoding	Specifies how the data is encoded before transmission. Select the option that best suits the receiving system and other requirements

Tip
Transmitting data is a far more energy consuming operation than reading the meter. For battery operated devices, it is thus wisely to set a larger transmit interval than the report interval to achieve a decent data granularity.

5.1.1. Transmission of meter data

Meter data can be transmitted using either MQTT-SN or LwM2M Send. The protocols can not be used simultaneously, but it's possible to configure via LwM2M or the Elvaco OTC App. Regardless of the chosen transport protocol, the content of the transmitted data remains the same. When a meter data transmission is scheduled, the data being sent is defined by the chosen message format. If there are any unsent meter data available in the module, they will be transmitted according to the chosen recovery strategy (First in First out (FiFO) or Last in First out (LiFo).

Meter data transmission using MQTT-SN

When using MQTT-SN, meter data messages are published to a MQTT-SN topic. Topic can be configured remotely using LwM2M resource `33905/./11`, or via the Elvaco OTC App.

Meter data transmission using LwM2M Send

When using LwM2M Send as the protocol for meter data transfer, the data is published to the LwM2M object `33911`.

5.1.2. Time handling

The module relies on the meter's clock for keeping time. Time in the meter is assumed to be in standard local time (no DST). When synchronizing time in the meter using the Elvaco OTC App, local standard time is always used, even if DST is in effect. The timestamped meter data sent from the module can be adjusted to be sent in UTC by specifying the "UTC offset" configuration parameter. The UTC offset will be subtracted from the timestamp prior to

transmission. If the meter is in Sweden, which uses CET (Central European Time), it should have UTC offset set to +60 (+1h). In this case at time 12.00 a telegram is sent with timestamp 11.00 as this is the corresponding UTC time. A meter in New York (USA) should have a UTC offset of "-300" (-5h) etc. A UTC offset of "0" means the meter time is used as-is.

If the meter is set to used DST this is ignored by the module and the standard time is used. Thus, the time on the meter's display may not match the time in the telegram or in the Elvaco OTC App.

5.1.3. Synchronization

All schedules are based on a synchronization with a clock. This indicates that if a readout schedule of 60 minutes is used, it is synchronized on top of the hour, so 11:00, 12:00, 13:00 etc. 120 minutes will give 12:00, 14:00, 16:00 etc. When time in the module (or meter) is synchronized, a rescheduling takes place such that the next meter readout is made according to an updated time. Synchronization of the meter clock can be done in various ways, either periodically (automatically) or manually. It is recommended using a periodic approach to avoid the risk of letting the clock drift too much.

Periodic synchronization options

- Network time (default setting)
- NTP (Network Time Protocol)

Manual synchronization options

- Over NFC using Elvaco OTC App
- Over LwM2M using resource `3/./13`
- Setting the meter clock in the meter

To handle the case where time synchronization "moves time" past a previously planned readout (like 23.58 → 00.02) the module will always make a readout and transmission of a new value when time is synchronized. The device will, therefore, send an additional readout which can be masked on the server side.

5.1.4. Randomized transmissions

In order to prevent a large population of devices from transmitting data at exactly the same time the devices have a random delay before transmitting data. The delay is configurable via either the Elvaco OTC App and the NFC interface, or remotely via a device management system.

Readouts from the meter are always performed on top of the hour, 11.00, 13.00 etc. Transmissions can be carried out at other times but are planned at full ours given a set *transmission interval* ($T_{transmit}$). The figure below illustrates this. The transmissions are planned at time T_1. The actual $T_{transmit}$ is a random time between (T_1 + T_{offset}) and (T_1 + T_{offset} + T_{delay}).

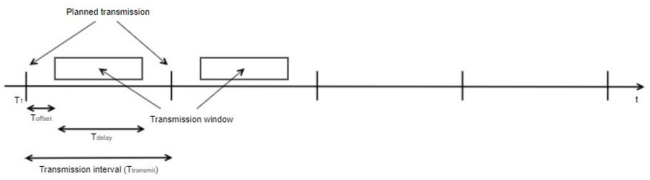
$T_{transmit}$, T_{offset} and T_{delay} are parameters in the product.

5.1.4.1. Conditions

- T_{offset} + T_{delay} <= $T_{transmit}$

This should be checked by the device and the OTC App.

- If $T_{transmit}$ is reduced below T_{offset} + T_{delay} , then T_{offset} should be set to 0 and T_{delay}.= $T_{transmit}$.



5.1.5. Data retransmission

If data cannot be sent, due for instance to network issues, there will be a number of retries after which the device will give up and leave the readout as “unsent” in its storage. Next time a transmission is attempted unsent data will be resent (if possible). Retransmission can be done by FIFO or LIFO.

Rules for retransmissions include maximum age of data, order of data, number of retransmitted data / transmission interval.

Example 1. Example 1

A device is configured the following way:

- Message encoding: M-Bus
- Auto upload order: FIFO
- Measurement interval: 60 minutes
- Transmit interval: 60 minutes
- Transmit offset: 15 minutes
- Transmit delay: 30 minutes
- Maximum uploads per transmission: 4
- Upload maximum age 72h

A network issue caused the module to be offline for 5 days, while still reading and storing measurement data. When the device manages to go online the following scenario takes place.

- The device will start by transmitting measurement data that is 3 days old (FIFO order)
- The device will send 4 measurement telegrams per hour, at a randomly chosen time between minute 15 and 45
- Each telegram contains a single readout, totaling 4 readouts per transmission
- The device will take approximately 1 day to “catch up” and start sending one measurement per hour

Example 2. Example 2

A device is configured the following way:

- Message encoding: SenML/CBOR/M-Bus
- Auto upload order: FIFO
- Measurement interval: 60 minutes
- Transmit interval: 60 minutes
- Transmit offset: 15 minutes
- Transmit delay: 30 minutes
- Maximum uploads per transmission: 4
- Upload maximum age 72h
- Device max payload size: 12 (readouts per telegram)

A network issue caused the module to be offline for 5 days, while still reading and storing measurement data. When the device manages to go online the following scenario takes place.

- The device will start by transmitting measurement data that is 3 days old (FIFO order)
- The device will send 4 measurement telegrams per hour, at a randomly chosen time between minute 15 and 45
- Each telegram contains 12 meter readouts, totaling 4 x 12 = 48 readouts per transmission
- The device will take approximately 2 hours to “catch up” and start sending one measurement per hour

5.2. Measurement mode

CMi6160 can be run in two different modes, meter-controlled or module-controlled. The choice of mode will give different options on what data to send from the module. To change between the modes, see LwM2M Object '**Elvaco MCM Config**' (Object ID 33906, resource 68). In meter-controlled mode, the payload is set depending on

the settings in the meter. When using module-controlled payload, settings controlled by the module will dictate what data are sent.

5.2.1. Meter-controlled: Transparent

If using mode meter-controlled, the customer-defined telegram, as specified in the Diehl SHARKY or SCYLAR meter, is sent transparently without modification. Typically, this telegram is preconfigured in the meter at delivery but can also be adjusted in the field. When using this mode, message encoding is automatically set to SenML/CBOR to accommodate larger customer-defined telegrams. Since the protocol used in Diehl SHARKY and SCYLAR is M-Bus, the payload delivered in the SenML/CBOR packages will also be in M-Bus format

5.2.2. Module-controlled: Preset

In the module-controlled mode, the data sent depends on the selected message format. CMi6160 has two different message formats to choose from, *Standard* and *Tariff*. See section Message formats [48] for a complete list of included records in each format.

5.3. Meter alarm monitor

The built-in meter alarm monitor lets the module react on alarms generated by the meter. Depending on the needs in field, the monitoring function can be set to be more or less responsive. To avoid repeatedly sending too much alarm information, it is possible to configure a suitable hysteresis. Below table lists the parameters available to make the alarm monitor act as needed. An alarm message is sent both when a meter alarm is set (discovered in the meter), and when it is reset (cleared from the meter, or having the module alarm state reset).

Table 3. Key settings for handling the meter alarm monitor

LwM2M resource	Setting	Description
33906/./0	Meter Readout Interval	**Interval in minutes**. Defines how often the module reads the meter registers. Ultimately, this setting determines how quickly an error appearing in the meter can be detected by the module. A shorter interval means higher responsiveness. The lowest practical interval that can be achieved is 5 minutes.
33906/./61	Alarm functionality enable bitmask	**Bitmask**. Defining which meter errors the module should monitor and react to. The table below describes the meter errors that can be monitored and how the bitmask should be configured to achieve this.
33906/./64	Alarm auto reset mask	**Bitmask**. Defining which meter errors should be automatically reset if the error in the meter is fixed (can e.g. be repaired via an on-site visit, or being a transient error). If an error is reset in the module, a set alarm message can be triggered again if appearing in the meter.
33906/./65	Alarm hysteresis period	**Hysteresis in minutes**. Hysteresis in minutes. Defines how long an error must be present in the meter before the module sends an alarm message. The experienced hysteresis period is also influenced by the Meter Readout Interval setting, as this determines how quickly the module can react. The same hysteresis applies for resetting an alarm.
33906/./63	Alarm mask reset period	**Interval in minutes**. This setting gives an opportunity to periodically resetting the alarm bitmask. Resetting the alarm mask allows alarms that have already been triggered and sent to be re-sent. This is typically used to resend alarms as reminders in the receiving system. A longer interval (such as monthly or yearly) is usually recommended for this purpose.
33906/./62	Alarm reset bitmask	**Bitmask**. Allowing manual reset of alarm mask, regardless of Alarm mask reset period.
33906/./66	Alarm transmit max delay	**Delay in minutes**. Maximum delay before an alarm is sent after being triggered.
33908/./4	Extended Error codes	Possibility to read the current error bitmask. The reply will be based on the latest read error codes from the meter, i.e. reading the resource will not trigger a new readout of the meter. When reading this resource, the response will reflect the current meter errors. This means it can indicate that an alarm is set in the meter without have being sent if the criteria for triggering an alarm message have not been met.
33906/./67	Alarm Topic	If using MQTT-SN, this parameter defines to which topic the alarm should be posted.

Note
The module never resets actual alarms in the meter. When referring to alarm resets in the context of the module, it only means resetting its own alarm state.

The alarm mask is not retained after a reboot, meaning alarms can be triggered again.

5.3.1. Transmission of meter alarms

Meter alarms can be transmitted using either MQTT-SN or LwM2M Send. Regardless of the chosen transport protocol, the content of the transmitted data remains the same. When an alarm is triggered, the message follows the selected format, with Extended Error Codes appended to it. If there are any unsent meter data available in the module, they will be transmitted simultaneously with the alarm.

