1 Introduction

The content of this document is the payload description of the Miromico RoomSensor.

2 General Payload Format

The payload of up and downlink messages consists of an arbitrary number of data structs (DS) of different types and lengths.

DS 1	DS 2	...	DS n

Each data struct is a combination of a header and the actual data payload

L	T	payload

The header constists of

Field	Desc
L	Length of data struct, 1 byte, not including the length byte itself
T	Data struct type

The payload of the package contains the acutal data. Data encoding is little endian.

3 Uplink Message Types

All uplink messages are sent on LoRaWAN port 15.

3.1 Temperature Data

The measurement data message type is used to upload a number of temperature and humidity measurement values. It has type T = 0x01 and the length is dependent on the number of accumulated measurements: L = 1 + 3 x N_{meas}. Each measurement consists of three bytes:

Byte	0:2	3:5	6:8	...	$3xN_{meas}$:$3xN_{meas}$+1
Field	M1	M2	M3	...	MN_{meas}

A single measurement is structured as follows:

Byte	0:1	2
Field	Temperature T [0.01 °C] (little endian)	Relative Humidity RH [0.5 %]

Temperature T is transmitted in the first two bytes in little endian format with 0.01 °C Precision.

The humidity is transmitted in the third byte with 0.5 % precision.

The number of value pairs per message is depending on the sensor configuration. The oldest measurement value is transmitted first. The last value is taken just before transmission. As the measurement interval configuration is known, the timestamps of all other pairs can be computed upon reception.

In case of a sensor failure, the payload will be 0xFFFFFF

Example:

07:01:C4:09:78:F8:09:77

Message Length: 7 → N_{meas} = 2

Message Type: 0x01 → Measurement Data

First Mesurement:

Temperature: 25 °C

Humidity: 60 %

Second Measurement:

Temperature: 25.5 °C

Humidity: 59.5 %

3.2 CO_2 Data 04011809400302ea01050302be0000

The CO_2 data message type is used to upload a number of CO_2 measurement values. It has type T=0x02 and the length is dependent on the number of accumulated measurements: L = 1 + 2 * N_{meas}. Each measurement consists of two bytes:

Byte	**0:1**	**2:3**	**4:5**	**...**	**$2xN_{meas}$:$2xN_{meas}$+1**
Field	M1	M2	M3	...	MN_{meas}

A single measurement is structured as follows:

Bit	**0:7**	**8:15**
Field	MSB CO_2 concentration in ppm	LSB CO_2 concentration in ppm

One special value is used:

CO_2 = 0 ppm → CO_2 sensor readout has failed. If this happens multiple times in a row, it most likely means the sensor is broken or not connected properly.

The number of values message is depending on the sensor configuration. The oldest measurement value is transmitted first. The last value is taken just before transmission. As the measurement interval configuration is known, the timestamps of all other pairs can be computed upon reception.

3.3 Settings Interval

This uplink will transmit the current device settings related uplink interval and retransmissions. It happens right after the sensor has joined a network and when the settings have been changed through a downlink. It will always be transmitted as a confirmed message. Retransmissions happen immediately after the NACK has been received. It has type T = 0x05 and length L = 0x05

Byte	0:1	2	3
Field	Measurement Interval	Send Interval	Flags & NbRetrans

Measurement Interval MI	Bits[0:15]	Interval of temperature and relative humidity measurements in seconds. Uses little endian format. Default: 900 (15 min)
Send Cycle SC	Bits[0:7]	This value is used to determine the number of measurements which are accumulated before sending under normal circumstances: N_{meas} = SC + 1 Default: 3
Flags	Bits[6:7]	Bit 6: 0 → LED off, 1 → LED on (Default: 0) Bit 7: 0 → Unonfirmed Uplinks, 1 → Confirmed Uplinks (Default: 0)
	Bits[4:6]	RFU
NbRetrans	Bits[0:3]	Number of Retransmissions in case of NACK for confirmed uplinks. Is limited to 5, even if something highter is selected (Default: 2). Total tries of uplinks for confirmed messages is 1 + NbRetrans

Example (including type and length in hex):

05:05:B0:04:04:C4

Measurement Interval: 1200 (10 min)

Send Interval: 4 (N_{meas} = 5)

Flags: LED on, CFM on

NbRetrans: 4

3.4 Settings CO_2

This uplink will transmit the current device settings related to the CO_2 sensor. It happens right after a device equipped with a CO_2 sensor has joined a network and when the settings have been changed through a downlink. It will always be transmitted as a confirmed message. Retransmissions happen immediately after the NACK has been received. It has type T = 0x06 and length L = 0x07

Byte	0:1	2:3	4:5
Field	CO_2 Measurement Period	CO_2 Subsamples	CO_2 ABC Period

CO_2 Measurement Period	Bits[0:15]	DEPRECATED! This setting was used in continous measurement mode. It is not for the energy efficient single measurement mode.
CO_2 Subsamples	Bits[0:15]	Number of subsamples per CO_2 measurement. Little endian format.
CO_2 ABC Period	Bits[0:15]	ABC calibration period in hours. Little endian format.

Example (including type and length in hex):

07:06:10:00:20:00:80:01

CO_2 Measurement Period: 16

CO_2 Subsamples: 32

CO_2 ABC Period: 384 (16 days)

3.5 Battery Consumption

The battery consumption message reports the estimated accumulated current consumption

of the sensor. It has type T = 0x03 and length L = 0x05 under normal circumstances. The current consumption is reported in uAh as unsigned 32 bit value.

