Climate | We make any home smart.

AQI sensor & notifier LoRaWAN communication protocol

The following document contains sensitive information and is shared with 3rd parties only under NDA.

Table of Contents

1 Communication concepts

1.1 Communication concepts related to LoRaWAN standard.

- Supported LoRaWAN MAC protocol version: 1.0.3;
- Supported LoRaWAN device class: A;
- LoRaWAN MAC Port: Uplink messages: 2. Downlink messages: 1, 2, 4-223;
- Maximum application payload size: Maximum allowed by document "LoRaWAN regional parameters" for DataRate 0 for the given region. In most of the cases this is 51 bytes;
- Consult the document "LoRaWAN regional parameters" for additional technical information (Especially for RX2 window timings);

1.2 Communication concepts related to device operation.

MClimate AQI sensor & notifier device periodically sends messages (keep-alive commands) to the server. The server can then send command to the device and the data will be received in the receiving windows, opened after each sent message, according to the LoRaWAN Class A devices protocol. The messages sent period is strict to the LoRaWAN duty cycle requirements. This device can send both confirmed/unconfirmed uplink messages depending on its configuration. We recommend configure it for confirmed uplinks, since they provide message retransmission in case of not acknowledge of the sent message.

When a command is sent to this device, the LoRaWAN message type can be with confirmation (recommended). In this way the server is sure the command is received by the device, by checking the ACK bit from the MAC header of the next received packet. If the server doesn't get a message confirmation, that command must be retransmitted by the server. One sent packet from the server may contain multiple commands for the end device to optimize communication time. The only restriction is the total number of bytes sent to be less or equal to the allowed application payload size. These sent bytes can combine both multiple write or/and read commands.

If this device receives valid command from the server, next uplink sending will be done as soon as possible. In this way the server can check faster the ACK bit, in case of confirmed downlink was received by the device, or the requested data or the keep-alive command data.

When the server wants to read some data from the device, the corresponding command code or command codes are sent to the device and the response will be sent together with the next keep-alive message. If the length of the command responses and the keep-alive packet is longer than the allowed by LoRaWAN MAC layer application payload size, the keep-alive packet will be omitted and only the command responses are sent by the device.

The aforementioned communication method is also described in Table 1.

Payload byte index	**Meaning**
0	Command 0 meaning
1	Command 0 data - optional
i	Command 1 meaning - optional
i+1	Command 1 data - optional
j	Command 2 meaning- optional
j+1	Command 2 data - optional
…	…

k	Command x meaning- optional
k+1	Command x data - optional

Table 1

When the server writes some device configuration with a command, the data is stored in the device non-volatile memory, so there isn't need to send this command again on next network join.

2 Messages explanation

Command code, [hex]	Command name	Sent from
01	Keep-alive	End device
02	Set keep-alive period	Server
03	Device buzzer control	Server
04	Read device hardware and software version	Server/ End device
05	Device LED's control	Server
10	Set network join retry period	Server
11	Set uplink messages type	Server
12	Get keep-alive period	Server/ End device
19	Get network join retry period	Server/ End device
1B	Get uplink messages type	Server/ End device
1C	Set device radio communication watch-dog parameters	Server
1D	Get device radio communication watch-dog parameters	Server/ End device

Table 2

2.1 Keep-alive command explanation

Periodically sent message which contains the most important device data.

The data is described in Table 3. In Table 4 example packet is given.

Payload index	Payload description		
	Value	Meaning	
0	01	Command byte for this packet.	
1	XX	sAQI bits 8:1 – sAQI[8:1].	
2	XX	Bit 7	sAQI bit 0 – sAQI[0]. Result value is calculated as: $sAQI = sAQI[8:0] * 16$
		Bits 6:2	AQI bits 4:0 – AQI[4:0]. Result value is calculated as:

			$AQI = AQI[4:0] * 16$
		Bits 1:0	CO2eq bits 9:8 – CO2eq[9:8].
3	XX	CO2eq bits 7:0 – CO2eq[7:0]. Result value is calculated as: $CO2eq, [ppm] = CO2eq[9:0] * 32$	
4	XX	VOC bits 7:0 – VOC[7:0]. Result value is calculated as: $VOC, [ppm] = VOC[7:0] * 4$	
5	XX	Relative humidity data. $\text{Relative humidity}, [\%] = \frac{XX*4}{10}$	
6	XX	Pressure data, bits 10:3 – P[10:3].	
7	XX	Bits 7:5	Pressure data, bits 2:0 – P[2:0]. Result value is calculated as: $P, [hPa] = \frac{P[10:0] * 40 + 30000}{100}$
		Bits 4:0	Temperature data, bits 10:6 – T[10:6].
8	XX	Bits 7:2	Temperature data, bits 5:0 – T[5:0]. Result value is calculated as: $t, [°C] = \frac{T[10:0] \ - \ 400}{10}$
		Bits 1:0	Air quality related parameters accuracy: 0: Sensor data is unreliable (Sensor stabilization); 1: Low accuracy, to reach higher accuracy please expose sensor once to good air (for about hour) and bad air for auto-trimming; 2: Medium accuracy: auto-trimming ongoing; 3: High accuracy.
9	XX	Device battery voltage. $\text{Voltage}, [\text{mV}] = XX * 8 + 1600$	

Table 3

Payload index	**Packet example**		
	Value	**Meaning**	
0	01	Command byte for this packet.	
1	07	sAQI[8:1] = 0x07.	
2	A4	Bit 7	sAQI[0] = 0x01. $sAQI = sAQI[8:0] * 16 = 0x0F * 16 = 240$
		Bits 6:2	$AQI = AQI[4:0] * 16 = 0x09 * 16 = 144$
		Bits 1:0	CO2eq[9:8] = 0x00.
3	4C	CO2eq[7:0] = 0x4C. $CO2eq, [ppm] = CO2eq[9:0] * 32 = 0x4C * 32 = 2432$	
4	03	$VOC, [ppm] = VOC[7:0] * 4 = 0x03 * 4 = 12$	
5	53	$\text{Relative humidity}, [\%] = \frac{0x53 * 4}{10} = 33.2$	
6	C9	P[10:3] = 0xC9.	
7	A9	Bits 7:5	P[2:0] = 0x05. $P, [hPa] = \frac{P[10:0] * 40 + 30000}{100} = \frac{0x64D * 40 + 30000}{100} = 945.2$
		Bits 4:0	Temperature data, bits 10:6 – T[10:6].
8	E3	Bits 7:2	Temperature data, bits 5:0 – T[5:0].

			$t, [°C] = \frac{T[10:0] - 400}{10} = \frac{0x278 - 400}{10} = 23.2$
		Bits 1:0	Air quality related parameters accuracy: 3
9	CD	$\text{Voltage}, [\text{mV}] = 0xCD * 8 + 1600 = 205 * 8 + 1600 = 3240$	

Table 4

2.2 Set keep-alive period command explanation

Sets the period of the device keep-alive command messages. See table 5 for details.

Byte index	Bit index	Hex value – Meaning
0	-	02 – The command will set the device keep-alive period.
1	-	XX – keep-alive period in minutes. Value 0x00 isn't applicable. Default value: 0x03.

Table 5

Example command, [Hex]: 020A

The example sets the keep-alive period to 10 minutes.

Note that the keep-alive period must respect the LoRaWAN messages duty cycle limitations. Otherwise the message will be sent when this is allowed. Also, the bigger keep-alive period, the less battery discharge. In most of cases, min. allowed period is 3 minutes and recommended values are 10 minutes or greater.

2.3 Get keep-alive period command explanation

This command is used to get the device keep-alive command messages period. Server sends the command code and the response is sent from the device together with next keep-alive command. The sent command request and the received command response are described in Table 6. The keep-alive in the response is omitted for clarity.

Byte index	Bit index	Hex value – Meaning	
		Sent request	Received response
0	-	12 – The command code.	12 – The command code.
1	-		XX – device keep-alive period in minutes.

Table 6

Example command sent from server: 0x12;

Example command response: 0x1209 – Device keep-alive period is 9 minutes.

2.4 Read device hardware and software version command explanation.

The server sends the command code to the device (Table 7) and with the next received message the response is received. The device response is described in Table 8.

Byte index	Bit index	Hex value – Meaning

0	-	04 – Read software and hardware version

Table 7

Example command: 0x04

Byte index	Bit index	Hex value - Meaning
0	-	04 – The command byte shows that is packet with the device hardware and software version.
1	7:4	X – Device primary hardware version.
	3:0	X – Device secondary hardware version.
2	7:4	X – Device primary software version.
	3:0	X – Device secondary software version.

Table 8

Example response, [Hex]: 041211 (Here the received keep-alive command data is omitted for clarity).