Meter alarm transmission using MQTT-SN

When using MQTT-SN, alarm messages are published on a separate topic, apart from the ordinary meter data delivery. The alarm topic is configurable via LwM2M resource `33906/./67`.

Meter alarm transmission using LwM2M Send

When using LwM2M Send, the alarm messages are published on an own instance of the meter data object `33911`.

5.3.2. Managing the alarm bitmasks

Each bit in the alarm mask is directly mapped to a meter error. The different alarm masks used for controlling the alarm monitor have the same mapping. Below sections describes how each bitmask should be managed and interpreted.

Alarm functionality enable bitmask

Monitoring an alarm is enabled by setting the bit to high (1), while keeping the alarm silent (i.e., preventing alarm messages from being triggered) is done by setting the bit to low (0). Below table gives some examples on how the bitmask can be set to achieve different behaviour.

Table 4. Examples of Alarm functionality enable bitmasks

Binary	Hex	Interpretation
0b1111 1111 1111 1111	0xFFFF	All alarms are monitored
0b0	0x0	No alarms are monitored.
0b0001 1100 0001	0x1C1	Alarm bits 0, 6, 7, and 8 are monitored.

Alarm auto reset mask

In practical means, the Alarm auto reset mask makes it possible to re-trigger on a meter alarm that has been reset (e.g. due to natural reset, i.e. the meter has healed or been repaired, or thanks to a manual reset). If not enabling any alarms in the Alarm auto reset mask, a specific alarm will only be sent maximum twice during the lifetime of the module (once while set, and once while possibly reset). Choosing what alarms that automatically should be reset is handled in the same way as the alarm functionality enable bitmask.

Table 5. Examples of Alarm auto reset mask and Alarm reset bitmasks

Binary	Hex	Interpretation
0b1111 1111 1111 1111	0xFFFF	All alarms in the alarm mask are reset.
0b0	0x0	No alarms in the alarm mask are reset (not interesting wanting to manually resetting the alarm mask).
0b0001 1100 0001	0x1C1	Alarm bits 0, 6, 7, and 8 are reset once the hysteresis criteria is fulfilled.

Alarm reset bitmask

A manual reset can take place any point in time, allowing the module to trigger on meter alarms again. Choosing what alarms that should be manually reset is handled in the same way as the alarm functionality enable bitmask.

Table 6. Alarm reset bitmasks

Binary	Hex	Interpretation
0b1111 1111 1111 1111	0xFFFF	All alarms in the alarm mask are reset.
0b0	0x0	No alarms in the alarm mask are reset (not interesting wanting to manually resetting the alarm mask).
0b0001 1100 0001	0x1C1	Alarm bits 0, 6, 7, and 8 are reset.

5.3.3. Meter error interpretation

The table below specifies available meter errors and how they are mapped to the bitmask.

Table 7. Meter error interpretation

Error meter display	Alarm mask bit (MSB first)	Error description
C-1	0	Basic parameter error in flash or RAM
E-8	1	No primary voltage (only if mains unit used); powered by back-up battery
E-4	2	Hardware error in ultrasonic measurement, e.g. short-circuit in ultrasonic transducer, or ultrasonic transducer defective
E-1	3	Temperature measurement error, e.g. sensor break, sensor short-circuit, or temperature range exceeded [-9.9 °C … 190 °C]
E-7	4	No meaningful ultrasonic receive signal, e.g. air in the measuring path
E-9	5	Warning: battery nearly exhausted
E-3**	6	Temperature sensors reversed in hot and cold lines
E-6**	7	Wrong direction of flow, e.g. flow sensor incorrectly installed
E-B*	8	Leakage: leakage detected in energy meter
E-C*	9	Leakage: leakage pulse input 1
E-D*	10	Leakage: leakage pulse input 2
E-A*	11	Leakage: pipe break detected
E-5	12	Reading too frequently M-Bus communication not possible for short time
N/A	13..31	Reserved, not used

* Optional, ** Application-dependent

5.3.4. Meter alarm payload example

Using message format *Tariff*, a typical payload can look like:

```
04139C2E0000023B0000325A0000325E000084100600000000008420FD320000000001FD1750
```

If the same payload is sent as an alarm message, the extended error codes will be appended to the payload:

```
04139C2E0000023B0000325A0000325E000084100600000000008420FD320000000001FD1750 02FD180800
```

The extended meter codes are identified by the M-Bus DIF/VIF `2FD18`, *Error Mask*, and is here followed by the two error bytes `0x0800`. Since M-Bus is encoded using little endian, the resulting complete error mask is `0b0000 0000 0000 1000`. This means error bit 3 is set, indicating there is some issue with the temperature measurements (see meter error interpretation table).

5.4. Module system log

Throughout the module's lifetime, various events may be of interest to log, for example, to analyze its behavior and overall health status. The built-in system log can be configured to store and report such events based on their criticality.

There are two settings available to control the behavior of the system log: *Syslog storage level* and *Syslog auto-upload level*. The storage level determines which events should be logged and stored in the module, while the auto-upload level defines which events should be sent from the module, depending on their criticality. This allows

storing critical events without necessarily sending them, helping to reduce data traffic or power consumption. New system log entries, with appropriate logging level, will be sent together with the subsequent meter data delivery.

> **Note**
> For a complete reference to LwM2M resources related to the module system log, see LwM2M Object '**Elvaco Syslog Config**' (Object ID: 33918).

Below tables list what logging levels there are, and the implemented events that can be logged.

Table 8. Supported logging levels

Log level	Log level name	Description	Remark
1	info	Informational, normal events	
2	notice	Normal operation notifications	No corresponding elements having this log level have been implemented. Can be added in future releases.
3	warning	Warning logs	
4	error	Error logs	No corresponding elements having this log level have been implemented. Can be added in future releases.
5	critical	Critical error logs	No corresponding elements having this log level have been implemented. Can be added in future releases.
Varies	*Varies*	Log level determined based on the content of the log entry.	Varying log level used e.g. for configuration changes. Some changes might be more or less sensitive. More sensitive log entries will get a higher logging level.

Table 9. Supported log elements and corresponding log level

Log level	Entry	Log data
info	Product activation	Reason for activation (button, NFC, Activation at flow etc).
info	Successful timesync	-
info	Mismatch in network parameters	-
info	Startup cause	Reason/cause for startup, for example watchdog, brownout, soft-reset (button/NFC/LwM2M/Console/…)
info	Successful meter readouts after retry	Number of retries needed.
info	Incompatible meter	Reason and meter version.
warning	Unsuccessful time sync	• Failed time synchronizations • Large time syncs (logged if time is adjusted more than expected)
warning	Low battery	Warnings and shutdowns
warning	Successful “connections” to Bootstrap, DB, MDM	-
warning	Failed “connections” to Network, Bootstrap, DM, MDM	Network rejects, failed DTLS handshakes etc
warning	Changed SIM	Change in ICCID
warning	Successful FW Upgrade	• Source • Target (main, modem)
warning	Unsuccessful FW Upgrade	• Source • Target (main, modem) • Cause of failure
warning	Config reset with result	• Source (NFC, console, DM) • Type (factory, stored) • Successful/unsuccessful
warning	Unsuccessful NFC writes	Reason for failure: • Lack of PAK • No PAK • Malformed message • Other
warning	Unsuccessful LwM2M config writes	-

Log level	Entry	Log data
warning	Failed staged settings	• Log reason for failure Bootstrap server URI • Radio bands • APN mode and manual APN • Manual PLMN • Roam home PLMN search
warning	Unsuccessful meter readout	Cause of issue.
warning	Deactivation by meter	Reason an target for deactivation.
varies	Configuration changes	Log level is dependent on what configuration change done. Source for change (NFC, DM) logged.

5.4.1. Transmission of system log

The module system log can be transmitted using either MQTT-SN or LwM2M Send. Which one is used depends on the chosen transport protocol for the meter data. Regardless of the choice, the content of the transmitted data remains the same. When a system log entry, having a system log level that is configured to be sent, is stored in the module, it will be sent together with the subsequent meter data transmission. All unsent log entries that are supposed to be sent will be included.

System log transmission using MQTT-SN

When using MQTT-SN, system log entries are sent on a separate topic, apart from the ordinary meter data. The system log topic is configurable via LwM2M resource `33918/./0`.

System log transmission using LwM2M Send

When using LwM2M Send, the system log entries are published on an own instance of the meter data object `33911`.

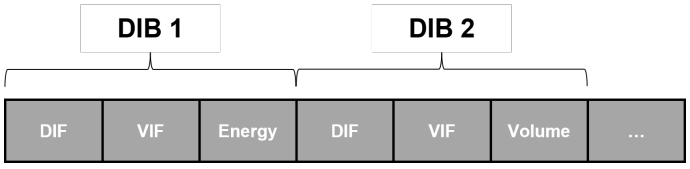
5.5. Message encoding

The product has three options when it comes to message encoding:

- M-Bus
- JSON
- SenML/CBOR

5.5.1. M-Bus

If using M-Bus as message encoding technique, data will be divided into Data Information Blocks (DIB) that include Data information field (DIF code), Value information field (VIF code) and a data field (DATA) where the actual payload is stored (illustrated in the following image).



DIB structure

The table below provides a detailed examples of how data is encoded when using message encoding M-Bus.

