CMi4160
User's Manual
English
v1.4

Contents

1 Document notes

All information in this manual, including product data, diagrams, charts, etc. represents information on products at the time of publication, and is subject to change without prior notice due to product improvements or other reasons. It is recommended that customers contact Elvaco AB for the latest product information before purchasing a CMi Series product.

The documentation and product are provided on an "as is" basis only and may contain deficiencies or inadequacies. Elvaco AB takes no responsibility for damages, liabilities or other losses by using this product.

1.1 Copyright and trademark

1.2 Contacts

Elvaco AB Headquarter
Kabelgatan 2T
434 37 Kungsbacka
SWEDEN
Phone: +46 300 30250
E-Mail: info@elvaco.com

Elvaco AB Technical Support
Phone: +46 300 434300
E-Mail: support@elvaco.se

Online: http://www.elvaco.com

2 Important usage and safety information

The following safety precautions must be observed during all phases of the operation, usage, service or repair of any CMi Series product. Users of the product are advised to convey the 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.

CMi4160 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.

3 Using this manual

3.1 Purpose and audience

This manual provides all information needed to mount, deploy and configure CMi4160 and targets installers and system integrators.

3.2 Online resources

To download the latest version of this user's manual, or to find information in other languages, please visit http://www.elvaco.com/.

3.3 Symbols

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

The Note symbol is used to mark information that is important to take into consideration for safety reasons or to assure correct operation of the meter connectivity module.

The Tip symbol is used to mark 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.

Table 1 provides information on how the product should be used.

Symbol	Description
	Waste electrical products should not be disposed of with household waste. Please recycle where facilities exist. Contact your Local Authority for recycling advise.
	Electrostatic-sensitive device. Please observe the necessary ESD protective measures when installing the module.

Table 1: Usage information

4 Introduction

4.1 Purpose

This chapter provides a general description of CMi4160. In the next-coming sections you will learn more about possible applications for the product and how CMi4160 can be combined with other products to build versatile solutions.

4.2 Application description

CMi4160 is a cost-effective meter connectivity module mounted in a Diehl Sharky heat meter. It uses a very energy-efficient scheme to deliver meter data to a receiving (application) server over a LoRaWAN network. Meter data is securely transmitted, using LoRaWAN end-to-end security scheme. CMi4160 can be retrofitted into already deployed meters or mounted in the meter before deployment.

4.3 Product features

CMi4160 has the capability to offer a combination of battery operation with very long lifetime and a versatile application through its many configuration options. Key features of the module include:

- **Extensive battery lifetime**
  The module's EcoMode feature enables the module to achieve a battery-lifetime of at least 11+1 years.
- **No meter installation needed**
  As soon as the meter connectivity module has been mounted in the meter and activated, it will join the LoRaWAN network and start to deliver meter data, i.e. no manual steps need to be taken in order to install the product in the meter.
- **Easy and secure commissioning**
  With Elvaco's One-Touch Commissioning (OTC), deployment, configuration and key transferring can be performed in a secure and flexible way. Use the Elvaco OTC App to enter your desired settings and place your phone next to the meter. New settings will be applied via NFC.
- **A unique and flexible message scheme**
  CMi4160 has several different message formats to choose from. This makes it easy to customize the product to the unique demands of each application.

4.4 Compatibility

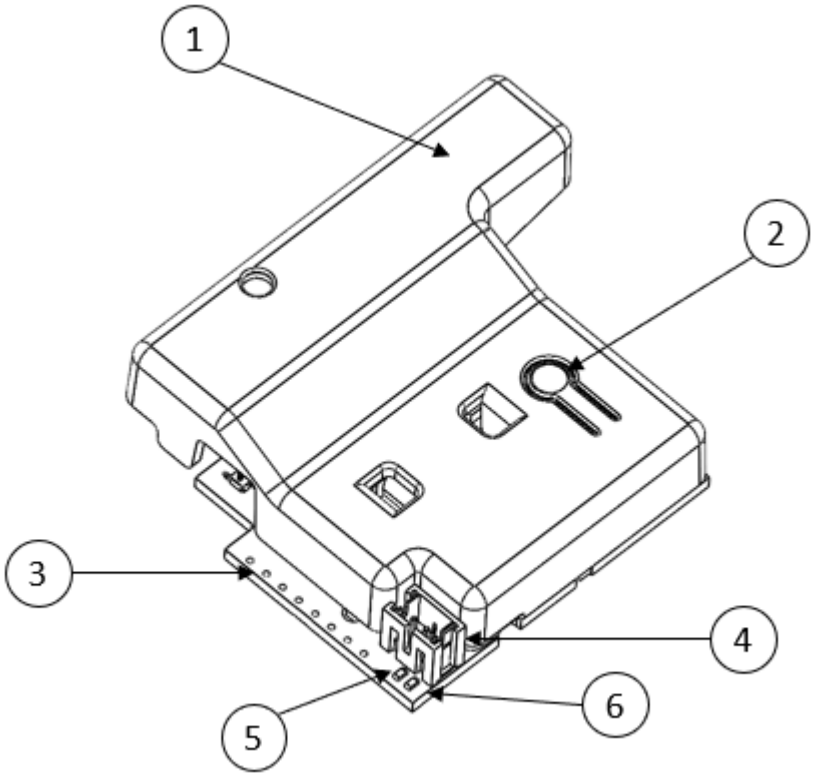
CMi4160 is compatible with Diehl Sharky 775 heat meters.

5 Getting started

5.1 Purpose

This chapter provides instructions on how to get started with CMi4160. After reading and carefully following each step of this chapter, your meter connectivity module should be mounted and connected to the LoRaWAN network.