Decoding:

- 0x04 – Read software and hardware version response
- 0x12 - Device hardware version – version 1.2
- 0x11 – Device software version – version 1.1

2.5 Set network join retry period command explanation.

This command is used to set the period (T) of LoRaWAN join request sending from the end node device, in case it was unable to join the network from the first attempt. The command is described in Table 9.

Byte index	Bit index	Hex value – Meaning
0	-	10 – The command code.
1	-	T, [s] = XX * 5. Value 0x00 isn't applicable. Default value: 3 minutes.

Table 9

Example command: 0x10F0 – the server sets device LoRaWAN join request send period to 20 minutes.

Notes to this command:

- This join retry period (T) must comply to LoRaWAN messages duty cycle. Otherwise the join request will be sent on the next attempt. In most of cases, min. acceptable value for T is 240s. Recommended are higher values, for less battery discharge, e.g. 480s;
- This join retry period (T) is for the first 15 sent messages. After, the used LoRaWAN stack automatically changes the possibility to send join request to ~20

minutes for 20 network join attempts. If the device is still not joined to the network after these 20 attempts, next join request can be sent after ~3 hours and 15 minutes.

2.6 Get network join retry period command explanation

This command is used to get the period (T) of LoRaWAN join request sending from the end-node device. Server sends the command code and the response is sent from the device together with the next keep-alive command. The sent command request and the received command response are described in Table 10. The keep-alive in the response is omitted for clarity.

Byte index	Bit index	Hex value – Meaning	
		Sent request	Received response
0	-	19 – Command code.	19 – The command code.
1	-		XX – Network join retry period value. T, [s] = XX * 5.

Table 10

Example command sent from server: 0x19;
Example command response: 0x19C6 => T = 0xC6*5 = 198*5 = 990s = 16.5 minutes.

2.7 Set uplink messages type command explanation.

This command is used to set the device sent uplink message type. The command is described in Table 11.

Byte index	Bit index	Hex value – Meaning
0	-	11 – The command code.
1	-	00 – The device sends unconfirmed uplink messages; 01 – The device sends confirmed uplink messages. Default message type for the device

Table 11

Example command: 0x1101 – The server sets device uplink message type to confirmed.

2.8 Get uplink messages type command explanation

This command is used to get the device sent uplink messages type. Server sends the command code and the response is received together with the next keep-alive command. The sent command request and the received command response are described in Table 12. The keep-alive in the response is omitted for clarity.

Byte index	Bit index	Hex value – Meaning	
		Sent request	Received response
0	-	1B – Command code.	1B – The command code.
1	-		00 – The device operates with unconfirmed uplinks; 01 – The device operates with confirmed uplinks.

Table 12

Example command sent from server: 0x1B;
Example command response: 0x1B00 => Device uplinks are unconfirmed.

2.9 Set device radio communication watch-dog parameters command explanation

This command is used to set independent radio watch-dog configurations for confirmed and unconfirmed uplink messages sent from the device. It other words, the radio watch-dog configuration for confirmed uplinks no matter when the device works with unconfirmed uplinks, and vice versa. When no downlink is received for the defined Watch-Dog Period (WDP), the device resets itself. The command is described in Table 13.

Byte index	Bit index	Hex value – Meaning
0	-	1C – The command code.
1	-	XX – Watch-dog period (WDP) when confirmed uplinks are used by the device. $WDP_{confirmed}, [min] = XX * (Keep - alive\ period, [min]) + 7$ Default value for XX: 0x02. Note that value 0x00 disables the watch-dog functionality when confirmed uplinks are used.
2	-	XX – Watch-dog period (WDP) when unconfirmed uplinks are used by the device. $WDP_{unconfirmed}, [min] = XX * 60$ Default value for XX: 0x18. Note that value 0x00 disables the watch-dog functionality when unconfirmed uplinks are used.

Table 13

Example command, [Hex]: 1C0300 – Assuming that the send Keep-alive period is 5 minutes, $WDP_{confirmed}$ = 22 minutes, but the watch-dog functionality for unconfirmed messages is totally disabled.

2.10 Get device radio communication watch-dog parameters command explanation

This command is used to get the radio watch-dog configurations. The command is described in Table 14. The keep-alive in the response is omitted for clarity.

Byte index	Bit index	Hex value – Meaning	
		Sent request	Received response
0	-	1D – Command code.	1D – The command code.
1	-		$WDP_{confirmed}$ value, as described in Table 13.
2	-		$WDP_{unconfirmed}$ value, as described in Table 13.