Table 10. Payload, M-Bus encoded message

DIB	Field	Size	Data type	Description
1	Date/time	6 bytes	INT32	Meter date and time (YY-MM-DD HH:MM) Mapped to OBIS 9.36 046Dxxxxxxxx Bit 31-28 = Year-high* Bit 27-24 = Month Bit 23-21 = Year-low* Bit 20-16 = Day Bit 15 = Summer time flag** Bit 14-13 = Century Bit 12-8 = Hour Bit 7 = Error flag Bit 6 = Reserved for future use*** Bit 5-0 = Minute *The year is read by combining the yearhigh and year-low field. For example, year-high = 0010 and year-low = 010 => year = 0010010 **0 = standard time, 1= daylight-saving time ***0 = timestamp is valid, 1 = timestamp is **not** valid
2	Meter ID	6 bytes	According to M-Bus EN13757- 3 identification field	Meter ID 0C78xxxxxxxx
3	Energy	6-7 bytes	INT32	Energy consumption (Wh, J) 0406xxxxxxxx = xxxxxxxx * 0.001 MWh (kWh) 0407xxxxxxxx = xxxxxxxx * 0.01 MWh 04FB00xxxxxxxx = xxxxxxxx * 0.1 MWh 04FB01xxxxxxxx = xxxxxxxx MWh 040Exxxxxxxx = xxxxxxxx * 0.001 GJ (MJ) 040Fxxxxxxxx = xxxxxxxx * 0.01 GJ 04FB08xxxxxxxx = xxxxxxxx * 0.1 GJ 04FB09xxxxxxxx = xxxxxxxx GJ

DIB	Field	Size	Data type	Description
4	Volume	6 bytes	INT32	Volume (m^3) 0413xxxxxxxx = xxxxxxxx * 0.001 m^3 0414xxxxxxxx = xxxxxxxx * 0.01 m^3 0415xxxxxxxx = xxxxxxxx * 0.1 m^3 0416xxxxxxxx = xxxxxxxx m^3
5	Power	4 bytes	INT16	Power (W) 022Bxxxxxx = xxxxxx * 0.001 kW (W) 022Cxxxxxx = xxxxxx * 0.01 kW 022Dxxxxxx = xxxxxx * 0.1 kW 022Exxxxxx = xxxxxx kW
6	Flow	4 bytes	INT16	Flow (m^3/h) 023Bxxxxxx = xxxxxx * 0.001 m^3/h 023Cxxxxxx = xxxxxx * 0.01 m^3/h 023Dxxxxxx = xxxxxx * 0.1 m^3/h 023Exxxxxx = xxxxxx m^3/h
7	Fw temp	4 bytes	INT16	Forward temperature (°C) 025Axxxx = xxxx * 0.1 °C 025Bxxxx = xxxx * °C
8	Rt temp	4 bytes	INT16	Return temperature (°C) 025Exxxx = xxxx * 0.1 °C 025Fxxxx = xxxx °C
9	Error flags	5 bytes	INT16	Error and warning flags 02FD17xxxx For further information about Error flags, refer to the latest meter's manual
10	Tariff 1 Energy	7 bytes	INT32	Tariff 1 Energy consumption (Wh, J) 841003xxxxxxxx = xxxxxxxx Wh 841003xxxxxxxx = xxxxxxxx * 10 Wh 841003xxxxxxxx = xxxxxxxx * 100 Wh 841003xxxxxxxx = xxxxxxxx kWh 841003xxxxxxxx = xxxxxxxx *10 kWh 841003xxxxxxxx = xxxxxxxx MJ 841003xxxxxxxx = xxxxxxxx * 10 MJ

DIB	Field	Size	Data type	Description
11	Tariff 2 Energy	7 bytes	INT32	Tariff 2 Energy consumption (Wh, J) 842003xxxxxxxx = xxxxxxxx Wh 842003xxxxxxxx = xxxxxxxx * 10 Wh 842003xxxxxxxx = xxxxxxxx * 100 Wh 842003xxxxxxxx = xxxxxxxx kWh 842003xxxxxxxx = xxxxxxxx *10 kWh 842003xxxxxxxx = xxxxxxxx MJ 842003xxxxxxxx = xxxxxxxx * 10 MJ
12	Tariff 3 Energy	7 bytes	INT32	Tariff 3 Energy consumption (Wh, J) 843003xxxxxxxx = xxxxxxxx Wh 843003xxxxxxxx = xxxxxxxx * 10 Wh 843003xxxxxxxx = xxxxxxxx * 100 Wh 843003xxxxxxxx = xxxxxxxx kWh 843003xxxxxxxx = xxxxxxxx *10 kWh 843003xxxxxxxx = xxxxxxxx MJ 843003xxxxxxxx = xxxxxxxx * 10 MJ
13	Missing time	6 bytes	INT32	3C22xxxxxxxx = xxxxxxxx hours 3C23xxxxxxxx = xxxxxxxx days

5.5.2. JSON

When using JSON message encoding, messages sent consist of an object with a list of key – value pairs. Example of the names of each value type and unit are presented in the table below. The values are encoded as numbers or strings and the units are encoded as strings. JSON offers data encoding that is human-readable, at the expense of not being equally compact as e.g. SenML/CBOR.

Table 11. Payload, JSON encoded message

Field	JSON key
Meter ID	ID
Meter date / time	TS
Energy	E
Energy unit	U
Volume	V
Volume unit	VU
Power	P
Power unit	PU
Flow	F
Flow unit	FU
Forward temperature	FT
Forward temperature unit	TU
Return temperature	RT
Return temperature unit	RU
Error flags	EF
Tariff 1 Energy	T1
Tariff 1 Energy unit	U1
Tariff 2 Energy	T2
Tariff 2 Energy unit	U2

Field	JSON key
Tariff 3 Energy	T3
Tariff 3 Energy unit	U3
Missing time	MT
Missing time unit	MU
Extended information	I

Example payload, JSON:

{"TS":"2025-08-27T07:36:01Z "," ID":81493511, "E":0, "U":"kWh", "V":0, "VU":"m3", "P":5012, "PU":W, "F":212, "FU":l/h, "FT":30.1, "TU":"C", "RT":22.5, "RU": "C", "I":"0x0010", "EF":"0x70"}

5.5.3. SenML/CBOR

For battery-powered devices it might be necessary to send several measurements in the same UDP frame to save energy. In order to achieve this, SenML (Sensor Measurement Lists) + CBOR (Concise Binary Object Representation) is used to define a measurement list.

The idea is to send a list of measurements, where the first entry contains the base time for all the readouts (which only need to specify an offset) and the meter id shared by all readouts. The other records in the list may contain fewer readout fields to save space. The format allows sending all the data for every readout, in which case the save (in terms of bytes) is smaller and lies in that fewer telegrams are sent, some data needs not be transferred for every reading (like meter-id) and timestamps can be handled more efficiently. SenML/CBOR also provides one way to structure lists of readings in an efficient manner.

Elvaco uses SenML/CBOR/M-Bus data representation for transferring meter data in a compact and self-describing manner. The data being transferred is referred to as a pack, containing one record per measurement.

> **Note**
> SenML, CBOR and M-Bus are separate standards, this section describes how products can use these three in conjunction for representing multiple measurement values in a compact format suitable for radio transmission over for instance NB-IoT.

5.5.3.1. Structure of SenML pack

Meter readout data is sent as SenML, i.e., a list (aka array) of readout values (records), encoded using CBOR. Each record is a map of key/value pairs using SenML.

> **Note**
> Each product that uses the SenML/CBOR format shall follow the requirements below. In addition, it shall specify the exact contents of the data values included, meter ID format etc. Thus, this specification alone is not sufficient for building a parser for a specific product.

Base Time

Base time is used to set a reference time.

- Timestamps are always encoded according to SenML (i.e., UNIX time)(SenML label -1 “Base time”, SenML definition of Time field)
- This value **must** be included in the first record of the pack
- All other values have a time value that is added to the base time to define the exact time of the readout

Base Name

Base name is used to represent the MeterID (Meter identification in M-Bus) for products that deliver measurement data for one meter.

- If base name is used

        - This value MUST be included in the first record of the pack
        - This is represented as a string array (CBOR Major Type 3 - SenML label -2 “Base name”)
            - The product shall specify the exact format for this field, as it may vary depending on what type of “meter” is used. For an M-Bus format it is typically the M-Bus data without DIF/VIF.
        - No name is set for remaining meter readout values, only values belonging to a single meter can be represented in one pack.
    - If readouts to be sent contain different MeterIDs (module moved between meters for instance) they must NOT be sent in the same SenML pack
        - All values within a SenML pack MUST be for the same MeterID
        - The data values SHALL NOT have a name value further specifying the name for each value.
    - Base name SHALL NOT be used for SenML packs that have data for multiple meters. In such cases each record shall contain the MeterID of the meter.

Data values

- The actual values from the meter can be encoded using multiple methods, such as M-Bus.
- The first record can also contain a data value field containing more information than the remaining records in the pack. This is to include more information for the first reading and then only a subset of values for the remaining records to save space. (SenML label 8 - “Data value”)

Other values

- (Base) Unit is not used, since the unit is specified by the M-Bus data
- An “Encoder Version field” is used in a separate record to define the type and version of the encoded payload data.

5.5.3.2. Additional records

All records in the SenML pack are expected to contain measurement values. If there is a need for transmitting additional information in the same pack additional records can be added. For such records the name field shall be used by defining a name of at least a single character. In SenML the base name and the name fields are appended to result in the final record name.

The name shall contain at least one character outside [A-Fa-f0-9] which signifies non-hexadecimal representation, since meter-id is typically decimal/hexadecimal, and this makes it easier to check the record name for validity. If a parser finds a record with a name field like described above that it does not recognize it shall ignore the record.

The following additional records are currently used:

Record	Name field	Comment
Encoder Type & Version	”V”	This field allows defining versions for the contents of the measurement field.
Extended Error Information	"I"	This record allows for sending extended information on a measurement, such as error information.

Encoder Type & Versioning

The following table defines allowed encoder types and versions. The information is sent in a special record “Encoder Version field”.

- This field encapsulates both the encoding of the data and versioning
- It contains no timestamp
- It is encoded as a SenML Value
- It has a *Name* field with the single letter “V”
- If, when parsing, an invalid version is encountered the parsing shall stop with an error
- The value shall be interpreted as an UINT16
    - The first byte is the encoder type and the second is the encoder version, both interpreted as UINT8. **Example:** value 0x0102 means Encoder type 0x01 and Encoder version 0x02.
    - Defined valid encoder types and versions are found in the table below

    - Size of whole record is maximum 7 bytes
- If record is excluded, encoder type is 0 and encoder version is 0

Encoder Type	Encoder Version	Data	Comment
0 (M-Bus)	0	0x0000	M-Bus encoding of payload data. Each data record contains all DIF/VIF/Values according to M-Bus. Note that M-Bus uses LSB first byte order for the data and it shall be preserved here as well.
1 (Gateway)	0	0x0100	Gateway for meter data, i.e., data for multiple meters are contained in a single pack. Payload can be of various types depending on the interface used. Example of payload format is unencrypted or encrypted M-Bus data, with additional meta data. This encoder type is currently not relevant for Elvaco meter connectivity products.
2 (Syslog)	0	0x0200	Syslog data message encoded as an ASCII hex string. Key 2 indicates Log level. See section covering Module system log for details on the values for each level.
3 (Transparent M-Bus)	0	0x0300	Transparent M-Bus data. The payload contains a full (wired) M-Bus telegram, including both header and CRC. This encoder type is relevant when using the measurement mode meter-controlled Transparent.