5.2 Product specification



1. NFC antenna
2. Push button
3. Meter interface*
4. Power connector
5. Green LED
6. Red LED

**Ribbon cable, not shown in figure*

5.3 Mount and start-up the device

5.3.1 Mounting and connection

CMi4160 is mounted in module slot 2 of a Diehl Sharky. Grab the device by the outer edges, gently mount it into position and connect the meter interface cable to the meter interface (ribbon cable) (3).

5.3.2 Connection of battery pack

The CMi4160 is delivered with a battery pack. Connect the battery pack to the module's power connector (4) and gently place it with in the meter with the module.

5.3.3 Connection of antenna

CMi4160 is available in two different versions, with internal antenna (CMi4160Int) and with external antenna (CMi4160Ext). If using the external antenna version of the module, the antenna is connected using an MCX connector.

⚠	If using an external antenna, make sure to mount it at least 0.5 meters away from the meter in order not to cause interference.

5.3.4 Network preparation

For the module to connect to the LoRaWAN network, it needs to be added in the network server. More specifically, the following device information needs to be registered: Device EUI, Application key and Join EUI. (If using ABP mode, the following information should be registered instead: Application Session key, Network Session key and Device address.)

> Elvaco strongly recommends using Over-the-air activation (OTAA) to facilitate deployment and minimize the risk of duplicated keys.

5.3.5 Activation

Module activation
Upon delivery, CMi4160 is set to passive mode, which means no messages will be transmitted from the module. The module can be activated in one of the following ways:

a) **By using the module push button:** Press down the push button of the module for at least five seconds until the green LED lights up.

b) **By using the Elvaco OTC App:** Open the Elvaco OTC app (downloadable via Google Play) and scan the module (make sure NFC is activated on the phone). Remove the front enclosure of the meter if needed. Go to Apply mode, set the Power mode to "active" and click Apply settings. Place the phone next to the module. New settings are applied via NFC. You can make sure that the module has joined the LoRaWAN network correctly by checking the "network join" field in the Inspect tab of the OTC App.

Network join
When activated, CMi4160 will attempt to join the LoRaWAN network. The phase is indicated by short flash on red light, then green and red LED lights up for 1 second, followed by short flashes on the green LED until the module has joined the LoRaWAN network. When CMi4160 succeeds in joining the LoRaWAN network, the green LED will light up for 8 seconds, as illustrated by Figure 1.

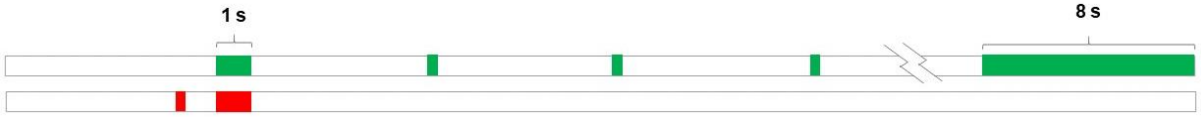


Figure 1: Network join LED indication

If the module fails to join the LoRaWAN 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.

When the module has joined the LoRaWAN network, it will perform three quick transmissions (regardless of transmit interval settings) before starting to use its configured transmit interval settings. By using the Elvaco OTC App, you can easily verify that the module is successfully communicating with the meter ("Meter communication") and is connected to the LoRaWAN network ("Network joined").

Soft start
In rare cases, if the battery is weak, the module will perform a soft start to be able to start-up despite the condition of the battery. The soft start takes 10 minutes to complete. The red LED will blink shortly every 10th second until the start-up has completed.

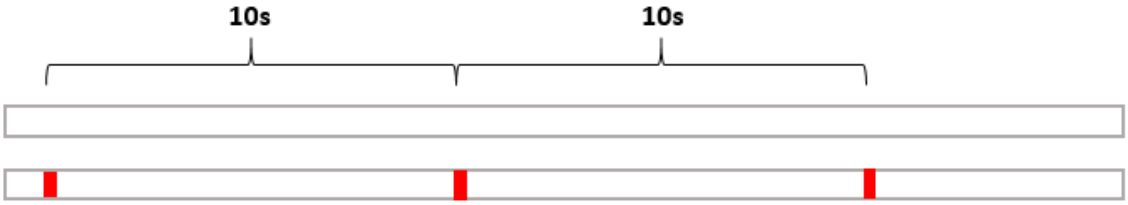


Figure 2: Soft start LED indication

5.3.6 Switch off/reboot module

To reboot the module, press and hold the push button (2) for 5-15 seconds. Release the button when the green LED is lit.

To switch off the module, press and hold the push button (2) for 15-20 seconds. Release the button when the red LED is lit.

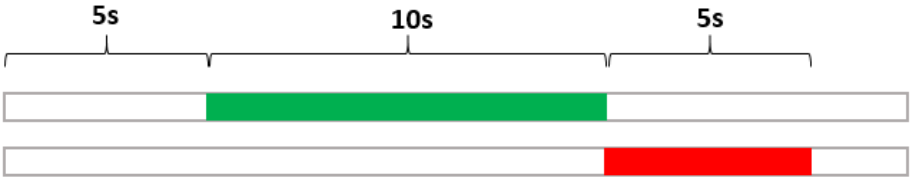


Figure 3: Reboot/switch off module

6 Administration reference

6.1 Purpose

This chapter contains detailed information about configuring options and the different message formats of CMi4160.

6.2 Security and access control

CMi4160 has a configuration lock feature, which prevents unauthorized access to the module. When configuration lock has been enabled, a Product Access Key will be needed to access the device. For more information about security and access control for CMi4160, please refer to the One-touch commissioning (OTC) documentation, available on the Elvaco website.

