Special Documentation

LoRaWAN® communication protocol, model PEW-1000

EN



WIKA

© 8/2026 WIKA Alexander Wiegand SE & Co. KG
All rights reserved.
WIKA® is a registered trademark in various countries.

Prior to starting any work, read the operating instructions!
Keep for later use!

14843257.01 09/2026 EN PUBLIC

Contents

1. General ınformation 5
1.1 Abbreviations and definitions 5
1.2 Scope of this document 5
1.3 Conventions....5
2. Application protocol description 7
2.1 Purpose 7
2.2 Protocol key points....7
2.3 Logical device architecture 8
2.4 Data channels 8
2.5 Data encoding....8
2.5.1 Compact encoding format 9
2.6 Device identification sequence....9
2.7 Periodic traffic 9
2.8 Triggered traffic....9
2.8.1 Process alarms....10
2.8.2 Device alarms....13
2.9 Reading and writing device registers....14
2.9.1 Remote read/write 14
2.9.2 Local read/write 14
3. Register tables 15
3.1 Identification registers 15
3.1.1 Communication module identification 15
3.1.2 Sensor 1 identification 15
3.1.3 Channel 1 of sensor 1 identification 16
3.1.4 Channel 2 of sensor 1 identification 16
3.2 Configuration registers....17
3.2.1 Communication module configuration....17
3.2.2 Sensor 1 configuration....17
3.2.3 Channel 1 of sensor 1 configuration....17
3.2.4 Channel 2 of sensor 1 configuration....18
4. Upstream messages 19
4.1 General format....19
4.2 Data message (without and with alarm active) 20
4.3 Alarm messages 21
4.3.1 Process alarm message 21
4.3.2 Communication module alarm message....22
4.3.3 Sensor alarm message....22

14843257.01 09/2026 EN PUBLIC

4.3.4 Channel alarm message 23
4.4 Identification registers messages 24
4.4.1 Initial identification message 24
4.4.2 Read identification response message 25
4.4.3 Write identification response message 25
4.4.4 Identification register update message 26
4.5 Configuration registers messages 26
4.5.1 Configuration register update message 26
4.5.2 Read configuration response message 27
4.5.3 Write configuration response message 27
4.6 Keep alive message 27
4.7 Generic downlink answer message 28
4.8 Fetch additional downlink message 28
5. Downstream messages 29
5.1 General format 29
5.2 Identification registers messages 30
5.2.1 Read identification register message 30
5.2.2 Write identification register message 30
5.3 Configuration registers messages 31
5.3.1 Read configuration register message 31
5.3.2 Write configuration register message 32
5.4 Miscellaneous messages 34
5.4.1 Force close session message 34
5.4.2 Restore default configuration message 34
5.4.3 New battery inserted message 34
5.4.4 Get alarm status message 34
6. Connectivity protocol: LoRaWAN® 36
6.1 Radio network integration 36
6.2 Join procedure 36
6.3 LoRaWAN® ports 36
6.4 Message confirmation 37
7. Annexe: tables of values 38
7.1 Table of measurands 38
7.2 Table of units 38
7.3 Table of LoRaWAN® channel plan 38

14843257.01 09/2026 EN PUBLIC

1. General information

Note:

This protocol description applies to all PEW-1000 devices using the US915 or AU915 or AS923 LoRaWAN® Regional Parameter set. If you are using devices with a different Regional Parameter configuration, please refer to the relevant protocol documentation.

1.1 Abbreviations and definitions

ACK	Acknowledgement or confirmation, response sent by the receiver of a message to inform the emitter of said message of correct reception.
Alarm	In this document “alarm” is used as a generic technical term for condition-based message sent by the radio unit and do not assume any level of severity.
CM	Communication Module
ISN	Intelligent Serial Number, WIKA’s advanced device numbering scheme
LPWAN	Low Power Wide Area Network, a type of computer network.
Platform	Generic term for the data collection, processing and storage system that will bring meaning to the data sent by a device.
OV/UV	Overflow/Underflow, when value go over/under numerical limits.

1.2 Scope of this document

This technical guide gives a description of the wireless communication protocol used by the PEW-1000 LPWAN device in LoRaWAN® US915, AU915 and AS923 variants.

It is targeted towards developers who wish to design a protocol interpreter (a.k.a. driver) for the product, and users that want to get a deeper understanding of the capabilities of this WIKA product.

1.3 Conventions

Messages that originate from the connected device and go towards the LPWAN network (via LPWAN gateways) and cloud are qualified as “uplink” or “upstream”, and messages that originate from the cloud or LPWAN network and are destined to the connected device are qualified as “downlink” or “downstream”.

14843257.01 09/2026 EN PUBLIC

1. General information

Unless specified otherwise, multi-octet fields are encoded following a “big endian” convention, a.k.a “network order”. That order of octet’s transmission is the same as the left-to-right reading order.

“Big endian” convention is also used for memory registers. Example:
32-bits register, whose value is 0x13579BDF, at memory address X

Memory address	X	X+1	X+2	X+3
Memory value (byte)	0x13	0x57	0x9B	0xDF

Bits are numbered from left-to-right, starting at 7 and ending at 0, with bit 7 representing the most significant bit (MSB). Example:

Bytes	0								1								2…		
Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	…
Value	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^7	2^6	…

All bit fields follow the boolean logic convention, where “0” represents “false” or “disabled” or “inactive” and “1” represents “true” or “enabled” or “active”.

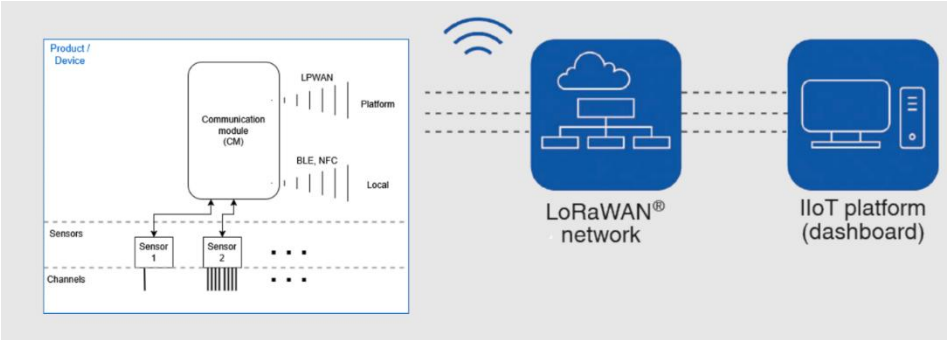
The digits in the document are written in English notation using ‘.’ as decimal separator and ‘,’ as a separator for large numbers. Example: 1,023.4

14843257.01 09/2026 EN PUBLIC

2. Application protocol description

2.1 Purpose

The purpose of the application protocol is to enable a connected device to communicate with an IIoT platform in order that one or several users are able to use all the features of the product remotely.



The protocol was designed to be compact in order to minimise the energy consumption (for longer battery life) and also use the shared radio spectrum more efficiently to enable many devices to be connected to a given network.

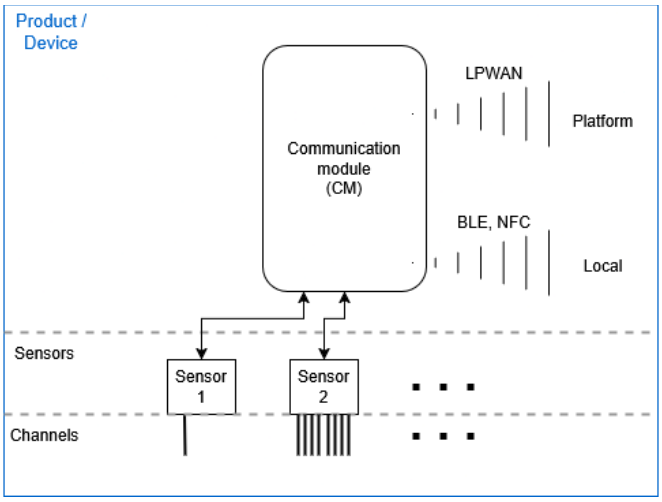
The translation, for uplink and downlink, between this binary context-dependent protocol, and a more versatile higher-level protocol chosen by the customer, is carried out by a software component called a “protocol interpreter” or “device driver”.

