

Synetica - enLink LoRaWAN decoders

License MIT

Table of Contents

- [Preamble](#)
- [Uplink Payload](#)
 - [Uplink Payload Structure](#)
 - [Sensor Details](#)
 - [Decoding Complex Messages](#)
 - [enLink KPI Payload Data](#)
- [Downlink Payload](#)
 - [Configuration Payload Structure](#)
 - [Settings Data Details](#)
 - [Downlink Message Examples](#)
 - [Downlink Message Index Tables](#)
 - [Settings for Lux Sensor](#)
 - [Settings for CO₂ Sensors](#)
 - [Uplink Replies to Downlink Messages](#)

Preamble

Synetica is a UK based company that designs and develops energy and environmental sensors. We specialise in highly accurate and reliable air quality monitoring using LoRaWAN long range wireless.

The **enLink** range of LoRaWAN devices are categorised into the following:

- Air Quality Monitors (Indoor and Outdoor, mains and battery powered)
- Indoor Environmental Sensors
- Modbus Reader - Serial RS485 RTU
- Pulse Counter
- Leak Sensor
- Differential Pressure / Air Flow
- Temperature Probes

This repository contains various decoders for the LoRaWAN data packets. The uplink data is telemetry data containing values like temperature, particulates and gas concentrations.

Uplink Payload

The enLink payload structure is designed to be as efficient as possible. Data for multiple sensor values can be concatenated into a single payload which can be easily decoded. If the payload length is restricted due to channel time limits, the whole message will be split into multiple payloads. Each payload will always be split on a **Sensor Data** boundary. This is done so each payload can be easily decoded. A payload will always have the first byte as a **Data Type Identifier**.

Uplink Payload Structure

The payload is an array of **Sensor Data** messages.



Sensor data consists of a **Data Type Identifier** byte followed by the **Data Value** as one or more bytes. The number of bytes in the data value is determined by the Data Type Identifier and is fixed. Details are here: [Sensor Details](#).

Example Payload (hexadecimal): 01 01 23 02 56 03 01 A4

These bytes can be split up as follows:



Finally, decoding the data:

| Data Type Identifier | Data Value Calculation | Result |
|----------------------|--|---------|
| 0x01 - Temperature | $((0x01 * 256) + 0x23) / 10 = (256 + 35) / 10$ | 29.1 °C |
| 0x02 - Humidity | 0x56 | 86 %rH |
| 0x03 - Ambient Light | $(0x01 * 256) + 0xA4 = 256 + 164$ | 420 Lux |

Each Data Type can use 1 or more bytes to send the value according to the following table

Sensor Details

| Type | Sensor | Sensor Range | Units | Num Bytes | Format | Scaling |
|------|----------------------------|--------------|-------|-----------|--------|---------|
| 0x01 | Temperature | -40 to 85 | °C | 2 | S16 | / 10 |
| 0x02 | Humidity | 0 to 100 | % | 1 | U8 | |
| 0x03 | Ambient Light | 0.01 to 83k | lux | 2 | U16 | |
| 0x04 | Pressure | 300 to 1100 | mbar | 2 | U16 | |
| 0x05 | Volatile Organic Compounds | 0 to 500 | IAQ | 2 | U16 | |
| 0x06 | Oxygen | 0 to 25 | % | 1 | U8 | / 10 |

| Type | Sensor | Sensor Range | Units | Num Bytes | Format | Scaling |
|------|---|---------------------------------|------------|-----------|--------|-----------------|
| 0x07 | Carbon Monoxide | 0 to 100 | ppm | 2 | U16 | / 100 |
| 0x08 | Carbon Dioxide | 0 to 2000 | ppm | 2 | U16 | |
| 0x09 | Ozone (O3) | 0 to 1 0 to 1000 | ppm ppb | 2 | U16 | / 10000 / 10 |
| 0x0A | Air Pollutants: CO, Ammonia, Ethanol, H2, Methane / Propane / Iso-Butane. | 100 to 1500 (Typ) | kΩ | 2 | U16 | / 10 |
| 0x0B | Particulate Matter 2.5 | 0 to 1000 | µg/m3 | 2 | U16 | |
| 0x0C | Particulate Matter 10 | 0 to 1000 | µg/m3 | 2 | U16 | |
| 0x0D | Hydrogen Sulphide (H2S) | 0 to 100 | ppm | 2 | U16 | / 100 |
| 0x0E | Pulse ID + Pulse Counter | ID: 0 to 3 Value: 0 to 2^32 | count | 1 + 4 | U32 | |
| 0x0F | MB ID + Modbus Exception | ID: 0 to 31 Error Num | | 1 + 1 | U8 | |
| 0x10 | MB ID + Modbus Interval value | ID: 0 to 30 Interval Value | | 1 + 4 | F32 | |
| 0x11 | MB ID + Modbus Cumulative value | ID: 0 to 31 Cumulative Value | | 1 + 4 | F32 | |
| 0x12 | bVOC – Breath VOC estimate equivalent | | ppm | 4 | F32 | |
| 0x13 | Detection count (PIR etc.) | | count | 4 | U32 | |
| 0x14 | Total occupied time | | seconds | 4 | U32 | |
| 0x15 | Occupied Status | 0 = Unoccupied 1 = Occupied | status | 1 | U8 | |
| 0x16 | Liquid Level Status | 0 = No Liquid 1 = Detected | status | 1 | U8 | |
| 0x17 | Probe 1 Temperature | -55 to 125 | °C | 2 | S16 | / 10 |
| 0x18 | Probe 2 Temperature | -55 to 125 | °C | 2 | S16 | / 10 |
| 0x19 | Probe 3 Temperature | -55 to 125 | °C | 2 | S16 | / 10 |
| 0x1A | Time temperature probe 1 has spent in 'in band' zone | | seconds | 4 | U32 | |
| 0x1B | Time temperature probe 2 has spent in 'in band' zone | | seconds | 4 | U32 | |