6.3 Configuration options

CMi4160 is configured via the Elvaco OTC App. It uses NFC to transfer settings to the module. Downlink may also be used for some applications, see section 6.8 Downlink for more information.

Please note that the Elvaco OTC app is only compatible with Android phones with Android 5.0 or later.

Table 2 provides a summary of all parameters and settings in CMi4160.

Field name (Abbr.)	**Description**	**Default value**	**Device access** Locked device & correct Product Access Key ***or*** Open device	**Device access** No Product Access Key	**Downlink**
Meter ID	Meter identification number of the meter. Not configurable.	N/A	Readable	Readable	N/A
Power mode	Used to activate/deactivate the module.	Passive	Readable / Writeable	Readable	N/A
Message format	The message format determines the structure and payload of the telegram sent from the module.	0x1E (Standard)	Readable / Writeable	Readable	Writeable
EcoMode	When activated, 11+1 years of battery-life is guaranteed by adapting the transmit interval of the module to current signal conditions.	On	Readable / Writeable	Readable	Writeable
Transmit interval	Sets the number of minutes between each transmission from the module.	60 min	Readable / Writeable	Readable	Writeable
Date & Time	Date and time set for the meter.	N/A	Readable	Readable	N/A

Set Time Relative	Adjusts the time of the meter relative to the current time.	N/A	N/A	N/A	Writeable
Set absolute time	Sets the time of the meter. When synchronizing the time via the Elvaco OTC App, the date, time and UTC offset settings of the phone will be applied to the meter.	N/A	Writeable	N/A	N/A
Set UTC offset	Sets the UTC offset of the meter.	N/A	Readable / Writeable	Readable	Witeable
Configuration Lock	Locks the module to prevent unauthorized access.	Open	Readable / Writeable	Readable	Writeable
LoRaWAN settings					
Device EUI	Unique module identification number. Not configurable.	Device-unique 64-bit number	Readable	Readable	N/A
Activation type	Sets the way the device joins the LoRaWAN network.	OTAA	Readable / Writeable	Readable	N/A
Network join	Displays whether the module has joined the LoRaWAN network or not.	N/A	Readable	Readable	N/A
Join EUI	Application ID that determines where data ends up.	0x 94 19 3A 03 03 00 00 01 (internal antenna version)' 0x 94 19 3A 03 04 00 00 01 (external antenna version)	Readable / Writeable	Readable	N/A
Device address	32-bit address used by the module to identify itself on the LoRaWAN network.	N/A	Readable / Writeable	N/A	N/A
Current data rate	The current data rate used for the module.	N/A	Readable	Readable	N/A

Table 2: Configuration options

6.4 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 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.

6.5 Adaptive data rate (ADR)

CMi4160 supports Adaptive Data Rate (ADR), part of the LoRaWAN standard, where the network server determines the optimal rate of communication for the module based on current signal conditions. In the best radio conditions, the module will use its highest data rate (DR5) in order to be as energy efficient as possible. When signal conditions are poor, the network server will incrementally lower the data rate until it is able to receive the message. When the data rate is low, the energy consumption per telegram will increase.

6.6 Transmit interval

The transmit interval sets how frequently the module transmits data on the LoRaWAN network. The parameter can be set to a value between 5 and 1440 minutes (i.e. between 5-minute and daily values.)

If Lora Frequency Band L or N (Radio characteristics, Page 24) (0,1% Duty cycle limitation), is used for transmitting messages (configurated by network server operator) the minimum transmit interval for DR0, DR1 and DR2 should be set to minimum 20 min. This will not affect the transmit interval if EcoMode is activated.

The user must always make sure the limits of the local Max. e.r.p and Duty cycle for the current frequency band is to be followed.

6.6.1 EcoMode

When EcoMode is active, it is able to achieve a battery-life of at least 11+1 years by utilizing a table of allowed transmit intervals settings for each data rate. When radio conditions are poor (and data rate is low), the module will be able to send data less frequently in order to conserve battery-life. When signal conditions are good, the module will be able to send data more frequently. When EcoMode is enabled, the module will continuously check if the set transmit interval is "allowed" by the EcoMode table. If a lower transmit interval is needed for the specific data rate in order to achieve 11 years of battery life, the module will adjust the parameter accordingly.

Table 3 shows the transmit intervals that the module will use for different data rates in order to achieve a 11-year battery-life.

Data rate	Transmit interval
DR0	180 min
DR1	120 min
DR2-DR3	60 min
DR4	30 min
DR5	20 min

Table 3: Transmit intervals for different data rates in EcoMode

NB! If using message format Scheduled Extended+ (available from FW 1,05), the transmit intervals used in EcoMode will be according to Table 4 below. Since the Scheduled Extended+ is scheduled, the minimal practical limit is always 60 minutes.

Data rate	Transmit interval
DR0	360 min
DR1	240 min
DR2	120 min
DR3	60 min
DR4-5	30 min

Table 4: Transmit intervals for different data rates in EcoMode, when using Scheduled Extended+

.

If EcoMode has been disabled, guarantees about battery-life is no longer valid (even if EcoMode is activated later).

6.7 Message formats

CMi4160 has seven different message formats: *Standard*, *Compact*, *JSON*, *Scheduled – Daily redundant, Scheduled – Extended, Combined heat/cooling* and *Scheduled Extended+*. Each message type will be described in detail in this section.

6.7.1 Message structure

Message formats *Standard, Compact, Scheduled – Daily redundant. Scheduled – Extended*, *Combined heat/cooling,* and *Scheduled Extended+* are encoded on M-Bus format (date/time field are encoded on M-Bus format F). Message format JSON is encoded on JSON format.

All messages begin with one byte specifying the message format used.