2.2 Protocol key points

- The device sleeps most of the time to save power. Commands queued in the network by the user to change device configuration can be received after each uplink transmission.
- The device “wakes up” at user-defined time intervals to take measurements. The measured value(s) are then checked for specific user-defined conditions (process alarms) that can trigger an immediate uplink transmission.
- One out of every N (user defined) measurements is send to the platform in an uplink transmission. The others are discarded.
- Should the device encounter any anomaly during internal self-tests, an alarm message will be sent in an immediate uplink transmission.

14843257.01 09/2026 EN PUBLIC

2.3 Logical device architecture



The Communication Module (CM) is the “LPWAN device” from the network perspective.

The CM is connected to one or more “sensors” whose purpose is to measure various physical phenomena. Each sensor is associated with identifying metadata that can be used to create a digital twin in the platform.

Sensors can have one or more channels to measure different physical phenomena (e.g. temperature, pressure, flow) or measure the same phenomenon on different axes (e.g acceleration on X, Y and Z). Each channel is associated with metadata, most importantly the measurand and measurement unit, that are needed to interpret the measured value in context.

2.4 Data channels

For this device, there is a single sensor, with two data channels:

Channel number	Physical measurement	Note
1	Pressure	Main channel, measurement provided by the electronic pressure sensor.
2	Temperature	Internal temperature of the sensor. Note: This is neither a process temperature nor an ambient temperature!

Both data channels are enabled by default.

2.5 Data encoding

To save radio bandwidth on messages sent the most frequently, measurement in data messages is encoded using a compact format.

14843257.01 09/2026 EN PUBLIC

2.5.1 Compact encoding format

Channel data can be expressed on a generic unitless scale [2,500 … 12,500] encoded as a 16-bit integer, with 2,500 corresponding to the start of the measuring range, and 12,500 to the end of it. One unit of measurement is equivalent of 0.01 % of the span of the channel.

This is the encoding format used for data messages (see §4.2).

The conversion between compact digital format and the real physical value is performed using the following formula:

$$physical\ value = \left(\left(\frac{encoded\ value - 2{,}500}{10{,}000}\right) * span\right) + start\ of\ measuring\ range$$

$$span = end\ of\ measuring\ range - start\ of\ measuring\ range$$

As an example, for a pressure channel of range of -1 … 5 bar, an encoded value of 0x1DC7 is interpreted as follows:

$$channel\ span = (5\ bar) - (-1\ bar) = 6\ bar$$
$$encoded\ value = 0x1DC7 = 7{,}623$$
$$physical\ value = \left(\left(\frac{7{,}623 - 2{,}500}{10{,}000}\right) * 6\ bar\right) + (-1\ bar)$$
$$physical\ value = 2.0738\ bar$$

Value 0xFFFF is used if a valid data can't be produced.

2.6 Device identification sequence

After successfully joining a network, the connected device will spontaneously send the values of some identification registers. This gives the platform relevant context to interpret the following messages, and specifically data messages.

2.7 Periodic traffic

At the end of each transmission period, the device will take a single measurement for each enabled channel of each enabled sensor, then transmit the measurement(s) in one or more data messages. The duration of a transmission is not set directly but defined as:

$$\boldsymbol{Transmission\ period\ =\ Measuring\ period\ *\ Transmission\ rate}$$

In addition, once per 24h, the connected device will send a "keepalive" message.

2.8 Triggered traffic

At the end of each measuring period, the device will take a single measurement for each enabled channel of each enabled sensor, and check if the values trigger any of the user-defined process alarm. If not, the device will go back to sleep without transmitting anything.

The device will also self-check and send communication module alarms, sensor alarms, or channel alarms if issues are detected.

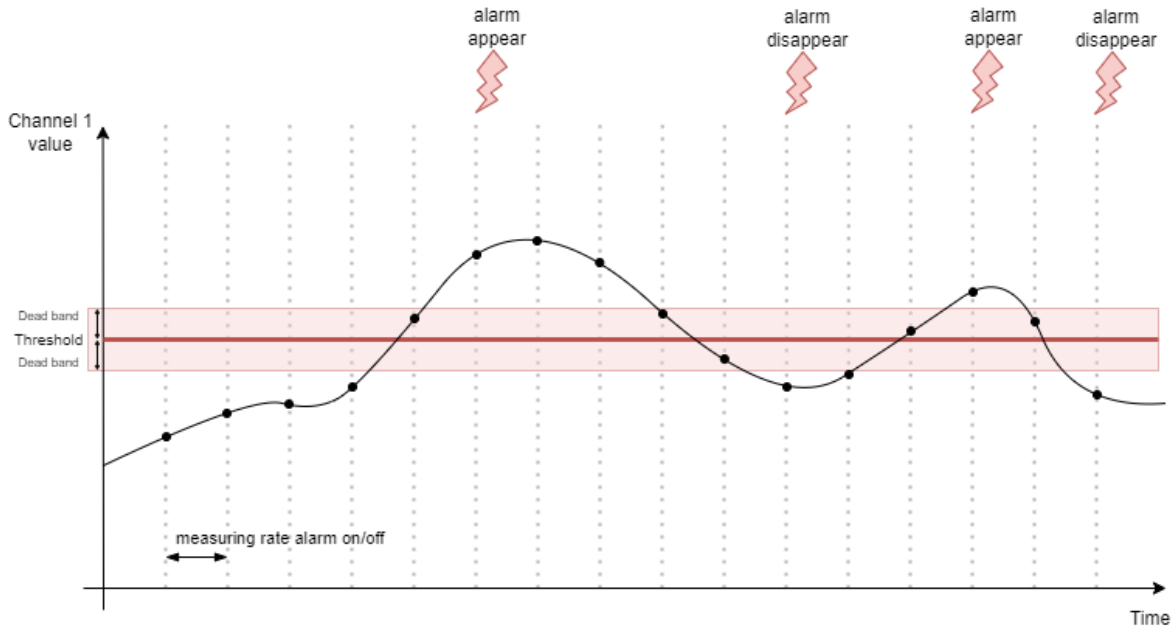
14843257.01 09/2026 EN PUBLIC

2.8.1 Process alarms

For each channel, up to six alarms can be configured and enabled:

Process alarm	Parameters		
	Threshold	Dead band	Delay
High threshold	Defined as floats in the native unit of the channel. High threshold ≥ low threshold	Single dead-band setting common to all threshold alarms of a channel	n/a
Low threshold			n/a
High threshold with delay			Defined in seconds
Low threshold with delay			
Rising slope	Defined as native unit of the channel per second (≥0)	n/a	n/a
Falling slope		n/a	n/a