| Type | Sensor | Sensor Range | Units | Num Bytes | Format | Scaling |
|------|---|--------------|---------|-----------|--------|---------|
| 0x1C | Time temperature probe 3 has spent in 'in band' zone | | seconds | 4 | U32 | |
| 0x1D | Number of times in band alarm has been activated for temperature probe 1 | | count | 2 | U16 | |
| 0x1E | Number of times in band alarm has been activated for temperature probe 2 | | count | 2 | U16 | |
| 0x1F | Number of times in band alarm has been activated for temperature probe 3 | | count | 2 | U16 | |
| 0x20 | Time temperature probe 1 has spent below low threshold | | seconds | 4 | U32 | |
| 0x21 | Time temperature probe 2 has spent below low threshold | | seconds | 4 | U32 | |
| 0x22 | Time temperature probe 3 has spent below low threshold | | seconds | 4 | U32 | |
| 0x23 | Number of times low threshold alarm has been activated for temperature probe 1 | | count | 2 | U16 | |
| 0x24 | Number of times low threshold alarm has been activated for temperature probe 2 | | count | 2 | U16 | |
| 0x25 | Number of times low threshold alarm has been activated for temperature probe 3 | | count | 2 | U16 | |
| 0x26 | Time temperature probe 1 has spent above high threshold | | seconds | 4 | U32 | |
| 0x27 | Time temperature probe 2 has spent above high threshold | | seconds | 4 | U32 | |
| 0x28 | Time temperature probe 3 has spent above high threshold | | seconds | 4 | U32 | |
| 0x29 | Number of times high threshold alarm has been activated for temperature probe 1 | | count | 2 | U16 | |

| Type | Sensor | Sensor Range | Units | Num Bytes | Format | Scaling |
|------|---|-----------------------------|-------------------|-----------|--------|---------|
| 0x2A | Number of times high threshold alarm has been activated for temperature probe 2 | | count | 2 | U16 | |
| 0x2B | Number of times high threshold alarm has been activated for temperature probe 3 | | count | 2 | U16 | |
| 0x2C | Differential Pressure | +/- 5000 | Pa | 4 | F32 | |
| 0x2D | Airflow | 0 to 100 | m/s | 4 | F32 | |
| 0x2E | Voltage | 0 to 10 | Volts | 2 | U16 | / 1000 |
| 0x2F | Current | 0 to 20 | mA | 2 | U16 | / 1000 |
| 0x30 | Resistance | 0 to 10 | kΩ | 2 | U16 | / 1000 |
| 0x31 | Leak Detection (resistance rope) | 0 = No Leak 1 = Detected | status | 1 | U8 | |
| 0x3F | CO2e estimate equivalent | | ppm | 4 | F32 | |
| 0x50 | Sound Level Minimum | | dB(A) | 4 | F32 | |
| 0x51 | Sound Level Average | | dB(A) | 4 | F32 | |
| 0x52 | Sound Level Maximum | | dB(A) | 4 | F32 | |
| 0x53 | Nitric Oxide | 0 - 100 | ppm | 2 | U16 | / 100 |
| 0x54 | Nitrogen Dioxide | 0 – 5 | ppm | 2 | U16 | / 10000 |
| 0x55 | Nitrogen Dioxide | 0 – 20 | ppm | 2 | U16 | / 1000 |
| 0x56 | Sulphur Dioxide | 0 – 20 | ppm | 2 | U16 | / 1000 |
| 0x57 | Particulate matter mass concentration at PM1.0 | | µg/m ³ | 4 | F32 | |
| 0x58 | As above, PM2.5 | | µg/m ³ | 4 | F32 | |
| 0x59 | As above, PM4.0 | | µg/m ³ | 4 | F32 | |
| 0x5A | As above, PM10.0 | | µg/m ³ | 4 | F32 | |
| 0x5B | Particulate matter number concentration at PM0.5 | | #/cm ³ | 4 | F32 | |
| 0x5C | As above, PM1.0 | | #/cm ³ | 4 | F32 | |
| 0x5D | As above, PM2.5 | | #/cm ³ | 4 | F32 | |
| 0x5E | As above, PM4.0 | | #/cm ³ | 4 | F32 | |