Extended Information

This record is used to send extended information that cannot be sent as part of the normal meter readout data. Typical use case is additional error codes from the meter/device.

- The format of the extended information is specific for the selected encoder. See table below.
- The detailed interpretation of the information is product specific. For instance error codes for a meter is specific for a particular meter, even though they are encoded using M-Bus.
- It is encoded as a SenML data Value
- It has a *Name* field “I”
- It shall contain a timestamp
    - The timestamp can be the same as another meter data record and then these two records shall be considered having been taken at the same time.
- The extended information is an optional record

Encoder type	Encoder version	Format of extended error information
0 (M-Bus)	0	M-Bus encoding of payload data. Each data record contains all DIF/VIF/Values according to M-Bus. Note that M-Bus uses LSB first byte order for the data and it shall be preserved here as well.
1 (Gateway)	0	Not relevant for Elvaco meter connectivity modules.
2 (Syslog)	0	Not defined yet
3 (Transparent M-Bus)	0	M-Bus encoding of payload data. Each data record contains all DIF/VIF/Values according to M-Bus. Note that M-Bus uses LSB first byte order for the data and it shall be preserved here as well.

5.5.3.3. Example Transparent M-Bus and extended information

Below example shows a data package with two records, the first with Transparent M-Bus data, and the second with extended information, indicated by "I".

```
1  83                                    # array(3)
2
3  ** Record for defining encoder and version **
4
```

```
A2                      # map(2)
   00                   # unsigned(0)   ## Key 1: Name
   61                   # text(1)       ## Value 1:
      56                # "V"           ## "V" = version
   02                   # unsigned(2)   ## Key 2: Value
   19 0300              # unsigned(768) ## Value 2: enc=3, ver=0

** first record of the pack **

A2                      # map(2)
   22                   # negative(2)          ## Key 1: Base Time
   C1                   # tag(1)               ## Value 1:
      1A 6835B149       # unsigned(1748349257) ## 2025-05-27T12:28:33+00:00
   08                   # unsigned(8)          ## Key 2: Data value
   58 36                # bytes(54)            ## Value 2: M-Bus telegram
      6830306808007274
      031961A51140048B
      0000000C06080000  # Payload data, line breaks added for readbility
      00046D220C3B350C
      13377400008C10FD
      32000000008C2013
      000000006D16

** second record of the pack, containing extended information **

A3                      # map(3)
   00                   # unsigned(0)          ## Key 1: Name
   61                   # text(1)              ## Value 1:
      49                # "I"                  ## "I" = extended information
   08                   # unsigned(8)          ## Key 2: Data value
   45                   # bytes(5)             ## Value 2: M-Bus data
      02FD187000        # "\u0002\xFD\u0018p\u0000"
   06                   # unsigned(6)          ## Key 3: Timestamp
   00                   # unsigned(0)          ## Value 3: Relative time 0
```

5.5.3.4. Validators

At https://cbor.me/, there's a Validator available for CBOR. Note that it does not understand SenML or M-Bus.

> **Note**
> A small bug has been identified in the hex interpretation of negative numbers in the validator. Please have this in mind if using the validator.

5.5.3.5. Configuration

SenML/CBOR is to be considered a message encoding. It defines how the messages are encoded, but not the actual contents of the messages (which fields from the meter are included). SenML/CBOR/M-Bus is one such encoding, but there could be several based on this SenML/CBOR specification and the encoder version field above defines exactly which type and version is used.

The contents of the message are defined by the message format. The message format sets which fields are to be included in both the first and the subsequent records of the SenML pack.

The number of records included in a pack is set by the readout and transmit intervals. See Scheduling Readouts for more details. If the readout interval is 120 minutes and the transmit interval is 1440 minutes 12 readouts in total will be included.

5.5.3.6. Message size restrictions

Each product may have different maximum payload sizes in a single telegram. Also depending on configuration (DTLS or not for instance) the net payload size may vary. Therefore, the device shall “fill up” as many telegrams as required to send the data. It is for the user to define a configuration that gives a reasonable tradeoff between power consumption (send fewer telegrams) and functional requirements (much data is sent).

If a device is configured using a Message Format and many readouts the data may not fit in a single telegram. In such cases multiple telegrams shall be sent and each telegram shall be fully self-described, i.e., contain Meter ID, timestamps etc.

Example 1

Parameter	Value
Readout interval	60
Transmit interval	1440 (daily)
Message encoding	SenML/CBOR/M-Bus version 0
Message format	Standard
Max transmissions per day	3

This example results in the transmission of one message per day, containing 24 readings, all with the contents defined in the Standard message format. Data is encoded using SenML/CBOR/M-Bus. Maximum 3 unsent such messages are sent each time (if for some reason the messages were not sent “last time”). So maximum transmitted messages per day is 3 (containing 3x24=72 readings, covering 3 days)

Example 2

Parameter	Value
Readout interval	120
Transmit interval	720
Message encoding	SenML/CBOR/M-Bus version 0
Message format	Tariff
Max transmissions per day	2

This example results in the transmission of one message every 12h, containing 6 readings, all with the content defined in the Tariff message format. Data is encoded using SenML/CBOR/M-Bus. Maximum 2 unsent such messages are sent each time (if for some reason the messages were not sent “last time”), so maximum transmitted messages per day is 4 (containing 4x6=24 readings, covering 2 days).

5.6. Security and access control

For local configuration, the device has a NFC interface, utilized in the Elvaco OTC app. To prevent unauthorized access to the module via the NFC interface, the product has a configuration lock feature. When the configuration lock has been enabled, the device-specific Product Access Key (PAK) is needed to configure the device. The PAK keys are managed in a secure way using Elvaco’s OTC solution which includes the mobile application for configuration. It's also possible to completely shut off the NFC functionality. This will prevent any user to do local configuration over NFC. The table below describes the available settings that control the NFC interface and the NFC configuration lock.

> **Note**
> To get access to a device having the NFC configuration lock enabled, log in to the Elvaco OTC app. If the device is claimed your organization, it will be possible to configure it.

Table 12. Settings for controlling the local access to the device

LwM2M resoruce	Setting	Description
33906/./4	NFC Enabled	**Boolean**. Turns on or off the NFC interface of the device. If disabled, it's not possible to do any local configuration to the device.

LwM2M resoruce	Setting	Description
33906/./5	NFC Config-locked	**Boolean**. Turn the NFC configuration lock on or off. If enabled, it will not be possible to do any local configuration to the device, unless having the PAK at hands.

5.6.1. Automatic NFC configuration lock

To minimize the risk of not enabling the NFC configuration lock when the device is installed in field, an Automatic NFC configuration lock is enabled by default. Once enabled, the NFC configuration lock will automatically be enabled after a time has elapsed (15 min by default). The timer starts when the device is activated. The feature also gives the possibility for installers, that necessarily don't own the device (has possession of the PAK), can apply any configuration during installation.

Tip
This feature can be utilized for temporarily disabling the NFC configuration lock, e.g. if the device needs to be visited in field by an operator without the PAK. Disable the NFC lock, let the temporailiy authorized operator make local configuration, let the automatic configuration lock comfortably lock the device.

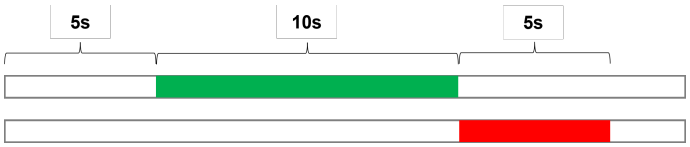
Table 13. Settings for controlling the automatic NFC configuration lock

LwM2M resource	Setting	Description
33906/./	Enable NFC automatic configuration Lock	**Boolean**. Turn the automatic NFC configuration lock On or Off. If turned on, the configuration lock will be enabled after the Automatic lock timeout has elapsed. The timer is started once the device has been activated.
33906/./	Automatic NFC configuration lock timeout	**Time in minutes**. Sets the time in minutes before the configuration lock will be enabled upon activation (requires the NFC automatic configuration lock to be enabled).

5.7. Reset procedures

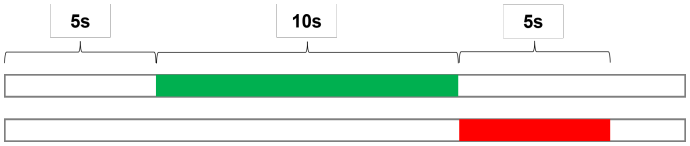
5.7.1. Rebooting the module

1. Press and hold the push button for 5-15 seconds.
2. Release the button when the green LED is lit.



5.7.2. Switching off the module

1. Press and hold the push button for 15-20 seconds.
2. Release the button when the red LED is lit.



5.7.3. Performing factory reset

By performing a factory reset, the device reverts its settings to factory default. It will also delete all meter data and syslog entries stored in the module. Factory reset can be done either via the Elvaco OTC app, or remotely over LwM2M (resource `3/./5`).

Note
Performing a factory reset restores all network and server parameters to their factory defaults. After the reset, the device attempts to bootstrap to Elvaco's default server using automatic APN detection.

If the SIM card is configured for a private APN only, the device may still register to the NB-IoT network but will not be able to reach Elvaco's bootstrap server. The device will be logically unreachable, and a local re-configuration will be necessary.

6. Configuration options

6.1. Introduction

The module is a versatile product that can be configured for a wide range of use cases and implementations. Configuration can be made either remotely, via a device management system using LwM2M, or via the Elvaco OTC App, using the NFC interface of the device. Below sections list all settings and operations possible for the different configuration interfaces.

6.2. Configuration options via Elvaco OTC App

Using the Elvaco OTC App for configuration is an efficient way of commissioning an Elvaco module. For repeated commission of devices with the same set of settings, it is recommended storing a profile holding those settings. Utilizing stored profiles will speed up the commissioning phase and reduce the risk of faulty settings.

> **Note**
> Make sure to have the latest version of the Elvaco OTC App installed before starting the process. To be able to apply changes to a locked device, the user must be logged in to the Elvaco EVO account to which the device has been appointed.

6.2.1. Elvaco OTC configuration options: Device

Below set of configuration options governs general the general behavior, such as opening or locking the CMi6160. It also includes a function to revert its settings to factory default.