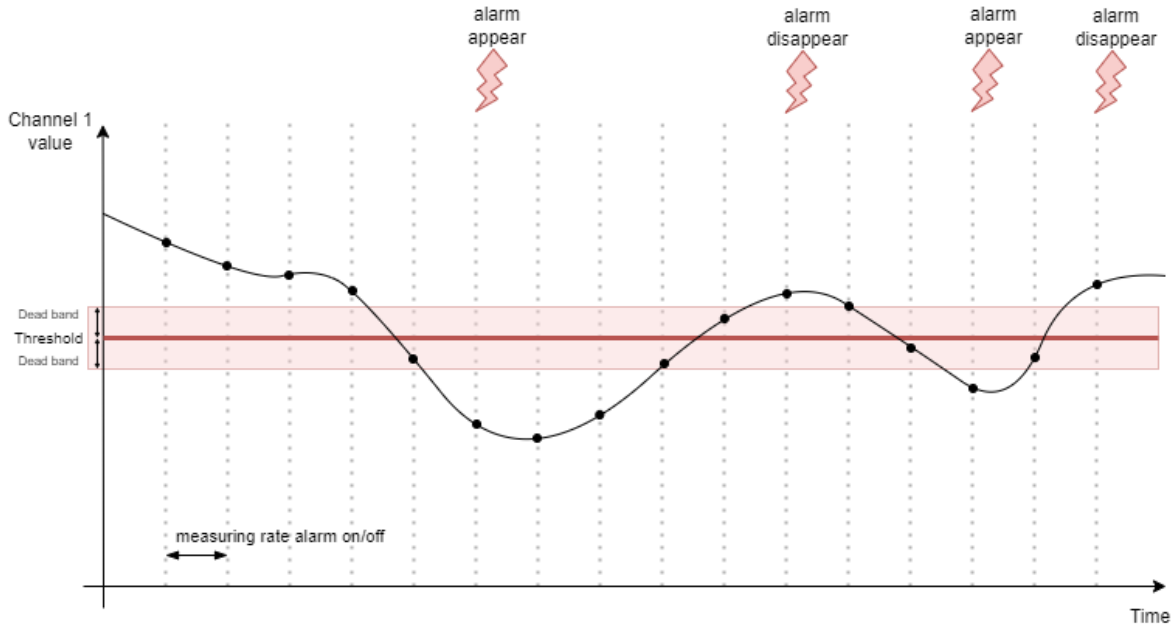
2.8.1.1 High threshold alarm



This alarm appears when the channel measurement is above the threshold limit plus dead band value. It disappears when the channel measurement is below the threshold limit minus dead band value.

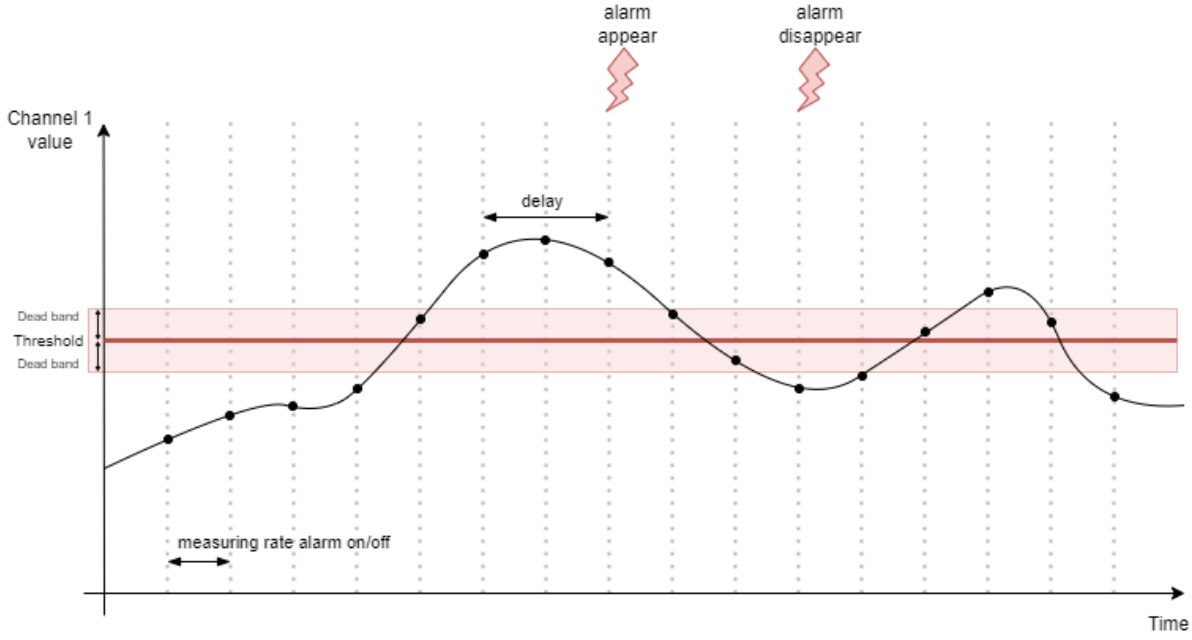
14843257.01 09/2026 EN PUBLIC

2.8.1.2 Low threshold alarm



This alarm appears when the channel measurement is below the threshold limit minus dead band value. It disappears when the channel measurement is above the threshold limit plus dead band value.

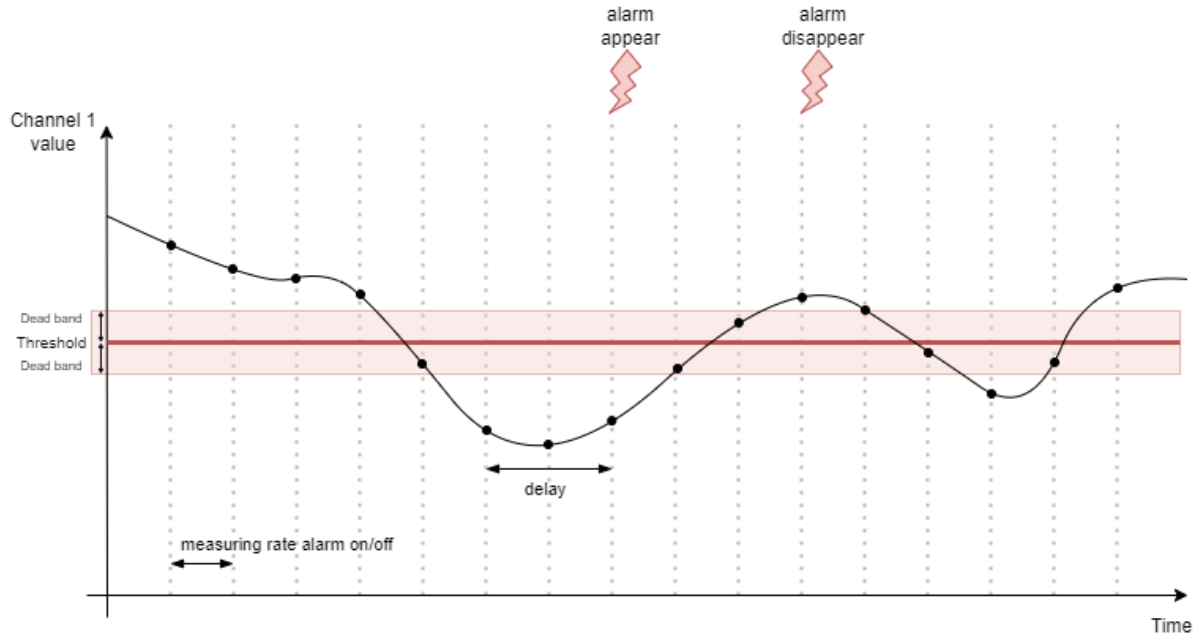
2.8.1.3 High threshold with delay alarm



This alarm appears when the channel measurement is above the threshold alarm limit plus dead band value during at least the configured delay. It disappears when the channel measurement is below the threshold limit minus dead band value.