All M-Bus encoded messages begin with one byte specifying the message format used. Then follows a sequence of data information blocks (DIBs), each one containing a data information field (DIF), a value information field (VIF) and a data field (DATA). The structure of the telegram is illustrated in Figure 4.

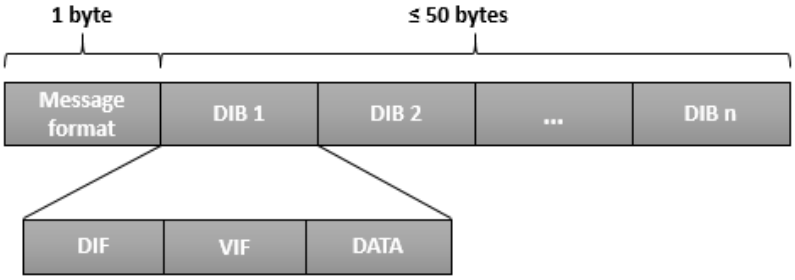


Figure 4: CMi4160 M-Bus message structure

Field	Size	Description
Message format identifier	1 byte	0x1E = Standard 0x1F = Compact 0x20 = JSON 0x21 = Scheduled – Daily redundant 0x22 = Scheduled – Extended 0x23 = Combined heat/cooling 0x3D = Scheduled Extended+ telegram 1* 0x3E = Scheduled Extended+ telegram 2* ** Only from FW 1.0.5*

Table 5: CMi4160 message formats

*Scheduled Extended+telegram 2 cannot be selected as a telegram type. It is the ID of the second telegram when using message format Scheduled Extended+.

6.7.2 Structure and payload

In this section, a detailed description of the structure and payload for each message format is presented.

Standard

Figure 5 illustrates the structure of message format *Standard*. For a detailed description of the payload, see Table 6.

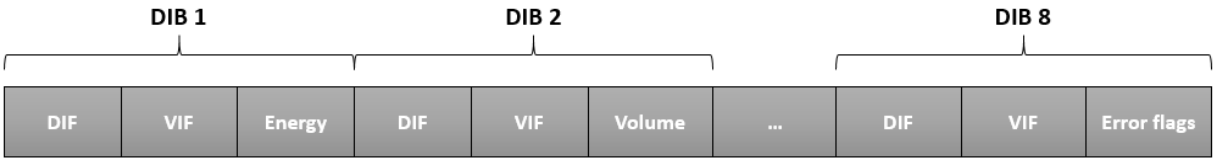


Figure 5: Structure, message format Standard

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x1E (Standard)
1	Energy	6-7 bytes	INT32	Energy consumption (Wh, J, Cal) 0403xxxxxxxx = xxxxxxxx Wh 0404xxxxxxxx = xxxxxxxx * 10 Wh 0405xxxxxxxx = xxxxxxxx * 100 Wh 0406xxxxxxxx = xxxxxxxx kWh 0407xxxxxxxx = xxxxxxxx * 10 kWh 040Exxxxxxxx = xxxxxxxx MJ 040Fxxxxxxxx = xxxxxxxx * 10 MJ 04FB0Dxxxxxxxx = xxxxxxxx MCal 04FB0Exxxxxxxx = xxxxxxxx * 10 MCal 04FB0Fxxxxxxxx = xxxxxxxx * 100 MCal
2	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 0417xxxxxxxx = xxxxxxxx * 10 m^3
3	Power	4 bytes	INT16	Power (kW) 022Bxxxx = xxxx W 022Cxxxx = xxxx * 10 W 022Dxxxx = xxxx * 100 W 022Exxxx = xxxx kW 022Fxxxx = xxxx * 10 kW
4	Flow	4 bytes	INT16	Flow (m^3/h) 023Bxxxx = xxxx * 0.001 m^3/h 023Cxxxx = xxxx * 0.01 m^3/h 023Dxxxx = xxxx * 0.1 m^3/h 023Exxxx = xxxx m^3/h 023Fxxxx = xxxx * 10 m^3/h

5	Fw temp	4 bytes	INT16	Forward temperature (°C) 0258xxxx = xxxx * 0.001 °C 0259xxxx = xxxx * 0.01 °C 025Axxxx = xxxx * 0.1 °C 025Bxxxx = xxxx °C
6	Rt temp	4 bytes	INT16	Return temperature (°C) 025Cxxxx = xxxx * 0.001 °C 025Dxxxx = xxxx * 0.01 °C 025Exxxx = xxxx * 0.1 °C 025Fxxxx = xxxx °C
7	Meter address	10 bytes	According to M-Bus EN13757-3 identification field	Meter address Byte 0-1 = DIF/VIF code Byte 2-5 = Meter ID (BCD8, lsByte -> msByte) Byte 6-7 = Meter manufacturer (lsByte -> msByte) Byte 8 = Meter version Byte 9 = Device type
8	Error flags	4 bytes	INT8	Error and warning flags 01FD17xx For further information about Alarm codes please refer to the meter's manual

Table 6: Payload, message format Standard

Compact

Figure 6 illustrates the structure for message format *Compact*. For a detailed description of the payload, see Table 7.