| Type | Sensor | Sensor Range | Units | Num Bytes | Format | Scaling |
|------|---|--------------|-------------------|-----------|--------|---------|
| 0x5F | As above, PM10.0 | | #/cm ³ | 4 | F32 | |
| 0x60 | Particulate matter typical particle size | | µm | 4 | F32 | |
| 0x61 | Gas ID + Gas Concentration | | ppb | 1 + 4 | F32 | |
| 0x62 | Corrosion: Metal ID + Metal Thickness | ~ 1000nm | nm | 1 + 4 | F32 | |
| 0x63 | Corrosion: Metal ID + Minimum thickness | | nm | 1 + 2 | U16 | |
| 0x64 | Corrosion: Metal ID + Original thickness | | nm | 1 + 2 | U16 | |
| 0x65 | Corrosion: percentage of thickness between original thickness (100%) and minimum (0%) | | % | 1 + 4 | F32 | |
| 0x66 | Gas ID + Gas Concentration | | µg/m ³ | 1 + 4 | F32 | |
| 0x67 | Outdoor EPA Index Sensor Fast AQI (reading taken over 1 minute) | 0 to 500 | AQI | 2 | U16 | |
| 0x68 | Outdoor EPA Index Sensor EPA AQI See: AirNow Technical Doc | 0 to 500 | AQI | 2 | U16 | |

Decoding Complex Messages

Most sensor data values are self-explanatory, additional information for decoding more complex sensor data is given in the sections below.

Modbus - Types: 0x0F, 0x10, 0x11

The enLink Modbus data types for Interval and Cumulative values use 5 bytes to encode the item index and value.

- Modbus Exception – standard Modbus exception codes, e.g. Code 2 – Illegal Data Address.
- Modbus Interval Value – for Modbus data types which do not accumulate, e.g. Voltage, Current, Temperature etc.
- Modbus Cumulative Value – for Modbus data types which are linked to a value which accumulates, e.g. kWh, Volume etc.

The first byte indicates which of the 32 available Modbus items is being accessed (0 to 31), followed by the Modbus Value represented as a 32 bit floating point value (IEEE754 format). Interval Value types are used for instantaneous values, such as Voltage, Current, Temperature, Pressure etc. Cumulative Values are used for items such as energy consumption and total volume.

Example Modbus Payload (hexadecimal): 10 04 41 BC 7A E1

Payload Data: 10 04 41 BC 7A E1

This is an interval data value, from configured item number 5. The value is 23.56.

For an online converter, see [Hex to Float Converter](#)

Gas Readings – Types: 0x61, 0x66

The full message is sent as 6 bytes. For example:

Payload (hexadecimal): 61 19 41 BC 7A E1

Payload Data: 61 19 41 BC 7A E1

This translates to Gas Type 0x19 or 25 which is **Carbon Monoxide**. The value is 23.56ppb.

The Gas types are listed here:

| | |
|--|---|
| 0x17 - Formaldehyde - HCHO / CH ₂ O | 0x1E - Hydrogen Cyanide - HCN |
| 0x18 - Volatile Organic Compounds | 0x1F - Hydrogen Fluoride - HF |
| 0x19 - Carbon Monoxide - CO | 0x20 - Ammonia - NH ₃ |
| 0x1A - Chlorine - Cl ₂ | 0x21 - Nitrogen Dioxide - NO ₂ |
| 0x1B - Hydrogen - H ₂ | 0x22 - Oxygen - O ₂ |
| 0x1C - Hydrogen Sulphide - H ₂ S | 0x23 - Ozone - O ₃ |
| 0x1D - Hydrogen Chloride - HCl | 0x24 - Sulphur Dioxide / Sulfur Dioxide (IUPAC) - SO ₂ |

Corrosion – Types: 0x62, 0x63, 0x64, 0x65

The full message is sent as 6 bytes. The second byte indicates the coupon and sacrificial metal of the sensor.