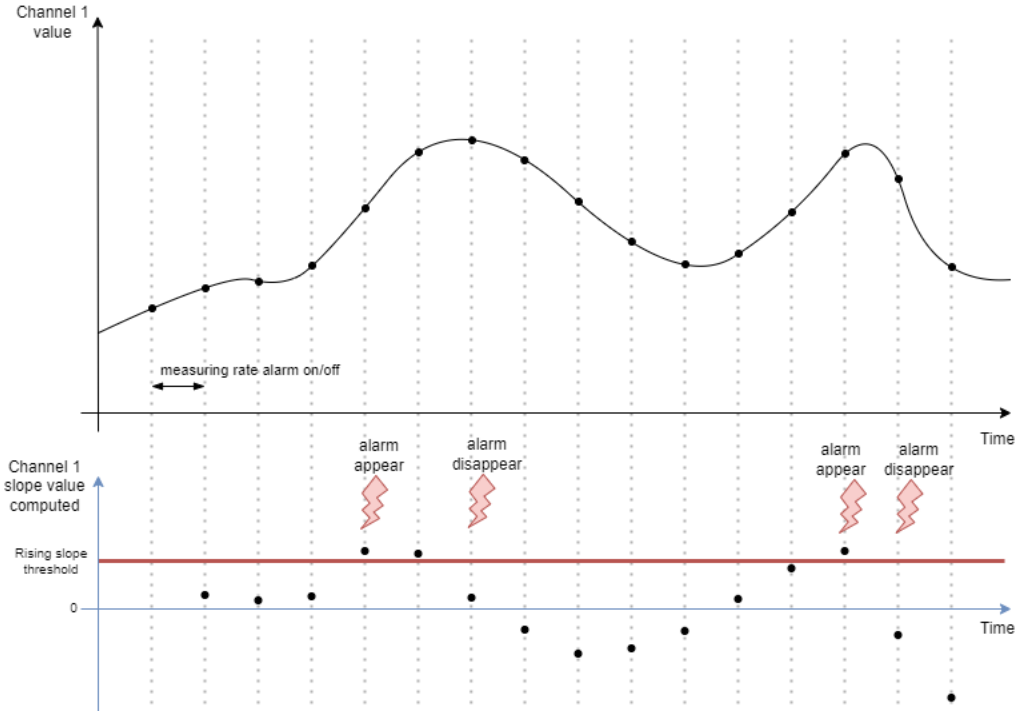
14843257.01 09/2026 EN PUBLIC

2.8.1.4 Low threshold with delay alarm



This alarm appears when the channel measurement is below the threshold alarm limit minus dead band value during at least the configured delay. It disappears when the channel measurement is above the threshold limit plus dead band value.

2.8.1.5 Rising slope alarm

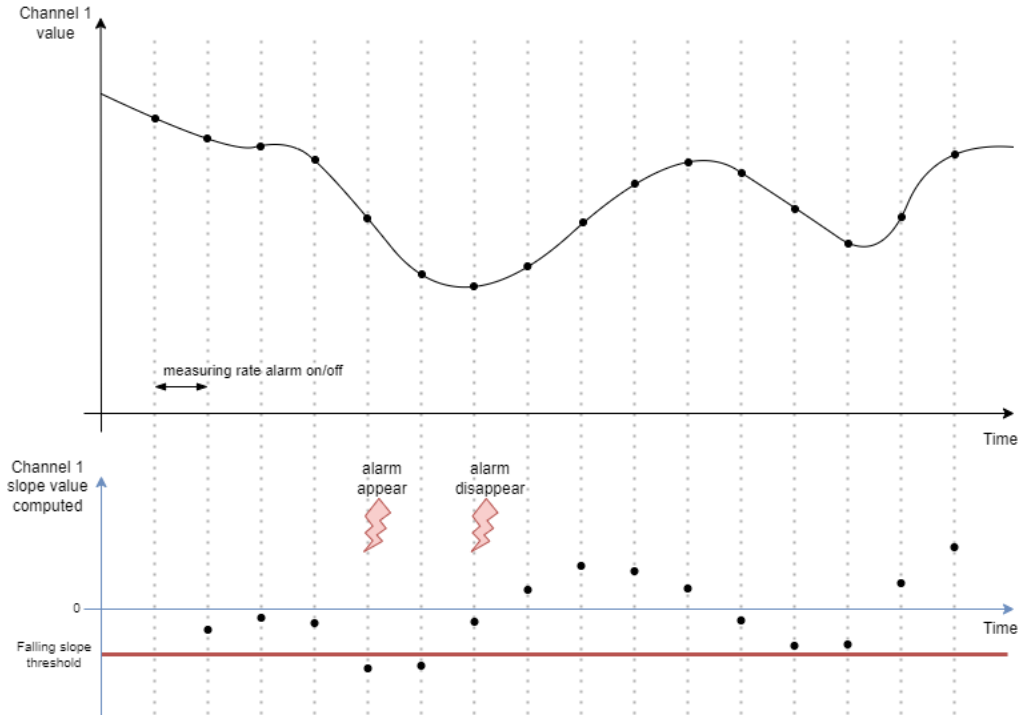


This alarm appears when the measurement slope is above the configured value. It disappears when the slope is below the configured value, or when the slope becomes negative.

$$slope = \frac{current\ measurement - previous\ measurement}{elapsed\ time\ between\ measurements}$$

14843257.01 09/2026 EN PUBLIC

2.8.1.6 Falling slope alarm



This alarm appears when the slope is below the configured value (multiplied by -1). It disappears when the slope is above the configured value (multiplied by -1), or when the slope becomes positive.

2.8.2 Device alarms

The communication module alarms, sensor alarms and channel alarms are predefined and can't be configured or disabled. They're used to inform the user of issues with:

- Battery on the communication module
- Airtime limitation being enforced, when device tries to emit too much for what is allowed by the regional laws for shared radio bands
- Measurement on a channel being out of measurement range or physical limits

14843257.01 09/2026 EN PUBLIC

2.9 Reading and writing device registers

Determining the state of the device and changing its configuration is done by reading and writing memory registers. The device has two set of registers: the identification registers, and the configuration registers. In both sets, registers are 1 to 6-bytes long and they can be “read only” (attempting to write will fail) or read/write.

When configuration is changed, using remote LPWAN commands or the local BLE interface, the device “revision counter” is incremented.

2.9.1 Remote read/write

Reading memory registers through LPWAN is done using two different commands, one for identification registers (see §5.2.1), one for configuration registers (see §5.3.1).

Several registers can be read with a single command, whether they are contiguous or not. Depending on the size to read, the device will respond in one or several packets (see §4.4.2 and §4.5.2). The response will contain indication of the start address and size of each read fragment.

Writing memory registers through LPWAN is also done using two different commands (see §5.2.2 for identification registers, §5.3.2 for configuration registers).

When writing registers to change a configuration, it’s important to be able to split the change of configuration over several packets, yet have the effective configuration change happen atomically. This is why register write messages have a counter and an “apply” bit, to be set at 1 in the last message of a write sequence.

2.9.2 Local read/write

It’s also possible to connect to a nearby device using Bluetooth® Low Energy and change the configuration this way. When a local session is initiated, LPWAN messages will be temporarily rejected to avoid any configuration conflict.

When a local configuration is changed and applied successfully, the device will send register update messages to inform the remote system of what changes were made locally. This allows the remote system to know the current device configuration, and allows it to correctly decode messages sent by the device.

14843257.01 09/2026 EN PUBLIC

3. Register tables

Identification and configuration data of the device are stored into registers tables. These two different memory tables are independent but handle in the same way, meaning that they are all read and write by similar commands.

For multi-byte registers, all read/write operations must cover the whole register, and not only sub-bytes of a register. That is to say, the “number of bytes” of a read or write command must always be equal to the full size of the targeted register(s).