Table 14. Elvaco OTC configuration options: Device

Name	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
Power mode	Inactive, Active, Stand-by	Inactive	RW	R	Activation status of the module. If not active, no actions will be taken by the module. The automatic activation feature is available in FW >=1.3.2.
Configuration Lock	Open, Locked	Open	RW	R	Locks the module to prevent unauthorized access.
Synchronize meter time	Time of mobile phone	N/A	W	N/A	Time of mobile phone used to synchronize meter clock.
UTC offset	-720 - 720	0	RW	R	UTC offset of the meter (in minutes).
Max meter retries	0-255	255	RW	R	Maximum amount of quick retries when failing communicating with a meter.
Power source	Battery, PSU	Battery	RW	R	Detection and correction of the power source of the meter. Only used if hardware does not support power source detection.
NFC Auto Lock Enable	Enabled, Disabled	Enabled	RW	R	Enables or disables automatic configuration lock to be turned on (once the device has been activated and the NFC Auto Lock Enable Timeout has elapse). Once the configuration lock is turned on, the PAK must be present to interact with the product.
NFC Auto Lock Enable Timeout	1-10080	15	RW	R	Time in minutes for the configuration lock to be enabled (requires the NFC Auto Lock Enable to be enabled).

Name	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
Factory reset	N/A	N/A	E	N/A	Resets the CMi6160 to factory settings (default factory setting for power mode is Active). **Note** Doing a factory reset will delete all meter data, alarms, credentials, and system log entries from the device.

6.2.2. Elvaco OTC configuration options: LwM2M

Below settings are related to the connection to a LwM2M server.

Table 15. Elvaco OTC configuration options: LwM2M

Name	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
BS URI	A valid Bootstrap URI	coaps://84.19.147.226:5694	RW	R	Bootstrap URI of the bootstrap server the module will connect to upon activation.
Bootstrap security	DTLS, No security	DTLS	RW	R	Sets the way data sent from the module is encrypted (also given by defined BS URI).
LwM2M trigger registration update	N/A	N/A	E	N/A	Upon execution, the device will perform a registration update with the LwM2M Device Management server.
Force bootstrap	N/A	N/A	E	N/A	Force the device to reboot and bootstrap. NB! This action will erase all LwM2M server URIs.
LwM2M queue mode	Enabled, Disabled	Enabled	RW	R	For battery operated devices, it is strongly recommended to have LwM2M queue mode enabled.
CoAP ack timeout	1..240	60	RW	R	LwM2M CoAP timeout value in seconds, See LwM2M specification for more info
CoAP max retransmit	1..6	1	RW	R	LwM2M max retransmit value. See LwM2M specification for more info
DTLS min timeout	1..3600	60	RW	R	The first timeout in seconds used when transmitting packets via DTLS for LwM2M.
DTLS max timeout	1..3600	90	RW	R	The last timeout in seconds used when transmitting packets via DTLS for LwM2M.
Communication retry count	0..10	1	RW	R	Number of connection attempts to a LwM2M server before marking a connection failed.
Communication retry delay	1..3600	3600	RW	R	Delay in seconds between connection attempts to LwM2M servers
Sequence retry count	1..10	2	RW	R	Number of connection sequence attempts to LwM2M servers.
Sequence retry delay	1..86400	86400	RW	R	Delay in seconds between connection sequence attempts to LwM2M servers.

Name	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
Sequence back-off	Must be formatted as min-max, min-max,..	0-5,60-120,1300 -1600	RW	R	Delay ranges in minutes to wait between full LwM2M sequence connection attempts. I.e. If both bootstrapping and connection to device management fails consecutively, delay progressively until connectivity can be restored.

6.2.3. Elvaco OTC configuration options: Cellular

The settings available on the Cellular tab allows tweaking how the device should behave in relation to the NB-IoT network. Typically, these settings should not be changed.

Table 16. Elvaco OTC configuration options: Cellular

Name	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
APN mode	Auto, Manual	Auto	RW	R	Sets how APN settings is implemented in the module.
APN	Name of APN	N/A	RW	R	APN to use if APN mode is manual
PLMN	0..999999	N/A	RW	R	PLMN of the network provider set by the user. Comprises of MCC and MNC codes (e.g. MCC = 240 and MNC = 01 results in a PLMN code of 24001. PLMN is also referred to as Home Network Identity (HNI).
Radio frequency band	0,3,8,20 Several bands can be chosen by e.g. typing 0,3,8	3,8,20	RW	R	Which NB-IoT frequency band to use. Setting this will make the modem skip scanning all bands and just use the supplied one if possible. If this fails, the modem will scan all bands.
Power saving mode	Disabled, eDRX, PSM, eDRX+PSM	eDRX+PSM	RW	R	Setting for power save mode.
T3324 timer	1..11160	120	RW	R	LTE Active Timer. This setting controls how long (in seconds) the modem will wait for network activity before entering power saving mode.
T3412 timer	1..35712000	252000	RW	R	LTE Extended TAU timer. This setting controls how long (in seconds) the modem will be in power saving mode before waking up.
eDRX mode	Automatic, Manual	Manual	RW	R	
eDRX value	0..255	19	RW	R	eDRX controls how often the device can be contacted when not in power saving mode.
Time sync source	Network, NTP, Manual	Network	RW	R	Source for setting meter clock.
Brown out delay	1..64800	21600	RW	R	The maximum delay in seconds before reconnecting after a power outage.
Search period	180..57600	21600	RW	R	Maximum network search period in seconds. After this period, the device will enter deep sleep until next connection attempt.
RAI	Disabled, MDM RAI enable, DM RAI enable, DM+MDM RAI enable	MDM RAI enable	RW	R	

6.2.4. Elvaco OTC configuration options: Communication

The settings within this block determines the behavior of the meter data sent from the CMi6160. Chosen message format determines what data that are sent, and the MDM upload protocol the protocol used for data transmission. There are also parameters for controlling how often the meter is read, and how of often it is sent (transmitted) to a receiving server over the NB-IoT network.

Table 17. Elvaco OTC configuration options: Communication

Name	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
Message format	Standard, Standard extended	Standard	RW	R	Sets the payload of the data message from the module.
Message encoding	M-Bus, JSON, SenML/CBOR	SenML/CBOR	RW	R	Sets the encoding of the payload.
Measurement mode	Transparent, Preset	Preset	RW	R	Sets whether to use any of the predefined message formats (Preset), or utilizing the customer defined telegram in the meter (Transparent).
Readout interval	1..1440	60	RW	R	Placeholder for future use, currently not implemented.
Report interval	1..1440	60	RW	R	Number of minutes between each meter data readout.
Transmit interval	1..1440	1440	RW	R	Number of minutes between each data transmission.
Transmit offset	1..1440	125	RW	R	Time before transmit window starts from transmit interval (in seconds).
Transmit delay	1..1440	45	RW	R	Time period were the transmission while be randomized (in minutes).
Max uploads per transmission	1..168	8	RW	R	The number of max packages / transmissions. This will effect the time to recover when communication link has been down.
MDM upload protocol	MQTT-SN Publish, LwM2M Send	MQTT-SN Publish	RW	R	Transport protocol used for meter data transfer
MQTT-SN Connection	Optimized, Compliant	Optimized	RW	R	Connection type used when publishing messages to the MQTT-SN broker.
MQTT-SN Topic	Valid MQTT-SN topic	N/A	RW	R	Topic used when publishing messages to the MQTT-SN broker, e.g. elvaco/#P/#E/#T/#D.
Auto-upload max. age	0..99999999	20160	RW	R	Maximum age of the resent data (in minutes).
Auto-upload order	Oldest first, Latest first (FiFo/LiFo)	Latest first	RW	R	Start with oldest or newest data when resending data.
Restart back-off	min-max,min-max,..	0-5, 720-1440	RW	R	Delay range in minutes between restarting the modem on failures.

6.2.5. Elvaco OTC configuration options: MDM

Below settings governs how the CMi6160 are related to detailed control over how meter data is transferred. It also contains the settings needed to control the System log.

Table 18. Elvaco OTC configuration options: MDM

Parameter	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
Communication timeout	1-600	60	RW	R	MQTT-SN timeout in seconds when communication actions. E. g. timeout when publishing without DTLS
Communication attempts	0-255	2	RW	R	MQTT-SN max number of retries before considering operation failed.
Max communication failures	0-255	2	RW	R	Maximum number of operation failures before considering connection lost.
DTLS min timeout	1-3600	60	RW	R	The first timeout in seconds used when transmitting packets via DTLS for MDM packages.
DTLS max timeout	1-3600	90	RW	R	The last timeout in seconds used when transmitting packets via DTLS for MDM packages.
Reconnect back-off	min-max,min-max,..	0-0, 60-60, 360-360, 1380-1380	RW	R	Delay ranges in minutes between re-starting modern on failures.
Syslog Level	Info, Notice, Warning, Error, Critical, Off	Info	RW	R	Sets what log entries to store in the module.
Syslog Auto Upload Level	Info, Notice, Warning, Error, Critical, Off	Notice	RW	R	Sets what log entries should be sent by the module, based on their log level. Note that the sending can be turned off, and then instead requested using LwM2M device management.
Syslog Auto Upload Age Limit	0-1051200	20160	RW	R	Time in minutes.

6.2.6. Elvaco OTC configuration options: Meter alarm monitor

Below settings control the meter alarm monitor in the CMi6160. Depending on the use case, it can be configured to be highly responsive or less responsive and thus more battery friendly.

Table 19. Elvaco OTC configuration options: Meter alarm monitor

Parameter	Possible values	Default value (battery)	Locked device & correct PAK or open device	Locked device & no PAK	Description
Alarm hysteresis	0-65535	0	RW	R	Sets the hysteresis in minutes. Controls how responsive the alarm monitor will be.
Enable meter alarms	List of all alarms available to monitor.	None	RW	R	From the drop-down list, individually activate the alarms of interest.
Enable alarms auto-reset	List of all alarms available to monitor.	None	RW	R	From the drop-down list, individually activate what alarms that should be automatically reset.
Alarm mask reset period	1-65535	0 (Off)	RW	R	Sets a period reset of meter alarms.
Manual alarm reset	N/A	N/A	E	N/A	Manually reset the alarm mask.
Alarm TX Delay Max	0-65535	5	RW	R	Maximum delay in minutes before a triggered alarm is sent from the module.

6.3. Remote device management via LwM2M

> **Note**
> This section is mainly targeted to system integrators with the ambition to integrate the device into their own device management system. Making practical use of the LwM2M device management possibilities presented below require having a LwM2M server up and running.

The module supports remote device management using LwM2M. This e.g. includes monitoring the device, making setting changes, and updating the firmware over the air. Below sections specify what LwM2M objects and related resources that are supported by the device.