The estimated current consumption is an approximation taken by the sensor and computed based on estimated active current during MCU on time, LoRa transmit and receive.

Byte	0:3
Field	Current Consumtion (little endian)

Example:

05:03:10:00:00:00

Message Length: 5

Message Type: 0x03 → Battery Consumption

Current Consumption: 16 uAh

3.6 Battery Voltage

The battery voltage message reports the voltage measured by the microcontroller. It is reported in 10 mV with an offset of 170 (minimum Voltage). To calculate Volts: (x + 170)/100. It has type T = 0x0A and length L = 0x02.

Byte	0
Field	Battery Voltage

Example:

02:0A:BE

Battery Voltage: 0xBE → 190 → 3.6 V

3.7 Firmware Hash

The firmware hash message transmits the git hash of the used firmware right after joining. This can be used to identify the exact firmware version and help debug issues in case they arise on the field. Additionally, some features may only be available on newer firmware, as identified by this uplink. It has type T = 0x0B and length L = 0x05.

Byte	0:3
Field	Firmware Hash

Example:

05:0B:03:89:A2:B9

Firmware Hash: 0xB9A28903

4 Downlink Message Types

Downlink messages are used to change the configuration of the device. They use the same general payload format as uplinks and must be sent on the LoRaWAN port 3.

4.1 Interval Configuration

The interval configuration message type is used to set the measurement interval MI and send cycle SC. It has type T = 0x80 and length L = 0x05.

Byte	0:1	2	3
Field	Measurement Interval	Send Interval	Flags

Measurement Interval	Bits[0:15]	Interval of temperature and relative humidity measurements in seconds. Uses little endian format. Default: 900 (15 min)
Send Cycle	Bits[0:7]	This value is used to determine the number of measurements which are accumulated before sending under normal circumstances: N_{meas} = SC + 1 Default: 3
Flags	Bits[6:7]	Bit 6: 0 → LED off, 1 → LED on (Default: 0) Bit 7: 0 → Unconfirmed Uplinks, 1 → Confirmed Uplinks (Default: 0)

MI is 16 bit measurement interval in seconds. SC is 8 bit send interval in measurement cycles. SC = 0 means every measurement is directly transmitted. SC = 1 means, two data packets are aggregated and transmitted in one message. Up to 57 messages (SC = 56) can be aggregated, but depending on the current datarate, the messages might be transmitted earlier to comply with duty cycle limitations. The flags are currently used to disable and enable confirmed messages (CFM) and the LED, using the two most significant bits.

Example (including type and length in hex):

05:80:94:02:02:C0

Measurement Interval: 660 (11 min)

Send Interval: 2 (N_{meas} = 3)

Flags: LED on, CFM on

3.8 CO_2 Configuration

This Downlink will configure the current device settings related to the CO_2 sensor, if one is available on the device. It has type T = 0x81 and length L = 0x07

Byte	0:1	2:3	4:5
Field	CO_2 Measurement Period	CO_2 Subsamples	CO_2 ABC Period

CO_2 Measurement Period	Bits[0:15]	DEPRECATED! This setting was used in continous measurement mode. It is not for the energy efficient single measurement mode.
CO_2 Subsamples	Bits[0:15]	Number of subsamples per CO_2 measurement. Little endian format.
CO_2 ABC Period	Bits[0:15]	ABC calibration period in hours. Little endian format.

Example (including type and length in hex):

07:81:10:00:20:00:80:01

CO_2 Measurement Period: 16

CO_2 Subsamples: 32

CO_2 ABC Period: 384 (16 days)

4.3 Reset

This downlink can be used to trigger a manual reset. After the reset, all sensors will be reinitialized and the LoRaWAN network will be rejoined. It has type T = 0x84 and length L = 0x05.

Caution: Use this with care! If the network coverage is bad, the device might not join again.

Byte	0:3
Field	Reset Magic bytes

Example (including type and length in hex):

05:84:F9:8B:D4:19

5 Application Notes

5.1 Confirmed Messages

If and uplink is sent as a confirmed message, either through general configuration or because many measurements have been accumulated during the night, the device expects an acknowledgement (ACK) from the LoRaWAN network.

When no ACK is received, the RoomSensor will try to retransmit the message again, up to the configured amount of tries (NbRetrans). If still no ACK is received, the measurements are discarded.

In the unlikely case that the accumulated data exeeds the maximum allowed payload size (LoRaWAN regulations), the oldest measurements will be discarded and lost. As many new data points as possible will be sent with the uplink. This scenario might happen if the datarate is decreased during retries (see ADR ACK request).

5.2 ADR ACK Request

The ADR ACK request is a LoRaWAN feature to ensure that a device will stay connected to a network whe ADR is enabled.

After 5 messages without any response from the network the RoomSensor will ask the network to respond with the next uplink. If the network fails to answer for 2 uplinks, the device will lower its datarate, to increase the chances of a message getting to the gateway. This continues, until either the network answers and the initial counter is reset, or 2 uplinks are sent with datarate 0 (SF12). If the network still has not answered, the device will reset and try to join again. With default settings, a device will approximately reset itself after not hearing from the network in one day.

This feature is useful, if a device is moved or connectivity changes. In the case that a device is reregistered in a different network, it will also try to join again on its own and no manual reset has to be performed.

Caution: As of 22.02.2021, there is a bug in the Loriot network server preventing this feature to work properly. If you experience daily resets, try to lower the ADR reactivity in Loriot (either in the device profile, or in the LoRaWAN parameters of the device directly)