Registers have a 11-bit address, represented as three hexadecimal digits: 0xHHH

All read/writes operations use a 16-bit value representing the target register address, left-shifted by 5 bits, and a 5-bit value for the length of the read/write (see table at the start of §4.4). For your convenience, the register tables of this section of the document contain, in addition to each register address, this address with a 5-bit left-shift and register size (column “@ + size”) to read, or write when applicable, each single register.

Register size is in byte, register type can be read-only or read-write.

3.1 Identification registers

“ID seq” is marked as “yes” if the value of the register is transmitted by the device during the identification sequence.

Identification registers not explicitly listed in the tables below must not be written to.

3.1.1 Communication module identification

Reg @	@ + size	Size	Field	Note	Type	ID seq
000	0001	1	Product ID	uint8, 0x0B = PEW-1000	R	yes
001	0021	1	Product sub-ID	uint8, 0x00	R	yes
002	0041	1	Channel plan	uint8, see table §7.3	R	yes
003	0061	1	Connected sensor	bit 0: sensor 1 connected	R	yes
004	0083	3	Firmware version	uint8.uint8.uint8 = major.minor.patch	R	
007	00E3	3	Hardware version	uint8.uint8.uint8 = major.minor.patch	R	
00A	0143	3	Production date	uint8.uint8.uint8 = Year (+2000).month [1-12].day [1-31]	R	
00D	01A5	5	WIKA ISN part 1	ASCII characters, not null-terminated	R	yes
012	0246	6	WIKA ISN part 2	ASCII characters, not null-terminated	R	yes

3.1.2 Sensor 1 identification

Reg @	@ + size	Size	Field	Note	Type	ID seq
03C	0782	2	Sensor type	uint16, = 0x0000	R	yes

Reg @	@ + size	Size	Field	Note	Type	ID seq
03E	07C1	1	Existing channels	bit 0: channel 1 exists bit 1: channel 2 exists	R	yes
03F	07E3	3	Firmware version	uint8.uint8.uint8 = major.minor.patch	R	
042	0843	3	Hardware version	uint8.uint8.uint8 = major.minor.patch	R	
045	08A3	3	Production date	same format as register @00A	R	
048	0905	5	WIKA sensor ISN 1#2	ASCII characters, not null-terminated	R	yes
04D	09A6	6	WIKA sensor ISN 2#2	ASCII characters, not null-terminated	R	yes

3.1.3 Channel 1 of sensor 1 identification

Reg @	@ + size	Size	Field	Note	Type	ID seq
078	0F01	1	Measurand	uint8, see table §7.1	R	yes
079	0F21	1	Unit	uint8, see table §7.2	R/W	yes
07A	0F44	4	Min measure range	float, in channel physical unit	R	yes
07E	0FC4	4	Max measure range	float, in channel physical unit	R	yes
082	1044	4	Min physical limit	float, in channel physical unit	R	
086	10C4	4	Max physical limit	float, in channel physical unit	R	
08A	1142	2	Accuracy	uint16, expressed in 0.001%	R	
08C	1184	4	Offset	float, in channel physical unit	R/W	
090	1204	4	Gain	float	R/W	
094	1283	3	Calibration date	same format as register @00A	R	

3.1.4 Channel 2 of sensor 1 identification

Reg @	@ + size	Size	Field	Note	Type	ID seq
0AA	1541	1	Measurand	uint8, see table §7.1	R	yes
0AB	1561	1	Unit	uint8, see table §7.2	R/W	yes
0AC	1584	4	Min measure range	float, in channel physical unit	R	yes
0B0	1604	4	Max measure range	float, in channel physical unit	R	yes
0B4	1684	4	Min physical limit	float, in channel physical unit	R	
0B8	1704	4	Max physical limit	float, in channel physical unit	R	
0BC	1782	2	Accuracy	uint16, expressed in 0.001%	R	
0BE	17C4	4	Offset	float, in channel physical unit	R/W	
0C2	1844	4	Gain	float	R/W	
0C6	18C3	3	Calibration date	same format as register @00A	R	

14843257.01 09/2026 EN PUBLIC

3.2 Configuration registers

Configuration registers not explicitly listed in the tables below should not be written to.

3.2.1 Communication module configuration

Reg @	@ + size	Size	Field	Note	Type
000	0004	4	Measuring period, no alarm active	uint32, seconds	R/W
004	0084	4	Measuring period, ≥1 alarm active	uint32, seconds	R/W
008	0102	2	Transmission rate, no alarm active	uint16, multiplier	R/W
00A	0142	2	Transmission rate, ≥1 alarm active	uint16, multiplier	R/W
00E	01C2	2	Under-voltage threshold	uint16, millivolts	R
014	0281	1	Enable BLE advertising	boolean	R/W

3.2.2 Sensor 1 configuration

Reg @	@ + size	Size	Field	Note	Type
02A	0541	1	Enabled channel	bit 0: channel 1 enabled bit 1: channel 2 enabled	R/W

3.2.3 Channel 1 of sensor 1 configuration

Reg @	@ + size	Size	Field	Note	Type
048	0901	1	Process alarm enable	bit 7: low threshold bit 6: high threshold bit 5: falling slope bit 4: rising slope bit 3: low threshold with delay bit 2: high threshold with delay	R/W
049	0924	4	Deadband	float, in channel physical unit	R/W
04D	09A4	4	Low threshold alarm	float, in channel physical unit	R/W
051	0A24	4	High threshold alarm	float, in channel physical unit	R/W
055	0AA4	4	Falling slope alarm	float, in channel physical unit per second	R/W
059	0B24	4	Rising slope alarm	float, in channel physical unit per second	R/W
05D	0BA4	4	Low threshold w/ delay	float, threshold in channel physical unit	R/W
061	0C22	2	Low threshold delay	uint16, delay in seconds	R/W
063	0C64	4	High threshold w/ delay	float, threshold in channel physical unit	R/W
067	0CE2	2	High threshold delay	uint16, delay in seconds	R/W

14843257.01 09/2026 EN PUBLIC

3.2.4 Channel 2 of sensor 1 configuration

Reg @	@ + size	Size	Field	Note	Type
082	1041	1	Process alarm enable	Same as register at address 0x048	R/W
083	1064	4	Deadband	float, in channel physical unit	R/W
087	10E4	4	Low threshold alarm	float, in channel physical unit	R/W
08B	1164	4	High threshold alarm	float, in channel physical unit	R/W
08F	11E4	4	Falling slope alarm	float, in channel physical unit per second	R/W
093	1264	4	Rising slope alarm	float, in channel physical unit per second	R/W
097	12E4	4	Low threshold w/ delay	float, threshold in channel physical unit	R/W
09B	1362	2	Low threshold delay	uint16, delay in seconds	R/W
09D	13A4	4	High threshold w/ delay	float, threshold in channel physical unit	R/W
0A1	1422	2	High threshold delay	uint16, delay in seconds	R/W

14843257.01 09/2026 EN PUBLIC

4. Upstream messages

4.1 General format

Upstream messages are messages sent wirelessly by the radio unit to the network and interpreted by the IIoT platform. Each upstream LPWAN packet contains a single message as its payload, and each message can contain none, one or a list of optional fields.