Table 14

2.11 Device buzzer control command explanation

This command is used to control the device buzzer. It's described in Table 15.

Byte index	Bit index	Hex value – Meaning
0	-	03 – The command code.
1	7:4	Buzzer volume: 0x0: Buzzer volume set to minimum available; 0x1: … 0x2: … … 0xE: Buzzer volume set to maximum available; 0xF: Buzzer is off.
	3:0	Buzzer frequency: 0x0: Buzzer frequency is 1kHz; 0x1: Buzzer frequency is 1.5kHz; 0x2: Buzzer frequency is 2kHz; … 0xA: Buzzer frequency is 6kHz; 0xB: Reserved; 0xC: Reserved; … 0xF: Reserved.
2	-	Time the buzzer to be active. Resolution – 1s. If zero, the buzzer will stay active until buzzer command with volume 0xF is received (buzzer turn-off) or the device button is pressed. During this time the buzzer continuously alternate loud and silent states.
3	-	On time from the buzz loud-silent period. Resolution – 10ms.
4	-	Off time from the buzz loud-silent period. Resolution – 10ms.

Table 15

Example command, [Hex]: 03E60A3264 – commands the buzzer as follows:

- Volume: max. available;
- Frequency: 4kHz;
- Time the buzzer to be active: 10s;
- On time from the buzz loud-silent period: 500ms.
- Off time from the buzz loud-silent period: 1000ms.

2.12 Device LED's control command explanation

This command is used to control the device LED's. It's described in Table 16.

Byte index	Bit index	Hex value – Meaning
0	-	05 – The command code.
1	7:5	**Red** LED behavior[1].

	4:0	**Red** LED duration of the specified behavior[2].
2	7:4	**Green** LED behavior[1].
	3:0	**Green** LED duration of the specified behavior[2].
3	7:4	**Blue** LED behavior[1].
	3:0	**Blue** LED duration of the specified behavior[2].

Table 16

[1] Bits 7:5 values:
1: Turn the LED ON.
2: Blink fast (10ms on, 200ms off).
3: Blink slow (10ms on, 2000ms off).
4: Turn the LED OFF.

[2] The LED behavior duration can be zero. In this case, the specified LED behavior will be available until it's changed from new radio command or the device button is pressed. If bits[4:0] are non-zero, the specified LED duration is calculated by the expression: $\text{Duration, [s]} = \text{bits}[4:0] * 10$. The duration matters in case the LED behavior isn't turned off.

Example command, [Hex]: 05460046 – commands the device LED's as follows:
Red LED: Blink fast for 1 minute;
Green LED: Turn off;
Blue LED: Blink fast for 1 minute.

3 Appendix

3.1 Command examples.

Example containing more than 1 command for sent/received server packet uses alternate colors for each command (and surrounding command data) for ease of understanding.

3.1.1 Command example 1

This example shows sent and received server message. The sent message contains single command for the end-node device – read software and hardware version (Command code 0x04). The next received server message will contain 2 commands from the device:

- The response to command code 0x04;
- The standard periodically sent keep-alive command with its data.

Server sends, [hex]: 04.
Server receives, [hex]: 041080010275 49CD0180.

3.1.2 Command example 2

This example shows sent and received server packet which consists of more commands. The sent packet consists of 8 end-node commands:

- Set keep-alive period (Command code 0x02) to 4 minutes;
- Set network join time (Command code 0x10) to 5 minutes (300s);
- Set uplink messages (Command code 0x11) type to unconfirmed;
- Disable LoRaWAN radio watch-dog (Command code 0x1C) for both confirmed and unconfirmed uplinks;
- Get keep-alive period (Command code 0x12);
- Get network join time (Command code 0x19);
- Get uplink messages type (Command code 0x1B);
- Get LoRaWAN radio watch-dog configurations (Command code 0x1D).

The next received packet from the server contains the requested commands answers and a keep-alive command data. Note that the order of the received command responses isn't equal to the order of the sent requests.

Server sends, [hex]: 0204103C11001C000012191B1D.
Server receives, [hex]: 12041B00193C1D000001027549CD0180.

4 Revision history.

Date	Version	Author	Comment
2020	1.0	Martin Peevski	Initial draft.
20.11.2020	1.1	Martin Peevski	Chapters 2.1 and 3.1 updated.
17.12.2020	1.2	Martin Peevski	Added commands for device buzzer and LED's control.