6.3.1. OMA defined LwM2M objects

The following section list what OMA defined LwM2M objects that are supported by the module. For a complete and detailed description of each object and their resources, see Open Mobile Alliance - LwM2M Registry.

6.3.1.1. LwM2M object OMA Device

This LwM2M Object provides a range of device related information which can be queried by the LwM2M Server, and a device reboot and factory reset function.

Table 20. Object definition: OMA Device

Name	Value	Description
Name	Device	The name of the object
Object ID	3	The Object ID
LwM2M version	1.1	The LwM2M Version used
Object version	1.1	The version of this object definition

Table 21. Resource definition: Device Resources (Elvaco implementation)

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
0	Manufacturer	R	Single	No	Str			Manufacturer ("Elvaco")
1	Model Number	R	Single	No	Str			Product model ("CMi61X0")
2	Serial Number	R	Single	No	Str			DevEUI
3	Firmware Version	R	Single	No	Str			Firmware version
4	Reboot	E	Single	Yes				Reboot the LwM2M Device to restore the Device from unexpected firmware failure.
6	Available Power Sources	R	Multiple	No	Int	0..7		Power source 1: Internal battery 2: External battery 6: AC (Mains) power
7	Power Source Voltage	R	Multiple	No	Int			Present voltage for each Available Power Sources Resource Instance. The unit used for this resource is in mV.
9	Battery Level	R	Single	No	Int	0..100	%	Battery level (in %)

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
11	Error Code	R	Multiple	No	Int	0..100		Error codes, according to LwM2M 1.1
13	Current Time	RW	Single	Yes	Int	0..8		Current time
14	UTC Offset	RW	Single	No	Time			UTC Offset UTC+X (ISO 8601)
18	Hardware Version	R	Single	No	Str			Hardware version

6.3.1.2. LwM2M object OMA Connectivity Monitoring

This LwM2M Object enables monitoring of parameters related to network connectivity. In this general connectivity Object, the Resources are limited to the most general cases common to most network bearers. It is recommended to read the description, which refers to relevant standard development organizations (e.g. 3GPP, IEEE). The goal of the Connectivity Monitoring Object is to carry information reflecting the more up to date values of the current connection for monitoring purposes. Resources such as Link Quality, Radio Signal Strength, Cell ID are retrieved during connected mode at least for cellular networks.

Table 22. Object definition: OMA Connectivity Monitoring

Name	Value	Description
Name	Connectivity Monitoring	The name of the object
Object ID	3	The Object ID
LwM2M version	1.1	The LwM2M Version used
Object version		The version of this object definition

Table 23. Resource definition: Connectivity Monitoring resources (Elvaco implementation)

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
0	Network Bearer	R	Single	Yes	Int	0-50		7 = NB-IoT
1	Available Network Bearer	R	Multiple	Yes	Int	0-50		7 = NB-IoT
2	Radio Signal Strength	R	Single	Yes	Int		dBm	RSRP (NRSRP)
7	APN	R	Multiple	No	Str			APN
8	Cell ID	R	Single	No	Int			Cell ID
9	SMNC	R	Single	No	Int	0-999	%	MNC PLMN = SMNC + SMCC
10	SMCC	R	Single	No	Int	0-999		MCC PLMN = SMNC + SMCC

6.3.1.3. LwM2M object OMA Firmware Update

This LwM2M Object enables management of firmware which is to be updated. The firmware update will require to reboot the device.

Table 24. Object definition: OMA Firmware Update

Name	Value	Description
Name	Firmware Update	The name of the object
Object ID	5	The Object ID
LwM2M version	1.1	The LwM2M Version used
Object version	1.1	The version of this object definition

Table 25. Resource definition: Firmware update resources (Elvaco implementation)

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
5	Update Result	R	Single	Yes	Int	0..11		Firmware Update Resul
8	Firmware Update Protocol Support	R	Multiple	No	Int	0..5		0 = CoAP
9	Firmware Update Delivery Method	R	Single	Yes	Int	0..2		0 = Pull only

6.3.1.4. LwM2M object OMA LwM2M Cellular Connectivity

Table 26. Object definition: OMA LwM2M Cellular Connectivity

Name	Value	Description
Name	LwM2M Cellular Connectivity	The name of the object
Object ID	10	The Object ID
LwM2M version	1.1	The LwM2M Version used
Object version	1.1	The version of this object definition

Table 27. Resource definition: LwM2M Cellular Connectivity (Elvaco implementation)

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
4	PSM Timer	RW	Single	No	Int		s	NB-IoT T3412.
5	Active Timer	RW	Single	No	Int		s	NB-IoT T3324.
9	eDRX parameters for NB-S1 mode	RW	Single	No	Opa	0..255		NB-IoT eDRX
11	Activated Profile Names	R	Multiple	Yes	ObjLink			Link to APN Connection Profile object

6.3.1.5. LwM2M object OMA LwM2M APN Connection Profile

Table 28. Object definition: OMA LwM2M APN Connection Profile

Name	Value	Description
Name	LwM2M APN Connection Profile	The name of the object
Object ID	11	The Object ID
LwM2M version	1.1	The LwM2M Version used
Object version	1.1	The version of this object definition

Table 29. Resource definition: LwM2M APN Connection Profile (Elvaco implementation)

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
0	Profile name	RW	Single	Yes	Str			Human-readable identifier. Multiple connection profiles can share the same APN value but e.g. have different credentials.
1	APN	RW	Single	No	Str			Manual APN Writable in object resource 1.
2	Auto select APN by device	RW	Single	No	Bool			Auto APN Mode Writable in object resource 1.

ID	Name	Op	Instances	Mandatory	Type	Range enumerations	Units	Description
4	Authentication Type	RW	Single	Yes	Int			3 = None Writing currently not supported

6.3.2. Elvaco product specific LwM2M objects

This section list Elvaco specific LwM2M objects and resources.

6.3.2.1. LwM2M object Elvaco MCM Config

Table 30. Object definition: Elvaco MCM Config

Name	Value	Description
Name	Elvaco MCM Config	The name of the object
Object ID	33906	The Object ID
Object description	Configuration of the meter device specifics	
LwM2M version	1.1	The LwM2M Version used
Object version	2.12	The version of this object definition

Table 31. Resource definition: Elvaco MCM Config resources

ID	Name	Op	Multiplicity	Mandatory	Type	Range enumerations	Units	Description
0	Meter Readout Interval	RW	Single	Yes	Int			Interval in minutes
1	Report data encoding	RW	Single	Yes	Int			0 = SenML/ CBOR 1 = JSON 2 = MBus
2	Report frame type	RW	Single	Yes	Int			Message format ID identificator.
3	Eco mode enabled	RW	Single	No	Bool			
4	NFC Enabled	RW	Single	No	Bool			
5	NFC Config-locked	RW	Single	No	Bool			
6	Adjust time	W	Single	Yes	Int			Adjustment in seconds
10	Instantaneous readout trigger	E	Single	Yes				Trigger a meter readout.
13	Historic resend trigger	E	Single	No				Trigger upload of all historic data
14	Historic resend status	R	Single	No	Int			Number of messages in uplink queue
15	Apply APN staging profile	E	Single	Yes				Apply APN staging profile.
16	Config write status	R	Single	Yes	Bool			Result of last config write to flash
17	Meter Report Interval	RW	Single	Yes	Int			Interval in minutes
18	Meter Transmit Interval	RW	Single	Yes	Int			Interval in minutes
19	Meter Transmit Offset	RW	Single	Yes	Int			Offset in minutes
20	Meter Transmit Delay	RW	Single	Yes	Int			Delay in minutes

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
21	Meter Uploads Per Tx	RW	Single	Yes	Int			Max number of messages per tx interval
22	DTLS Min Time-out	RW	Single	Yes	Int			Timeout in sec-onds
23	DTLS Max Time-out	RW	Single	Yes	Int			Timeout in sec-onds
24	MQTT-SN Com-munication Timeout	RW	Single	Yes	Int			Timeout in sec-onds
25	MQTT-SN Com-munication At-tempts	RW	Single	Yes	Int			Total number of attempts
26	MQTT-SN Reg-ister Timeout	RW	Single	Yes	Int			OBSOLETE! Timeout in sec-onds
27	MQTT-SN Reg-ister Attempts	RW	Single	Yes	Int			OBSOLETE! To-tal number of at-tempts
28	MQTT-SN Pub-lish Timeout	RW	Single	Yes	Int			OBSOLETE! Timeout in sec-onds
29	MQTT-SN Pub-lish Attempts	RW	Single	Yes	Int			OBSOLETE! To-tal number of at-tempts
30	CoAP ACK Timeout	RW	Single	Yes	Int			Timeout in sec-onds
31	CoAP Max Re-transmit	RW	Single	Yes	Int			Number of re-transmissions
32	IOWA DTLS Min Timeout	RW	Single	Yes	Int			Timeout in sec-onds
33	IOWA DTLS Max Timeout	RW	Single	Yes	Int			Timeout in sec-onds
34	IOWA Communi-cation Retry Count	RW	Single	Yes	Int			Number of re-tries
35	IOWA Communi-cation Retry De-lay	RW	Single	Yes	Int			Delay in sec-onds
36	IOWA Communi-cation Sequence Retry Count	RW	Single	Yes	Int			Number of re-tries
37	IOWA Communi-cation Sequence Retry Delay	RW	Single	Yes	Int			Delay in sec-onds
38	Network Con-nection Maxi-mum Hold-off	RW	Single	Yes	Int			Delay in sec-onds
39	Network Search Period	RW	Single	Yes	Int			Period in sec-onds
40	Modem Restart Back-off Inter-vals	RW	Single	Yes	Str			min0-max0,min1-max1,... in mi-nutes
41	MDM Re-con-nect Back-off In-tervals	RW	Single	Yes	Str			min0-max0,min1-max1,... in mi-nutes

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
42	LwM2M Resume Back-off Inter-vals	RW	Single	Yes	Str			min0-max0,min1-max1,... in mi-nutes
43	Meter Max Retry Count	RW	Single	Yes	Int			Max number of retries when me-ter communica-tion fails
44	Auto Upload Age Limit	RW	Single	Yes	Int			Max age in mi-nutes of unsent measurements to upload
45	Auto Upload Or-der	RW	Single	Yes	Int			In what order should unsent measurements be uploaded. 0 = FIFO, 1 = LIFO.
46	Time Sync Source	RW	Single	Yes	Int			Which source to use for time syn-chronization. 0 = Manual, 1 = Net-work.
47	MDM Communi-cation Failures	RW	Single	Yes	Int			Maximum num-ber of failures before connec-tion is consid-ered broken.
48	Upload Protocol	RW	Single	Yes	Int	0..1		Protocol to use for meter data upload 0 = MQTT-SN 1 = LwM2M
49	Use PSM	RW	Single	Yes	Int	0..3		Power saving mode: 0 = Disa-bled, 1 = eDRX, 2 = PSM, 3 = PSM + eDRX
50	eDRX Mode	RW	Single	Yes	Int	0..1		eDRX mode: 0 = Automatic, 1 = Manual
51	Enable RAI	RW	Single	Yes	Int	0..3		RAI flag value(0|1|2|3): 0 - Disa-bled, 1 - Ena-bled for MQTT-SN, 2 - Enabled for LwM2M, 3 - Enabled for MQTT-SN and LwM2M
52	Power Source	RW	Single	Yes	Int	0..1		Configuration value for power source. Used when hardware unit cannot de-termine source. 0 = Battery, 1 = PSU
53	NB-IoT Radio Bands	RW	Single	Yes	Str			NB-IoT Radio Bands to use: band0,band1,...