The packet format is as follows:

Byte	Size (bytes)	Note
0	1	Message type, see next table for details
1	1	Message subtype, see next table for details
2…	0 or more	Optional message content

The first two bytes of a message identifies its type:

Type	Subtype	Upstream message
0x10	**0x01**	Data message when no alarm is active
0x11	**0x01**	Data message when at least one alarm is active
0x12	**0x01**	Process alarm message
0x13	**0x01**	Communication module alarm message
0x13	**0x02**	Sensor alarm message
0x13	**0x03**	Channel alarm message
0x14	**0x01**	Initial identification message
0x14	**0x02**	Read identification response message
0x14	**0x03**	Write identification response message
0x14	**0x04**	Identification register update message
0x15	**0x01**	Configuration register update message
0x15	**0x02**	Read configuration response message
0x15	**0x03**	Write configuration response message
0x16	**0x01**	Keep alive message
0x17	**0x01**	Generic downlink answer message
0x17	**0x02**	Fetch additional downlink message

14843257.01 09/2026 EN PUBLIC

4.2 Data message (without and with alarm active)

One or several data messages are sent on each “transmission period” containing a data point for each enable channel or each enabled sensor.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1001 = when no alarm is active 0x1101 = when at least one alarm is active
2	1	Data ID	See next table for decoding
3-4	2	Data	Compact format, see §2.5.1
5	1	Data ID	Optional additional data
…	2	Data	Optional additional data

Data ID byte:

Bit	Description	Value
7-6	Sensor ID	0b00 = sensor 1
5-3	Channel ID	0b000 = channel 1 0b001 = channel 2
2-1	Data encoding	0b11 = compact encoding
0	Reserved	0

For PEW-1000: 0x06 => sensor 1 channel 1 data, 0x0E => sensor 1 channel 2 data

Example:

0x1001 06 2E97 0E 152F

Decoding	
1001	Data message, no alarm active
06	Following data is for sensor 1, channel 1, with compact encoding
2E97	Data measured at 94.27 % of channel 1 span
0E	Following data is for sensor 1, channel 2, with compact encoding
152F	Data measured at 29.23 % of channel 2 span

14843257.01 09/2026 EN PUBLIC

4.3 Alarm messages

Those alarm messages are sent:

- When an alarm appears or disappears
- As a response to a “get alarm status” downlink message

4.3.1 Process alarm message

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1201
2	1	Sensor & channel ID	0x00 = sensor 1, channel 1 0x08 = sensor 1, channel 2
3	1	Process alarm Bit = 1 when alarm is enabled and active Bit = 0 when alarm is not active, or disabled	bit 7: low threshold bit 6: high threshold bit 5: falling slope bit 4: rising slope bit 3: low threshold with delay bit 2: high threshold with delay
4	1	Sensor & channel ID	Optional additional alarm for another channel
…	1	Process alarm flags	Optional additional alarm for another channel

Examples:

0x1201 08 80 when low threshold alarm of second channel appears (data message with alarm active containing the channel value that made the alarm appear will follow)

Decoding	
1201	Process alarm message
08	Sensor 1, channel 2
80	The “low threshold” alarm of this channel is active

0x1201 08 00 when low threshold alarm of second channel disappears (data message without alarm active containing the channel value that made the alarm disappear will follow)

Decoding	
1201	Process alarm message
08	Sensor 1, channel 2
00	None of the channel process alarms are active

14843257.01 09/2026 EN PUBLIC

4.3.2 Communication module alarm message

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1301
2-3	2	Communication module alarms	bit 5: low voltage bit 4: memory error bit 3: air-time limitation

Example:

0x1301 0028

Decoding	
1301	Communication module alarm message
0028	“Low voltage” and “Air time limitation” alarms are active

4.3.3 Sensor alarm message

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1302
2	1	Sensor ID	0x00 = sensor 1
3-4	2	Sensor alarms	bit 15: sensor is busy bit 14: sensor memory integrity bit 13: sensor ALU saturation bit 0: sensor communication error

Example:

0x1302 00 0001

Decoding	
1302	Sensor alarm message
00	Sensor 1
0001	“Sensor communication error” alarm is active

14843257.01 09/2026 EN PUBLIC

4.3.4 Channel alarm message

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1303
2	1	Sensor & channel ID	0x00 = sensor 1, channel 1 0x08 = sensor 1, channel 2
3-4	2	Channel alarms	bit 5: max physical limit reached bit 4: min physical limit reached (only for channel 1)
5	1	Sensor & channel ID	Optional additional alarms for another channel
6-7	2	Channel alarms	Optional additional alarms for another channel

Example:

0x1303 00 0010 08 0020

Decoding	
1303	Channel alarm message
00	Following alarm(s) are for sensor 1, channel 1
0010	“Minimum physical limit reached” alarm is active
08	Following alarm(s) are for sensor 1, channel 2
0020	“Maximum physical limit reached” alarm is active

14843257.01 09/2026 EN PUBLIC

4.4 Identification registers messages

Register read/write are addressed this way:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Register address (high part)							
Byte 1	Register address (low part)				Size of the read/write, in bytes			

Read write up to 31 contiguous bytes are possible but the payload size limit may be lower due to LPWAN regional parameters and adaptive data rate.

4.4.1 Initial identification message

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1401
2-3	2	Register addressing	See first table in §4.4
4…	1…6	Register value	
x	2	Register addressing	Optional additional register(s) value, more can follow
x+1…	1…6	Register value	Optional additional register(s) value, more can follow

Example:

0x1401 0004 0B000201 07C1 03 0F02 0307 1542 0101

Decoding	
1401	Initial identification message
0004	4 bytes of data from the identification registers, starting at address 0x000
0B000201	Product ID, sub-ID, channel plan and connected sensors
07C1	1 byte of data from the identification registers, starting at address 0x03E
03	Existing channels for sensor 1
0F02	2 bytes of data from the identification registers, starting at address 0x078
0307	Measurand and unit for channel 1 of sensor 1, pressure in bar
1542	2 bytes of data from the identification registers, starting at address 0x0AA
0101	Measurand and unit for channel 2 of sensor 1, temperature in Celsius

14843257.01 09/2026 EN PUBLIC

4.4.2 Read identification response message

This message is sent by the device as a response to “Read identification register message” (see §5.2.1) if the read operation was successful.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1402
2-3	2	Register addressing	See first table in §4.4
4…	1…6	Register value	
x	2	Register addressing	Optional additional register(s) value, more can follow
x+1…	1…6	Register value	Optional additional register(s) value, more can follow

For examples of command and response, see §5.2.1

4.4.3 Write identification response message

This message is sent by the device as a response to “Write identification register message” (see §5.2.2).

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1403
2	1	Frame number	bit 6 to 2: same frame counter as downlink
3	1	Status	0x00 = write received but not applied 0x01 = write received and applied with success 0x02 = write rejected, read-only register 0x03 = write rejected, invalid value for 1 register or more 0x04 = write rejected, misaligned with boundaries of targeted register(s) 0x05 = whole set of write frames rejected due to invalid combination of parameters 0x06 = whole set of write frames rejected due to lack of platform response 0x07 = missing frame 0x08 = frame rejected, frame number already received
4-5	2	Revision counter	Present after message with “apply” bit was received
6	1	Total wrong frame counter	Present after message with “apply” bit was received

For examples of command and response, see §5.2.2

4.4.4 Identification register update message

Sent by the device after identification registers have been changed using the local interface. Only the content of registers that have been changed is sent.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1404
2-3	2	Register addressing	See first table in §4.4
4…	1…6	Register value	
x	2	Register addressing	Optional additional register(s) value, more can follow
x+1…	1…6	Register value	Optional additional register(s) value, more can follow

Example:

0x1404 0F21 07

Decoding	
1404	Identification register update message
0F21	1 byte of data from the identification registers, starting at address 0x079
07	Unit of channel 1 of sensor 1

4.5 Configuration registers messages

4.5.1 Configuration register update message

Sent by the device after configuration registers have been changed using the local interface. Only the content of registers that have been changed is sent.

Message format is identical to §4.4.4 with a type/subtype of 0x1501 instead of 0x1404.