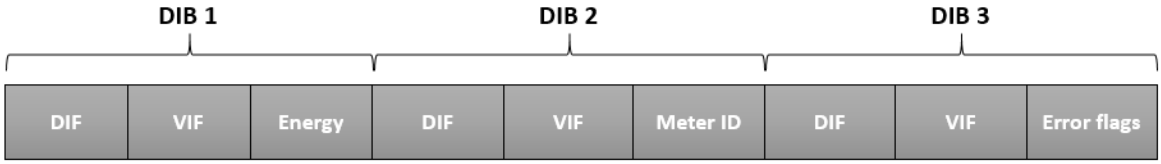


Figure 6: Structure, message format Compact

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x1F (Compact)
1	Energy	6-7 bytes	INT32	Energy consumption (Wh, J, Cal) 0403xxxxxxxx = xxxxxxxx Wh 0404xxxxxxxx = xxxxxxxx * 10 Wh 0405xxxxxxxx = xxxxxxxx * 100 Wh 0406xxxxxxxx = xxxxxxxx kWh 0407xxxxxxxx = xxxxxxxx * 10 kWh 040Exxxxxxxx = xxxxxxxx MJ 040Fxxxxxxxx = xxxxxxxx * 10 MJ 04FB0Dxxxxxxxx = xxxxxxxx MCal 04FB0Exxxxxxxx = xxxxxxxx * 10 MCal 04FB0Fxxxxxxxx = xxxxxxxx * 100 MCal

2	Meter address	10 bytes	According to M-Bus EN13757-3 identification field	Meter address Byte 0-1 = DIF/VIF code Byte 2-5 = Meter ID (BCD8, lsByte -> msByte) Byte 6-7 = Meter manufacturer (lsByte -> msByte) Byte 8 = Meter version Byte 9 = Device type
3	Error flags	4 bytes	INT8	Error and warning flags 01FD17xx For further information about Error flags please refer to the meter's manual

Table 7: Payload, message format Compact

JSON

For message format *JSON*, the data is presented in a plain text format. Table 8 provides a description of all fields included in the telegram.

Field	Description
Message format identifier	20 (Message format *JSON*)
Energy	Energy consumption
Unit	Unit of energy consumption (Wh, kWh, MWh, GWh, J, kJ, MJ, GJ, Cal, kCal, MCal or GCal)
Meter ID	Identification number of the meter in which the module is mounted.

Table 8: Fields, message format JSON

An example of a telegram for message format *JSON* is presented below:

```
{"E":12345678,"U":"kWh","ID":87654321}
```

Figure 7: JSON message example

Scheduled mode

For message formats of type "Scheduled" (Scheduled – Daily redundant and Scheduled – Extended), two types of messages will be transmitted from the module - a clock message and a data message. The difference between the two is described in Table 9. A detailed description of the payload of the clock message is provided in Table 10.

Message	Time interval	Description
Clock message	Once per day	The clock message presents the current time of the meter. It can be used to verify that the clock is correct and has not drifted more than accepted.
Data message	Determined by transmit interval parameter.	The actual meter data collected from the meter. For more information, see Table 12 and Table 12

Table 9: Clock message and data message

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0xFA (=Clock message)
1	Date/time	6 bytes	32-bit binary integer M-Bus type F	046Dxxxxxx = Valid date/time message 346Dxxxxxx = Invalid date/time message

Table 10: Payload, clock message

The clock message will be transmitted once every day and the data message at least (regulated by transmit interval parameter or EcoMode) once every day. The transmit interval can only be set the values listed in Table 11.

Note that although the meter readout will occur on top-of-the-hour, the data message will not necessarily be transmitted at that exact time. The LoRa transmission will occur after a random delay of 0-15 minutes to decrease the risk of collisions. The readout for the clock message occurs at a random hour (00:00-23:00) at a random minute in the 35-45 interval and will be transmitted immediately after readout.

When using message format S*cheduled*, the transmit interval cannot not be set to higher than 1440.

Parameter	Values	Unit
Transmit interval	60, 120, 180, 240, 360, 480, 720, 1440	Minutes

Table 11: Transmit interval options for Scheduled message formats

Scheduled- daily redundant

The data message of *Scheduled mode-daily redundant* contains an accumulated daily energy field, which is updated at 24:00 each day. In other words, depending on transmit interval settings and data rate, the field will be included in between 1-24 telegrams per day. This will increase the probability of the value being received. For example, if the transmit interval is set to "2", the accumulated energy read at 24:00 will be transmitted 12 times during the 24 next coming hours (every 2nd hour).

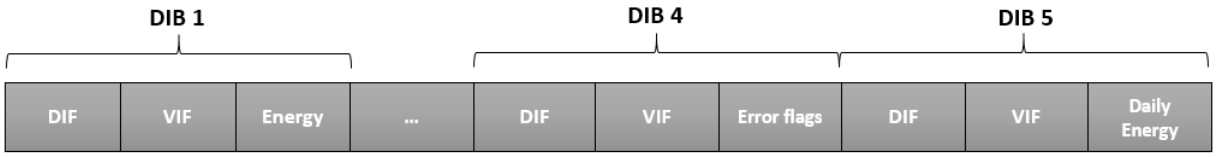


Figure 8: Structure, message format Scheduled – daily redundant

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x21 (Scheduled-daily redundant)
1	Energy	6-7 bytes	INT32	Energy consumption (Wh, J, Cal) 0403xxxxxxxx = xxxxxxxx Wh 0404xxxxxxxx = xxxxxxxx * 10 Wh 0405xxxxxxxx = xxxxxxxx * 100 Wh 0406xxxxxxxx = xxxxxxxx kWh 0407xxxxxxxx = xxxxxxxx * 10 kWh 040Exxxxxxxx = xxxxxxxx MJ 040Fxxxxxxxx = xxxxxxxx * 10 MJ 04FB0Dxxxxxxxx = xxxxxxxx MCal 04FB0Exxxxxxxx = xxxxxxxx * 10 MCal 04FB0Fxxxxxxxx = xxxxxxxx * 100 MCal
2	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 0417xxxxxxxx = xxxxxxxx * 10 m^3
3	Meter address	10 bytes	According to M-Bus EN13757-3 identification field	Meter address Byte 0-1 = DIF/VIF code Byte 2-5 = Meter ID, BCD8 Byte 6-7 = Meter manufacturer Byte 8 = Meter version Byte 9 = Device type
4	Error flags	4 bytes	INT8	Error and warning flags 01FD17xx For further information about Error flags please refer to the meter's manual