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
54	Meter Identifica-tion source	RW	Single	Yes	Int	0..1		Use Fabrication number (aka Se-rial number) or Customer num-ber as identifica-tion. 0 = Fabrica-tion nbr, 1 = Customer nbr
55	MQTT Keepalive Timeout	RW	Single	Yes	Int	5..1092	Minutes	MQTT keeplive timeout for com-pliant mode in minutes.
56	Device reboot is required	R	Single	Yes	Bool			The device re-quires a reboot to apply the lat-est configuration changes.
57	LwM2M device queue mode	RW	Single	Yes	Int	0..2		Forced queue mode selection: 0 = automatic queue mode (as configured by the BS server), 1 = always ena-bled, 2 = always disabled(needs PSM turned off and/or private APN)
58	Additional Meter log enable	RW	Single	No	Bool			Enable addition-al meter log, on-ly some meters are supported
59	Roaming Home PLMN search	RW	Single	No	Int	0..3		Search value (0|1|2|3): bit 0 hpPlmnSearch, bit 1 overri-deLrplmnsi
60	DTLS hand-shake max re-winds	RW	Single	Yes	Int	0..255		Maximum num-ber of DTLS handshake step rewinds: 0 = dis-abled - infinite, x = non zero num-ber of rewinds allowed during dtls handshake
61	Alarm functional-ity enable bit-mask	RW	Single	Yes	Int			Select which meter info the module should monitor and re-act to. Ex: "0x000f" would enable alarms 0, 1, 2 and 3. All other are disa-bled.
62	Manual alarm reset bitmask	E	Single	No				Select which meter info to manually reset. Ex: "0x10" would reset alarm bit 4

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
63	Alarm mask re-set period	RW	Single	No	Int	0..42949 67295	Minutes	Time period in minutes for peri-odic reset of alarm mask, 0 = disable periodic reset.
64	Alarm auto-reset bitmask	RW	Single	No	Int			If enabled, an alarm message will be sent for every meter info triggered. Bit mask used to select which alarm mask bit(s) to auto-re-set.
65	Alarm hysteresis period	RW	Single	No	Int	0..65535	Minutes	Hysteresis in mi-nutes. Defines how long a me-ter info must be present in the meter before the module sends an alarm mes-sage. The same hysteresis ap-plies for reset-ting an alarm.
66	Alarm transmit max delay	RW	Single	No	Int	0..255	Minutes	Maximum delay before triggered alarm is trans-mitted: 0 = in-stantly transmit.
67	Alarm Topic	RW	Single	No	Str			Alarm MQTT-SN topic
68	Measurement mode	RW	Single	No	Int	0..1		Measurement mode of the de-vice: 0 = Preset - default operating mode, 1 = Transparent mode - stores the complete measusment/ telegram and forces SenML/ CBOR encoding
69	Bootstrap URI	RW	Single	No	Str			URI of bootstrap server, e.g. coaps:// 1.2.3.4:5684
70	Force bootstrap	E	Single	No				Force device to reboot and boot-strap; n.b. this will erase all LwM2M server URIs

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
71	NFC Auto Lock enable	RW	Single	No	Bool			If enabled, NFC configuration will be locked after a timeout. PAK is required to ena-ble configuration again. Remote configuration is still possible.
72	NFC Auto Lock enable timeout	RW	Single	No	Int	2..14400	Minutes	Time in minutes after which NFC configuration will be locked auto-matically.

6.3.2.2. LwM2M object Elvaco MDM Server

Table 32. Object definition: Elvaco MDM Server

Name	Value	Description
Name	Elvaco MDM Server	The name of the object
Object ID	33905	The Object ID
Object description		
LwM2M version	1.1	The LwM2M Version used
Object version	2.2	The version of this object definition

Table 33. Resource definition: Elvaco MDM Server resources

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
0	URI	RW	Single	Yes	Str	0	0	URI to the meter data server
1	Protocol	RW	Single	Yes	Int	0..	0	0 = MQTT-SN
2	Transport Secur-ity Mode	RW	Single	Yes	Int	0..3	0	0 = PSK mode 3 = No security
5	Transport Secret Key	W	Single	Yes	Opa	0	0	Key to use with the selected se-curity mode
10	Connection con-fig	RW	Single	No	Int	0..1	0	0 = Optimized 1 = Compliant
11	Topic	RW	Single	No	Str	0	0	MQTT-SN topic

6.3.2.3. LwM2M object Elvaco Meter Data

Table 34. Object definition: Elvaco Meter Data

Name	Value	Description
Name	Elvaco Meter Data	The name of the object
Object ID	33911	The Object ID
Object description		
LwM2M version	1.1	The LwM2M Version used
Object version	2.0	The version of this object definition

Table 35. Resource definition: Elvaco Meter Data resources

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
0	Message type	R	Single	Yes	Int			
1	Message encod-ing	R	Single	Yes	Int			
2	Message data	R	Single	Yes	Opaque			

6.3.2.4. LwM2M object Elvaco Meter Info

Table 36. Object definition: Elvaco Meter Info

Name	Value	Description
Name	Elvaco Meter Info	The name of the object
Object ID	33908	The Object ID
Object description		
LwM2M version	1.1	The LwM2M Version used
Object version	2.4	The version of this object definition

Table 37. Resource definition: Elvaco Meter Info resources

ID	Name	Op	Multiplicity	Mandatory	Type	Range enumerations	Units	Description
0	Meter Model	R	Single	Yes	Str			User-friendly string
1	Meter ID	R	Single	Yes	Int			
2	Communication status	R	Single	Yes	Int	0..2		0 = OK 1 = No meter detected 2 = Error
3	Error flags	R	Single	No	Opaque			Error codes/Status bytes from meter
4	Extended Error codes	R	Single	No	Integer			Extended Error Codes from meter. Interpret as a bit mask.

6.3.2.5. LwM2M object Elvaco NB-IoT Info

Table 38. Object definition: Elvaco NB-IoT Info

Name	Value	Description
Name	Elvaco NB-IoT Info	The name of the object
Object ID	33909	The Object ID
Object description		
LwM2M version	1.1	The LwM2M Version used
Object version	2.2	The version of this object definition

Table 39. Resource definition: Elvaco NB-IoT Info resources

ID	Name	Op	Multiplicity	Mandatory	Type	Range enumerations	Units	Description
0	IMSI	R	Single	Yes	Int			International mobile subscriber identity
1	ICCID	R	Single	Yes	Str			Integrated circuit card identifier
2	Registrations	R	Single	Yes	Int			Number of network registrations done
3	Last registration duration	R	Single	Yes	Int			Duration in seconds
4	Modem model	R	Single	Yes	Str			
5	Modem firmware	R	Single	Yes	Str			
6	Registration uptime	R	Single	Yes	Int			Last network registration uptime in seconds

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
7	Modem firmware update	E	Single	Yes				Trigger modem FOTA. Parame-ter 0='<url>'
8	Modem firmware update result	R	Single	Yes	Int	0..3		Result of mo-dem FOTA. 0: Initial value. 1: Modem Firm-ware updated successfully. 2: Error during download. 3: Er-ror during up-date.

6.3.2.6. LwM2M object Elvaco NB-IoT Status

Table 40. Object definition: Elvaco NB-IoT Status

Name	Value	Description
Name	Elvaco NB-IoT Status	The name of the object
Object ID	33907	The Object ID
Object description		
LwM2M version	1.1	The LwM2M Version used
Object version	2.4	The version of this object definition

Table 41. Resource definition: Elvaco NB-IoT Status resources

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
0	Uptime	R	Single	Yes	Int			Uptime in sec-onds
1	Average current consumption	R	Single	No	Int			Consumption in uA (micro-amps)
2	Network classifi-cation	R	Single	No	Int			0 = Excellent 1 = Good 2 = Fair 3 = Poor
3	ECL	R	Single	Yes	Int	0..2		
4	RSSI	R	Single	Yes	Int			In tenths of dBm
5	SNR	R	Single	Yes	Int			In tenths of dB
10	MDM connection status	R	Single	No	Int	0..7		0 = OK 1 = Con-necting 2 = No credentials 3 = DTLS failed 4 = Communication failed 6 = Socket failed 7 = Idle
11	Current radio band	R	Single	Yes	Int	0..85		Current radio band id
12	Current Access Technology	R	Single	No	Int	0..1		Always 1 = NB-IoT.