Example:

0x1501 1041 00 0004 00004650 0102 0001

when local interface is used to disable all alarms on channel 2 (sensor 1), change the measuring period, and change the transmission multiplier without alarm active.

Decoding	
1501	Configuration register update message
1041	1 byte of data from updated configuration registers, starting at address 0x082
00	Disable all alarms for channel 2 of sensor 1
0004	4 bytes of data from updated configuration registers, starting at address 0x000
00004650	18,000 seconds (5 hours) measuring period when no alarm is active
0102	2 bytes of data from updated configuration registers, starting at address 0x008
0001	1x transmission multiplier when no alarm is active

14843257.01 09/2026 EN PUBLIC

4.5.2 Read configuration response message

This message is sent by the device as a response to “Read configuration register message” (see §5.3.1) if the read operation was successful.

Message format is identical to §4.4.2 with a type/subtype of 0x1502 instead of 0x1402.

For examples of command and response, see §5.3.1

4.5.3 Write configuration response message

This message is sent by the device as a response to “Write configuration register message” (see §5.3.2) if the write operation was successful.

Message format is identical to §4.4.3 with a type/subtype of 0x1503 instead of 0x1403.

For examples of command and response, see §5.3.2

4.6 Keep alive message

Message sent every 24h by the device

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1601
2	1	Status	bit 7: device is mains powered (‘0’ = battery powered) bit 6: device able to compute battery level bit 5: did the communication module restart since previous keep alive message?
3-4	2	Revision counter	Incremented with configuration changes, whether they’re done through LPWAN or the local interface.
5	1	Battery level indicator, in %	Present if device can compute battery level

Example:

0x1601 60 0002 64

Decoding	
1601	Keep alive message
60	Device is battery-powered, able to compute battery level and has been restarted
0002	Two modifications of configuration have been made since the default configuration
64	Battery is at 100%

4.7 Generic downlink answer message

This message is sent as a response to generic commands (force close, restore default configuration, new battery inserted), to error in generic request, to Identification or Configuration read register errors, or when some local configuration is ongoing (LPWAN configuration is not allowed at the same time).

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1701
2-3	2	Downlink message header	Type and subtype of the message being answered
4	1	Status	0x00 = success 0x01 = unsupported command (type/subtype invalid) 0x02 = logical error (e.g. non-existing channel) 0x03 = memory error (e.g. invalid register address) 0x04 = device error 0x05 = wrong frame format 0x06 = communication session lock (i.e. local configuration is ongoing)
5	1	Device error code	Only present for “device error” status

For examples of command and response, see §5.4.3.

4.8 Fetch additional downlink message

Sent by the device few seconds after a downlink to give the opportunity to the platform to send additional downlinks.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x1702

14843257.01 09/2026 EN PUBLIC

5. Downstream messages

5.1 General format

Downstream messages are messages sent wirelessly by the platform to the radio unit via the LPWAN network. Each downstream LPWAN packet contains a single message as its payload, and each message can contain none, one or a list of optional fields.

The general downstream packet format is identical to upstream packets, see §4.1.

The first two bytes of a message identifies its type:

Type	Subtype	Downstream message
0x01	**0x01**	Read identification register message
0x01	**0x02**	Write identification register message
0x02	**0x01**	Read configuration register message
0x02	**0x02**	Write configuration register message
0x03	**0x01**	Force close session message
0x03	**0x02**	Restore default configuration message
0x03	**0x03**	New battery inserted message
0x04	**0x01**	Get alarm status message

14843257.01 09/2026 EN PUBLIC

5.2 Identification registers messages

5.2.1 Read identification register message

This command is used to read one or several identification registers (see §3.1). Device response will be an “Read identification response message” (see §4.4.2) in case of success, and a “Generic downlink answer message” (see §4.7) in case of error.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x0101
2-3	2	Register addressing	See first table in §4.4
4…	2	Register addressing	Optional additional register(s) value, more can follow

Example:

0x0101 01AB

Decoding	
0101	Read identification register message
01AB	Start from register address 0x00D and read 11 bytes registers

Device response is: 0x1402 01AB 3141303132 334142433435

Decoding	
1402	Read identification response message
01AB	Read starts at 0x00D for a length of 11 bytes
3141303132	Identification register at address 0x00D has value = “1A012” (ASCII characters)
334142433435	Identification register at address 0x012 has value = “3ABC45” (ASCII characters)

5.2.2 Write identification register message

This command is used to write one or several identification registers (see §3.1). Device response will be an “Write identification response message” (see §4.4.3) in case of write success or error, and a “Generic downlink answer message” (see §4.7) in case the device is being configured locally.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x0102
2	1	Apply config / frame number	bit 7: “apply” bit, tell the device to apply the received write message(s) bit 6 to 2: frame counter
3-4	2	Register addressing	See first table in §4.4
5…	1…6	Register value	Value to write to the targeted register(s)
x	2	Register addressing	Optional additional register(s) value, more can follow
x+1…	1…6	Register value	Optional additional register(s) value, more can follow

14843257.01 09/2026 EN PUBLIC

If a write operation is split across several frames to respect the LPWAN frame size limit:

- First frame must have a counter=0, incremented by 1 for each following frame
- Only the last frame should have the “apply” bit set to ‘1’

Example:

0x0102 80 0F21 0E 17C8 3D4CCCCD 3F8020C5

Decoding	
0102	Write identification register message
80	Frame counter = 0, apply configuration after receiving this frame
0F21	Write starts at address 0x079 for a length of 1 byte (cover a 1-byte register)
0E	Sensor 1, channel 1 unit set to “psi”
17C8	Write starts at address 0xBE for a length of 8 bytes (cover 2 whole registers of 4 bytes each)
3D4CCCCD	Sensor 1, channel 2 offset = 0.05
3F8020C5	Sensor 1, channel 2 gain = 1.001

Device response is: 0x1403 00 01 0003 00

Decoding	
1403	Write identification response message
00	Response to frame with counter = 0
01	Configuration received and applied with success
0003	Revision counter = 3, three modifications have been made starting with factory default
00	Wrong frame counter = 0

5.3 Configuration registers messages

5.3.1 Read configuration register message

This command is used to read one or several configuration registers (see §3.2). Device response will be an “Read configuration response message” (see §4.5.2) in case of success, and a “Generic downlink answer message” (see §4.7) in case of error.

Message format is identical to §5.2.1 with a type/subtype of 0x0201 instead of 0x0101.

Example:

0x0201 0915 to read alarm mask, deadband, high/low threshold and up/down slope of channel 1 of sensor 1 (all registers are contiguous).

Decoding	
0201	Read configuration register message
0915	Start from configuration register address 0x048 and read 21 bytes

14843257.01 09/2026 EN PUBLIC

Device response is: 0x1502 0909 80 3E29FBE7 3E99999A

Decoding	
1502	Read configuration response message
0909	Read starts at 0x048 for a length of 9 bytes
80	Alarm mask => high & low threshold and rising & falling slope alarms enabled
3E29FBE7	Deadband = 0.166 channel unit for all threshold alarms (delayed and not delayed)
3E99999A	Low threshold = 0.3 channel unit

Device follows with: 0x1502 0A2C 414FAE14 3951B717 3DA1CAC1

Decoding	
1502	Read configuration response message
0A2C	Read starts at 0x051 for a length of 12 bytes
414FAE14	High threshold = 12.98 channel unit
3951B717	Falling slope = 0.0002 channel unit/second
3DA1CAC1	Rising slope = 0.079 channel unit/second

Device automatically splits messages to comply with LPWAN packet size limit (dynamic).

5.3.2 Write configuration register message

This command is used to write one or several configuration registers (see §3.2). Device response will be an “Write configuration response message” (see §4.5.3) in case of write success or error, and a “Generic downlink answer message” (see §4.7) in case the device is being configured locally.