5	Meter date/time	6 bytes	INT32	Meter date and time (YY-MM-DD HH:MM) 046Dxxxxxxxx Bit 31-28 = Year-high* Bit 27-24 = Month Bit 23-21 = Year-low* Bit 20-16 = Day Bit 15 = Summertime flag** 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 year-high 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
6	Accumulated energy at 24:00	6-7 Bytes	INT32	Energy consumption (Wh, J, Cal) 4403xxxxxxxx = xxxxxxxx Wh 4404xxxxxxxx = xxxxxxxx *10 Wh 4405xxxxxxxx = xxxxxxxx *100 Wh 4406xxxxxxxx = xxxxxxxx kWh 4407xxxxxxxx = xxxxxxxx * 10 kWh 440Exxxxxxxx = xxxxxxxx MJ 440Fxxxxxxxx = xxxxxxxx * 10 MJ 44FB0Dxxxxxxxx = xxxxxxxx MCal 44FB0Exxxxxxxx = xxxxxxxx * 10 MCal 44FB0Fxxxxxxxx = xxxxxxxx * 100 MCal Note: Before a midnight reading has been performed the Function field of the DIF is set to "value during error state" to indicate that the value is not valid.

Table 12: Payload, message format Scheduled – daily redundant

Scheduled – Extended

The data message of *Scheduled mode-Extended* contains all the data fields from message format *Standard.* In addition to these, it also includes the meter date/time to transmit messages on the full hour. As for all *Scheduled* message formats, the transmit interval can only be set to the values included in Table 11.

A detailed description of the payload in the message format is presented in Table 13.

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x22 (Scheduled-Extended)
1	Energy	6-7 bytes	INT32	Energy consumption (Wh, J, Cal) 0403xxxxxxxx = xxxxxxxx Wh 0404xxxxxxxx = xxxxxxxx * 10 Wh 0405xxxxxxxx = xxxxxxxx * 100 Wh 0406xxxxxxxx = xxxxxxxx kWh 0407xxxxxxxx = xxxxxxxx * 10 kWh 040Exxxxxxxx = xxxxxxxx MJ 040Fxxxxxxxx = xxxxxxxx * 10 MJ 04FB0Dxxxxxxxx = xxxxxxxx MCal 04FB0Exxxxxxxx = xxxxxxxx * 10 MCal 04FB0Fxxxxxxxx = xxxxxxxx * 100 MCal
2	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 0417xxxxxxxx = xxxxxxxx * 10 m^3
3	Power / Flow / Fw temp / Rt temp	12 bytes	INT64	Byte 0-2 = DIF/VIF codes, 0x07FFA0 Byte 3 = Scaling of Power/Flow -Bit 6.4 (n), 10^{n-3} W, n = 0..7 -Bit 2..0 (m), $10^{\mathbf{m-6}}$ m^3/h, m = 0..7 Byte 4-5 = Fw temp (lsByte -> msByte), °C, 2 decimals Byte 6-7 = Rt temp (lsByte -> msByte), °C, 2 decimals Byte 8-9 = Flow (lsByte -> msByte), 10^{m-6} m^3/h Byte 10-11 = Power (lsByte -> msByte), 10^{n-3} W
4	Meter ID / Error flags	13 bytes	INT72	Byte 0-3 = DIF/VIF codes, 0x0DFF21E9 Byte 4 = Error flags Byte 5-8 = Meter ID (BCD8, lsByte -> msByte) Byte 9-10 = Meter manufacturer (lsByte -> msByte) Byte 11 = Meter version Byte 12 = Device type

5	Meter date/time	6 bytes	INT32	Meter date and time (YY-MM-DD HH:MM) 046Dxxxxxxxx Bit 31-28 = Year-high* Bit 27-24 = Month Bit 23-21 = Year-low* Bit 20-16 = Day Bit 15 = Summertime flag** 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 year-high 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 **0 = standard time, 1= daylight-saving time ***0 = timestamp is valid, 1 = timestamp is **not** valid

Table 13: Payload, message format Scheduled - Extended

Combined heat/cooling

Message format *Combined heat/cooling* is developed to be used in meters that measures both heating and cooling energy. Table 14 describes the telegram of the meter.

Message format *Combined heat/cooling* is only meant to be used in combined heat/cooling meters.

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x23 (Combined heat/cooling)
1	Heat energy	6-7 bytes	INT32	Energy consumption (Wh, J, Cal) 0403xxxxxxxx = xxxxxxxx Wh 0404xxxxxxxx = xxxxxxxx * 10 Wh 0405xxxxxxxx = xxxxxxxx * 100 Wh 0406xxxxxxxx = xxxxxxxx kWh 0407xxxxxxxx = xxxxxxxx * 10 kWh 040Exxxxxxxx = xxxxxxxx MJ 040Fxxxxxxxx = xxxxxxxx * 10 MJ 04FB0Dxxxxxxxx = xxxxxxxx MCal 04FB0Exxxxxxxx = xxxxxxxx * 10 MCal 04FB0Fxxxxxxxx = xxxxxxxx * 100 MCal