6.3.2.7. LwM2M object Elvaco Syslog config

Table 42. Object definition: Elvaco Syslog config

Name	Value	Description
Name	Elvaco Syslog config	The name of the object
Object ID	33918	The Object ID
Object description	Configuration of syslog function-ality for device	
LwM2M version	1.1	The LwM2M Version used

Name	Value	Description
Object version	1.0	The version of this object definition

Table 43. Resource definition: Elvaco Syslog config

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
0	Syslog MQTT-SN topic	RW	Single	Yes	Str			Syslog MQTT-SN topic
1	Syslog storage level	RW	Single	Yes	Int	0..6		Minimum log level for storing entry (0..5, indicates DEBUG..CRITICAL), 6=OFF
2	Syslog auto-upload level	RW	Single	Yes	Int	0..6		Minimum log level for auto-uploading an entry (0..5, indicates DEBUG..CRITICAL), 6=OFF
3	Syslog auto-upload age limit	RW	Single	Yes	Int	0..1051200		Maximum age limit in minutes for auto-uploading log entry

6.3.2.8. LwM2M object Elvaco Transaction statistics

Table 44. Object definition: Elvaco Transaction statistics

Name	Value	Description
Name	Elvaco Transaction statistics	The name of the object
Object ID	33910	The Object ID
Object description		
LwM2M version	1.1	The LwM2M Version used
Object version	2.0	The version of this object definition

Table 45. Resource definition: Elvaco Transaction statistics resources

ID	Name	Op	Multi-plicity	Manda-tory	Type	Range enumer-ations	Units	Description
0	Reset statistics	E	Single	Yes				Resets statistics for what this object instance is tracking.
1	Transactions	R	Single	Yes	Int			Number of transactions.
2	Retransmissions	R	Single	Yes	Int			Number of retransmissions.
3	Lost transactions	R	Single	Yes	Int			Number of lost transactions.
4	Average response time	R	Single	No	Int			
5	Minimum response time	R	Single	No	Int			
6	Maximum response time	R	Single	No	Int			

6.3.3. Changing APN via LwM2M

Since changing APN is a potentially hazardous operation that may render the device disconnected from the mobile network, there is a rollback functionality in place when changing the APN.

To change APN, write the APN to the resource /10/1/1 and set APN mode to manual in /10/1/2. Once done, stage the changes by executing /33906/0/15. When executed, the device will reset and try to use the new APN.

If the device manages a successful bootstrapping, the new APN will be saved as the default. If a successful bootstrapping has not happened for some time, the device will roll back to the old APN and reset again.

6.4. Message formats

CMi6160 has two different message formats, *Standard* and *Tariff*, defining what data records are sent from the module. Below, the content of the message formats are listed in detail.

6.4.1. Standard

Table 46. Meter registers, standard message [0x38]

Field	Size	Data type	Description
Date and Time	6	INT32	Date and Time (Type F), e.g. 046Dxxxxxxxx M-Bus Format F for Date & Time
Meter ID	10	INT64	(Enhanced) Identification e.g. 0779xxxxxxxxxxxxxxxx Enhanced Meter Address for Diehl SHARKY
Energy	6-7	INT32	Energy consumption (Wh, J) e.g. 0406xxxxxxxx = xxxxxxxx * 0.001 MWh (kWh)
Volume	6	INT32	Volume (m^3) e.g. 0413xxxxxxxx = xxxxxxxx * 0.001 m^3
Power	4	INT16	Power (W) e.g. 022Bxxxxxx = xxxxxx * 0.001 kW (W)
Flow	4	INT16	Flow (m^3/h) e.g. 023Bxxxxxx = xxxxxx * 0.001 m^3/h
Fw temp	4	INT16	Forward temperature (°C) e.g. 025Axxxx = xxxx * 0.1 °C
Rt temp	4	INT16	Return temperature (°C) e.g. 025Exxxx = xxxx * 0.1 °C
Alarm codes	4	INT8	Alarm codes e.g. 01FD17xx Error Codes for Diehl SHARKY 775

6.4.2. Tariff

Table 47. Meter register, tariff message [0x39]

Field	Size	Data typ	Description
Date and Time	6	INT32	Date and Time (Type F), e.g. 046Dxxxxxxxx M-Bus Format F for Date & Time

Field	Size	Data typ	Description
Meter ID	10	INT64	(Enhanced) Identification e.g. 0779xxxxxxxxxxxxxxx Enhanced Meter Address for Diehl SHARKY
Fw temp	4	INT16	Forward temperature (°C) e.g. 025Axxxx = xxxx * 0.1 °C
Rt temp	4	INT16	Return temperature (°C) e.g. 025Exxxx = xxxx * 0.1 °C
Flow	4	INT16	Flow (m^3/h) e.g. 023Bxxxxxx = xxxxxx * 0.001 m^3/h
Volume	6	INT32	Volume (m^3) e.g. 0413xxxxxxxx = xxxxxxxx * 0.001 m^3
Tariff 1	6-7?	INT32	Tariff register 1 or cooling energy in combined meters Depending on meter configuration, Tariff 1 can be either energy, volume or time.
Tariff 2	6-7?	INT32	Tariff register 2 or heat energy in combined meters Depending on meter configuration, Tariff 2 can be either energy, volume or time.
Alarm codes	4	INT8	Alarm codes e.g. 01FD17xx Error Codes for Diehl SHARKY 775

6.5. Diehl Metering SHARKY & SCYLAR error codes

The error codes transported from an MCM is the status byte of the M-Bus header. Bits 71:64 of M-Bus header.

Meaning of Error Codes

The following information is from the documentation received from Diehl Metering. It has only been enhanced by formatting, content is as-is straight from the document. For latest error description please use the latest documentation available from Diehl Metering. Table 48, “Diehl error codes” [49] corresponds to Table 7 of EN 13757-3:2013 for M-Bus.

Table 48. Diehl error codes

Bit	Description	Usage
0	Reserved	-
1	Any application error	-
2	Power low	E-8, E-9
3	Permanent error	C-1, E-4
4	Temporary error	E-1, E-3, E-6, E-7, leak error
5	Manufacturer specific	See Table 49, “Error code positioning” [50]
6	manufacturer specific	See Table 49, “Error code positioning” [50]
7	manufacturer specific	See Table 49, “Error code positioning” [50]

Table 49, “Error code positioning” [50] explains the position of error codes from the User’s Manual of the meter.

Table 49. Error code positioning

Error	C-1	E-8	E-4	E-1	E-7	E-9	E-3	E-6	Leak error	E-5
M-bus status Byte	0x08	0x04	0x28	0x50	0x70	0x84	0xB0	0xD0	0xF0	0x10
Prio	Highest									Lowest

The table below lists existing meter error codes that, if present, are displayed on the meter along with their corresponding explanations

Table 50. Meter error interpretation

Error meter display	Error description
C-1	Basic parameter error in flash or RAM
E-8	No primary voltage (only if mains unit used) Powered by back-up battery
E-4	Hardware error in ultrasonic measurement Short-circuit in ultrasonic transducer Ultrasonic transducer defective
E-1	Temperature measurement error Sensor break Sensor short-circuit Temperature range exceeded [-9.9 °C … 190 °C]
E-7	No meaningful ultrasonic receive signal Air in the measuring path
E-9	Warning: battery nearly exhausted
E-3**	Temperature sensors reversed in hot and cold lines
E-6**	Wrong direction of flow Flow sensor incorrectly installed
E-B*	Leakage: leakage detected in energy meter
E-C*	Leakage: leakage pulse input 1
E-D*	Leakage: leakage pulse input 2
E-A*	Leakage: pipe break detected
E-5	Reading too frequently M-Bus communication not possible for short time

* Optional

** application-dependent

7. Technical specifications

Table 51. Mechanics

Type	Value	Unit	Comments
Dimensions (w x h x d)	45 x 37 x 9	mm	
Weight	20	g	
Mounting	In Diehl SHARKY & SCYLAR module slot 1.	-	
Antenna connector	MCX female	-	Can be used for either internal or external antenna.
SIM card	Slide, size Nano	-	

Table 52. Electrical connections

Type	Value	Unit	Comments
Power supply	Diehl Battery Pack D-cell	-	PS2 Rated

Table 53. Electrical characteristics

Type	Value	Unit	Comments
Nominal voltage	2.2 – 3.4	VDC	
Power consumption (max)	400	mA	
Power consumption (sleep mode)	6	µA	

Table 54. Environmental specifications

Type	Value	Unit	Comments
Operating temperature	+5 to +55	ºC	
Operating humidity	0 - 93	% RH	No condensation
Operating altitude	2000	m	
Pollution degree	Degree 1	-	
Usage environment	According to meter classification.	-	
Storage temperature	-20 to +60	ºC	Storage temperature for battery pack is separated. See info on specific battery pack.

Table 55. Mobile network

Type	Value	Unit	Comments
Max Transmit power (EIRP)	23.0	dBm	Not. Max Antenna gain 0dBi
Receiver sensitivity	-135	dBm	
Certified for Bands	20,8,3	-	Hardware support for: B1/B2/B3/B4/B5/B8/B12/B13/B17/B18/B19/20/B25/B26/B28/B66/B71/B85
3GPP	Release 14 (NB2)	-	

Table 56. User interface

Type	Value	Unit	Comments
Green LED	Green LED	-	
Red LED	Start-up, Error	-	
Push button	Start-up, reboot	-	
NFC Configuration	13,56	MHz	ISO/IEC 14443 Type 2Tag
Configuration	• NFC via Elvaco OTC App • Via LwM2M (Elvaco Evo DM-system, or third-party DM system) • Preconfiguration on delivery	-	

Table 57. General

Type	Value	Unit	Comments
Supported Protocols	LwM2M, MQTT-SN	-	Both over UDP
Security	DTLS 1.2	-	

Table 58. Data storage (examples)

Type	Value	Unit	Comments
Meter data storage	4500 meter data entries 1500 syslog data entries	-	Non-volatile memory. Storage partitioned to keep meter data and syslog entries separated.

8. Simplified Declaration of Conformity

Hereby, Elvaco declares that CMi6160 is in compliance with the following directives:

EU	UK
2014/53/EU (RED)	2017 No. 1206 (RED)
2014/30/EU (EMC)	2016 No. 1091 (EMC)
2014/35/EU (LVD)	2016 No. 1101 (LVD)
2011/65/EU + 2015/863 (RoHS)	2012 No. 3032 (RoHS)

See EU Declaration of Conformity (elvaco.com) for a complete description.

See UK Declaration of Conformity (elvaco.com) for a complete description.

9. Document history

Table 59. Versions

Version	Date	Description
1.0	2022-03	Release document was updated.
1.1	2022-05	Updated from feedback loop
1.2	2022-07	Information about configuration and default values was updated.
1.3	2022-10	LwM2M tables were updated.
1.4	2022-12	Following content was updated: • Information about message formats. • Images
1.5	2023-02	Status and configuration parameters were updated.
1.6	2023-12	Minor layout changes were made.
2.0	2024-09	Adaption for release of new CMi6160 hardware and new FW 1.3.2.
2.1	2025-07	Adaptations for release of FW 1.4.0 and updated Radio Equipment Directive (RED).
2.2	2025-11	Clarifications made: • Improved description of Meter alarm monitor and Syslog • Corrected some typos in defualt configuration • Added 3 missing LwM2M resources

Manufacturer:

Elvaco AB / Kabelgatan 12 / 434 37 Kungsbacka / SWEDEN

info@elvaco.com

elvaco.com

Support:

support.elvaco.com

support@elvaco.se