Message format and splitting mechanism is identical to §5.2.2 with a type/subtype of 0x0202 instead of 0x0102.

Example: splitting an atomic write over several messages to limit packet size

Platform sending: 0x0202 00 0004 00000708

Decoding	
0202	Write configuration register message
00	Frame counter = 0
0004	Write starts at address 0x000 for a length of 4 bytes
00000708	Measuring period, no active alarm = 1800 seconds

Device response is: 0x1503 00 00

Decoding	
1503	Write identification response message
00	Response to frame with counter = 0
00	Configuration received but not applied, expecting more messages

14843257.01 09/2026 EN PUBLIC

Platform sending: 0x0202 01 0086 00000708 0006

Decoding	
0202	Write configuration register message
01	Frame counter = 1
0086	Write starts at address 0x004 for a length of 6 bytes
00000708	Measuring period, at least one active alarm = 1800 seconds
0006	Transmission rate, no alarm active = x6

Device response is: 0x1503 01 00

Decoding	
1503	Write identification response message
01	Response to frame with counter = 1
00	Configuration received but not applied, expecting more messages

Platform sending: 0x0202 82 0142 0002

Decoding	
0202	Write configuration register message
82	Frame counter = 2, “Apply config” bit = ‘1’
0142	Write starts at address 0x00A for a length of 2 bytes
0002	Transmission rate, at least one active alarm = x2

Device response is: 0x1503 02 01 000A 00

Decoding	
1503	Write identification response message
02	Response to frame with counter = 2
01	Configuration received and applied with success
000A	Revision counter = 10, ten modifications have been made starting with factory default
00	Wrong frame counter = 0

14843257.01 09/2026 EN PUBLIC

5.4 Miscellaneous messages

5.4.1 Force close session message

This message is used to discard a “session”, after one or several “write identification register” or “write configuration register” messages have been sent but not applied. As a result, the current register state is left unchanged.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x0301

5.4.2 Restore default configuration message

This command returns all configuration registers to the default ‘factory” value. Revision counter is also reset to 0.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x0302

5.4.3 New battery inserted message

Inform the device that the battery has been replaced with a fresh new battery and that the battery level indicator should be reset to 100% minus consumption since last device power cycling.

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x0303

Example:

0x0303

Decoding	
0303	New battery inserted

Device response: 0x1701 0303 00

Decoding	
1701	Generic answer
0303	Response to a “new battery” command
00	Success

5.4.4 Get alarm status message

This command allows to get the current alarm status of selected alarms or especially only selected sensors/channels.

14843257.01 09/2026 EN PUBLIC

Byte	Size (bytes)	Value	Note
0-1	2	Message type and subtype	0x0401
2	1	Alarm(s) requested	bit 0: request process alarms bit 1: request communication module alarms bit 2: request sensor alarms bit 3: request channel alarms
3	1	Sensor & channel ID	0x00 = sensor 1, channel 1 0x08 = sensor 1, channel 2 If sensor/channel/process alarms are requested but this optional byte is absent, device will respond with all available sensor, channel and/or process alarms.
4	1	Sensor & channel ID	Optional additional alarm for another channel

Example:

0x0401 03

Decoding	
0401	Communication module alarm message
03	Request communication module alarms, and process alarms of all sensors/channels

Device response: 0x1301 0000

Decoding	
1301	Communication module alarm message
0000	No active communication module alarm

Device follows with: 0x1201 00 00 08 04

Decoding	
1201	Process alarm message
00	Sensor 1, channel 1
00	No active process alarms
08	Sensor 1, channel 2
04	The high threshold with delay alarm is active

14843257.01 09/2026 EN PUBLIC

6. Connectivity protocol: LoRaWAN®

6.1 Radio network integration

The PEW-1000 LoRaWAN® version is a “class A” battery-powered radio end-device using version 1.0.4 of the protocol and US915, AU915 or AS923 regional parameters. The device is using the OTAA (over–the-air activation) LoRaWAN® procedure. Each device is configured at the factory with a securely generated 128-bit random key. Knowledge of this key is required to enable a network to communicate with the device.

Only in case you have ordered the device to be used with your own network provider the following information are provided to you together with the device:

- LoRaWAN® DevEUI
- LoRaWAN® AppEUI
- 128-bit application key (AppKey)

Refer to the network service provider for further details on how to integrate your device. In case you are using a network server provided by WIKA, your device is already pre-attached.

6.2 Join procedure

At power on, the device will start a LoRaWAN® join sequence:

- For the LoraWAN® US915 and AU915 product variant: 8 join attempts, one for each of the 8 sub-bands with 125 kHz channels followed by 1 join attempt in the sub-band with 500 kHz channels (total of 9 join attempts). All attempts have a ~40 second delay between them.
- For the LoraWAN® AS923 product variant: 2 join attempts with a 3 minute delay between the two attempts (total of 2 join attempts).

If a network is in radio range and has knowledge about the “Application Key” the device will join the network. If the join attempt is unsuccessful, the device goes to sleep for a random period, then launches a new join sequence. The sleep duration between two join sequences is defined as:

- 10 to 15 minutes, the first time,
- 55 to 60 minutes, the second time,
- 10 hours to 10h05, the following times for AS923
- 13 hours to 13h05, the following times for US915 and AU915

It keeps this last period infinitely, until join succeeds or device is turned off.

6.3 LoRaWAN® ports

All upstream traffic generated by the device is sent on LoRaWAN® port 1.

14843257.01 09/2026 EN PUBLIC

The device will process all data downstream messages, no matter via which port they have been received. However, it is recommended to use port 1 for downstream communication.

6.4 Message confirmation

The following upstream messages are sent as “confirmed” LoRaWAN® packets:

- Process alarm message and the Data message that follows immediately
- Communication module alarm message
- Sensor alarm message
- Channel alarm message
- Initial identification message
- Read identification response message
- Write identification response message
- Identification register update message
- Configuration register update message
- Read configuration response message
- Write configuration response message
- Keep alive message
- Generic downlink answer message

If the network server does not confirm the reception of these messages, the device will try to retransmit them up to two times after the initial transmission attempt.

The “keep alive” messages are sent as confirmed messages with a LoRaWAN MAC “Link Check Request”. If device does not receive a network “Link Check” answer for three consecutive days, it will stop doing measurement and start again to join a network.

14843257.01 09/2026 EN PUBLIC

7. Annexe: tables of values

7.1 Table of measurands

Value	Physical measurand
0x01	Temperature
0x03	Gauge pressure
0x04	Absolute pressure

7.2 Table of units

Value	Physical unit
0x01	[ºC] degree Celsius
0x07	[bar] bar
0x0D	[MPa] megapascal
0x0E	[psi] pound per square inch (lbf/in2, lb/in2)

7.3 Table of LoRaWAN® channel plan

Value	Channel plan
0x02	US915
0x05	AU915
0x07	AS923 (alias AS923-1)

14843257.01 09/2026 EN PUBLIC

LoRaWAN® is a trademark used under licence from LoRa Alliance®

The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by WIKA is under license. Other trademarks and trade names are those of their respective owners.

WIKA subsidiaries worldwide can be found online at www.wika.com.

14843257.01 09/2026 EN PUBLIC

Importer for UK
WIKA Instruments Ltd
Unit 6 & 7 Goya Business Park
The Moor Road
Sevenoaks
Kent • United Kingdom
TN14 5GY

WIKA Alexander Wiegand SE & Co. KG
Alexander-Wiegand-Straße 30
63911 Klingenberg • Germany
Tel. +49 9372 132-0
info@wika.de
www.wika.de