2	Cooling energy	8-9 bytes	INT32	Energy consumption (Wh, J, Cal) 0483FF02xxxxxxxx = xxxxxxxx Wh 0484FF02xxxxxxxx = xxxxxxxx * 10 Wh 0485FF02xxxxxxxx = xxxxxxxx * 100 Wh 0486FF02xxxxxxxx = xxxxxxxx kWh 0487FF02xxxxxxxx = xxxxxxxx * 10 kWh 048EFF02xxxxxxxx = xxxxxxxx MJ 048FFF02xxxxxxxx = xxxxxxxx * 10 MJ 04FB8DFF02xxxxxxxx = xxxxxxxx MCal 04FB8EFF02xxxxxxxx = xxxxxxxx * 10 MCal 04FB8FFF02xxxxxxxx = xxxxxxxx * 100 MCal
3	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 0417xxxxxxxx = xxxxxxxx * 10 m^3
4	Fw temp	4 bytes	INT16	Forward temperature (°C) Forward temperature (°C) 0258xxxx = xxxx * 0.001 °C 0259xxxx = xxxx * 0.01 °C 025Axxxx = xxxx * 0.1 °C 025Bxxxx = xxxx °C
5	Rt temp	4 bytes	INT16	Return temperature (°C) 025Exxxx = °C, 1 decimal
6	Meter address	10 bytes	According to M-Bus EN13757-3 identification field	Meter address Byte 0-1 = DIF/VIF code Byte 2-5 = Meter ID, BCD8 Byte 6-7 = Meter manufacturer Byte 8 = Meter version Byte 9 = Device type
7	Error flags	4 bytes	INT8	Error and warning flags 01FD17xx For further information about Error flags please refer to the meter's manual

Table 14: Payload, message format Combined heat/cooling

Scheduled Extended+

The data message of message format *Scheduled mode-extended* + contains three tariffs and all the meter data included in the *Standard* telegram. In addition to these, a timestamp from the meter (meter date/time) is included in each telegram. Table 14 and Table 15 show detailed descriptions of the payload for telegram 1 and 2 respectively.

Telegram 1

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x3D (Scheduled Extended+ telegram 1)
1	Energy	7	INT32	Energy consumption (Wh, J, Cal) 0403xxxxxxxx = xxxxxxxx Wh 0404xxxxxxxx = xxxxxxxx * 10 Wh 0405xxxxxxxx = xxxxxxxx * 100 Wh 0406xxxxxxxx = xxxxxxxx kWh 0407xxxxxxxx = xxxxxxxx * 10 kWh 040Exxxxxxxx = xxxxxxxx MJ 040Fxxxxxxxx = xxxxxxxx * 10 MJ 04FB0Dxxxxxxxx = xxxxxxxx MCal 04FB0Exxxxxxxx = xxxxxxxx * 10 MCal 04FB0Fxxxxxxxx = xxxxxxxx * 100 MCal
2	Energy Tariff 1	8	INT32	Energy Tariff 1 Same encoding as heat energy, e.g., 841003xxxxxxxx = xxxxxxxx Wh
3	Energy Tariff 2	8	INT32	Energy Tariff 2 Same encoding as heat energy, e.g., 842003xxxxxxxx = xxxxxxxx Wh
4	Meter address	10 bytes	According to M-Bus EN13757-3 identification field	Meter address Byte 0-1 = DIF/VIF code Byte 2-5 = Meter ID, BCD8 Byte 6-7 = Meter manufacturer Byte 8 = Meter version Byte 9 = Device type

5	Meter date/time	6 bytes	INT32	Meter date and time (YY-MM-DD HH:MM) 046Dxxxxxxxx Bit 31-28 = Year-high* Bit 27-24 = Month Bit 23-21 = Year-low* Bit 20-16 = Day Bit 15 = Summertime flag** 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 year-high 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 **0 = standard time, 1= daylight-saving time ***0 = timestamp is valid, 1 = timestamp is **not** valid

Table 15: Payload Scheduled Extended, telegram 1

Telegram 2

DIB	Field	Size	Data type	Description
0	Message format identifier	1 byte	-	0x3E (Scheduled Extended+ telegram 1)
1	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 0417xxxxxxxx = xxxxxxxx * 10 m^3
2	Power	4 bytes	INT16	Power (kW) 022Bxxxx = xxxx W 022Cxxxx = xxxx * 10 W 022Dxxxx = xxxx * 100 W 022Exxxx = xxxx kW 022Fxxxx = xxxx * 10 kW
3	Flow	4 bytes	INT16	Flow (m^3/h) 023Bxxxx = xxxx * 0.001 m^3/h 023Cxxxx = xxxx * 0.01 m^3/h 023Dxxxx = xxxx * 0.1 m^3/h 023Exxxx = xxxx m^3/h 023Fxxxx = xxxx * 10 m^3/h
4	Fw temp	4 bytes	INT16	Forward temperature (°C) 0258xxxx = xxxx * 0.001 °C 0259xxxx = xxxx * 0.01 °C 025Axxxx = xxxx * 0.1 °C 025Bxxxx = xxxx °C

5	Rt temp	4 bytes	INT16	Return temperature (°C) 025Cxxxx = xxxx * 0.001 °C 025Dxxxx = xxxx * 0.01 °C 025Exxxx = xxxx * 0.1 °C 025Fxxxx = xxxx °C
6	Meter address	10 bytes	According to M-Bus EN13757-3 identification field	Meter address Byte 0-1 = DIF/VIF code Byte 2-5 = Meter ID, BCD8 Byte 6-7 = Meter manufacturer Byte 8 = Meter version Byte 9 = Device type
7	Meter date/time	6 bytes	INT32	Meter date and time (YY-MM-DD HH:MM) 046Dxxxxxxxx Bit 31-28 = Year-high* Bit 27-24 = Month Bit 23-21 = Year-low* Bit 20-16 = Day Bit 15 = Summertime flag** 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 year-high 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 **0 = standard time, 1= daylight-saving time ***0 = timestamp is valid, 1 = timestamp is not valid
8	Error flags	4	INT8	Error and warning flags 01FD17xx For further information about Error flags please refer to the meter's manual

Table 16: Payload Scheduled Extended+, telegram 2

6.7.3 Meter communication error message

The payloads using M-Bus encoded data uses the Function field of the DIF to indicate errors. In this case it is set to "value during error state" (M-Bus standard EN 13757-3:2013) and the value sent should not be used. A typical case for this is when the module is unable to communicate with the meter and retrieve meter values, in which case all the fields in the payload have the DIF indicating "value during error state".

In case of erroneous data or if the module is unable to communicate with meter, bit 4-5 of the bit of the DIF code (first byte of each index field) will be set to 11b. For example, a DIF code of 0x02 will be set to 0x32 in case of error. (This applies to all message format except JSON.) For message format JSON, the actual value will be replaced with "null" in case of error state. If no meter communication is possible at all, all fields have this error indication set.

6.8 Downlink

CMi4160 supports configuration via downlink, i.e. sending commands to an end-device via the LoRaWAN network. **Note that this feature should only be used sparingly due to bandwidth consideration.** Communication via downlink is sent on port 2 and can only be made in a short window after an uplink transmission from module to server. Therefore, time-critical communication should not be performed over downlink.

Downlink commands are structured according to the following format: *"0x00" "TLV" "Number of bytes in configuration" "Configuration".* For a complete description of all available downlink commands, see Table 17**.**

Field name	TLV	Number of bytes in configuration	Configuration	Example
Configuration lock	0x05	0x01	0x00 = Locked 0x01 = Open	0x00050101 (Enables configuration lock)
Transmit interval	0x06	0x02	0xNumber of minutes between transmission (lsByte -> msByte)	0x0006021E00 (Sets the Tx interval to 30 minutes)
Message format	0x07	0x01	0x1E = Message format *Standard* 0x1F = Message format *Compact* 0x20 = Message format *JSON* 0x21 = Message format *Scheduled-daily redundant* 0x22 = Message format *Scheduled - Extended* 0x23 = Message format *Combined heat/cooling*	0x00070101F (Sets the message format to compact)
EcoMode	0x0F	0x01	0x00 = Disable EcoMode 0x01 = Enable EcoMode	0x000F0100 (Disables EcoMode)
Set Time Relative	0x13	0x04	0xNumber of seconds* (lsByte -> msByte) *Negative numbers supported.	0x0013043C000000 (Adds 60 seconds to the current time) 0x0013043C000080 (Subtracts 60 seconds from the current time)
UTC offset	0x17	0x02	0xNumber of minutes* (lsByte -> msByte) *Negative numbers supported.	0x17023C00 (Sets the UTC offset to +60 minutes) 0x17023C80 (Sets the UTC offset to -60 minutes)
Reboot	0x22	0x02	0x759E is used to reboot device (note endianness, LSB first).	0x0022029E75 (Reboots the device)

Table 17: Downlink commands

7 Technical specifications

Type	Value	Unit	Comments
Mechanics			
Dimensions (w x h x d)	63 x 50 x 15	mm	
Weight	19	g	
Mounting	In module slot of Diehl Sharky heat meter	-	
External antenna connector	MCX		
Electrical connections			
Supply voltage	Battery, lifetime up to 11 years	-	
Electrical characteristics			
Nominal voltage	3.0	VDC	
Power consumption (max)	50	mA	
Power consumption (sleep mode)	2.5	µA	
Environmental specifications			
Operating temperature	+5 to +55	ºC	
Operating humidity	0 – 93	% RH	No condensation
Operating altitude	2000	m	
Usage environment	Indoors	-	
Storage temperature	-20 to + 60	ºC	
Radio characteristics			
Operating Frequency (Frequency Band)	863,1 - 864,9 (K) 865,1 - 867,9 (L) 868,1 - 868,5 (M) 868,8 - 869,1 (N) 869,8 - 869,9 (Q)	MHz	Data Rate: DR0-DR5 Occupied Bandwidth: 200kHz
Output power	13	dBm	
Receiver sensitivity	-135	dBm	
LoRaWAN characteristics			
Device class	Class A	-	Bi-directional
LoRa version	1.0.2	-	
Activation	OTAA or ABP	-	
Data rate	DR0 - DR5	-	250 – 5470 bit/s
User interface			
Green LED	Start-up / reboot / switch-off indicator	-	
Red LED	Supercap indicator	-	
Push button	Start-up / reboot / switch off module	-	

Configuration	NFC via Elvaco OTC app or downlink data	-	

8 Type approvals

CMi4160 is designed to comply with the directives and standards listed below.

Approval	Description
EMC	EN 301 489-1, EN 309 489-3
RED	EN300 220-2 Short Range Devices (SRD), operating In the frequency range 25 MHz to 1 000 MHz;
LVD	EN 62368-1
LoRa Alliance®	LoRaWAN® Certified

9 Document history

9.1 Versions

Version	Date	Description
v0.1	2018-11	Proof of concept
v0.2	2019-04	Evaluation samples
v0.3	2019-10	Evaluation samples (final)
v1.0	2019-12	Commercial release
V1.1	2020-06	Commercial release v1.1
V1.2	2020-09	Commercial release v1.2
V1.3	2020-10	Commercial release v1.3
V1.4	2022-09	Commercial release v1.4

10 References

10.1 Terms and abbreviations

Abbreviation	Description
DIB	Data Information Block
DIF	Data Information Field
VIF	Value Information Field
MCM	Meter Connectivity Module

10.2 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 prefix 0b, i.e. 0b00001010 (ten)
- Payload data follows the M-Bus Standard with the least significant byte (LSB) first

10.3 European Standards

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