Payload (hexadecimal): 62 01 44 58 D0 27

Payload Data: 62 01 44 58 D0 27

The example shows Coupon #1 is Copper and the thickness is 867.252 nanometres (equivalent to 8672.52 Ångströms).

Other Coupon/Metal types are:

| Coupon #1 | Coupon #2 |
|------------------------------|------------------------------|
| 0x00 - Unknown Metal / Error | 0x80 - Unknown Metal / Error |
| 0x01 - Copper | 0x81 - Copper |
| 0x02 - Silver | 0x82 - Silver |

Coupon #1**Coupon #2****0x03** - Chromium**0x83** - Chromium**enLink KPI Payload Data**

Each enLink end-node device can have optional Key Performance Indicators (KPI) added to the payload message. Each KPI can use 1 or more bytes to send the value according to the following table.

| Type | KPI | Comments | Units | Num Bytes | Format |
|-------------|----------------------|---|---------|-----------|---------|
| 0x40 | CPU Temperature | Packed Byte. See JS Code | °C | 2 | S16 |
| 0x41 | Battery Status | 0=Charging; 1 - 254 (1.8 - 3.3V); 255=Ext Power | status | 1 | U8 |
| 0x42 | Battery Voltage | 0 -> 3600 mV (3600=Ext Power) | mV | 2 | U16 |
| 0x43 | RX RSSI | Received Signal Strength | dBm | 2 | S16 |
| 0x44 | RX SNR | Received Signal-Noise Ratio | dB | 1 | S8 |
| 0x45 | RX Count | Downlink message count | count | 2 | U16 |
| 0x46 | TX Time | Time to send message | ms | 2 | U16 |
| 0x47 | TX Power | Transmit power | dBm | 1 | S8 |
| 0x48 | TX Count | Uplink message count | count | 2 | U16 |
| 0x49 | Power up count | Number of times unit powered up | count | 2 | U16 |
| 0x4A | USB insertions count | Number of times USB activated | count | 2 | U16 |
| 0x4B | Login OK count | Successful logon count | count | 2 | U16 |
| 0x4C | Login fail count | Failed logon count | count | 2 | U16 |
| 0x4D | Fan runtime | Total time the air intake fan has run (AIR models only) | seconds | 4 | U32 |
| 0x4E | CPU Temperature | New from Ver: 4.9 | °C | 2 | S16 /10 |

Example code for different LoRaWAN Network Servers (LNS) is including in the folders on this site.

Node-RED : 192.168.1.211 - Mozilla Firefox

Node-RED : 192.168.1.211 x

https://192.168.1.211:1880/#

Node-RED

filter nodes

Synetica Send Files Send SMS / Email Web page

input

- inject
- catch
- status
- link
- digital gpio
- analog gpio
- mqtt
- http

lora connected

enLink Decoder

msg.human_readable

info debug

all flows current flow

```
21/04/2020, 14:18:08 30dab6a.0c64cc2
msg.human_readable : string [348]
{ "eui": "00-04-a3-0b-00-06-05-9d", "short_eui": "06-05-9d", "temperature_c": 23.8,
"temperature_f": 74.84, "humidity": 35, "lux": 734, "pressure_mbar": 1009, "iaq":
441, "bvoc": "271.333", "co2e_ppm": "4415.33", "co2_ppm": 568, "det_count": 295,
"det_occ_time_s": 47177, "batt_volt": 3.346 }

21/04/2020, 14:18:18 30dab6a.0c64cc2
msg.human_readable : string [1200]
{ "eui": "00-04-a3-0b-00-04-0b-c4", "short_eui": "04-0b-c4", "mb_int_val": [ [ 0,
"52.62" ], [ 3, "52.62" ], [ 6, "52.62" ], [ 9, "52.62" ], [ 12, "52.62" ], [ 15, "52.62" ], [
16, "52.62" ] ], "mb_cum_val": [ [ 1, "50.04" ], [ 2, "703.79" ], [ 4, "50.04" ], [ 5,
"703.91" ], [ 7, "50.05" ], [ 8, "703.91" ], [ 10, "50.05" ], [ 11, "703.96" ], [ 13, ...

21/04/2020, 14:18:35 30dab6a.0c64cc2
msg.human_readable : string [1123]
```

Screenshot of example using NodeRED

Downlink Payload

Downlink payloads are sent to re-configure the device. When the device processes the payload, it acknowledges the message by transmitting an ACK/NACK and the identifier code. This is to notify the user that the message has been received. There is an extra decoder example that can decode the ACK/NACK messages that are sent from the end-node to the LNS.

Configuration Payload Structure

| Header | Msg Len | Command | Value |
|--------|---------|---------|----------------|
| 1 byte | 1 byte | 1 byte | <i>n</i> bytes |

The header byte is always **0xA5**.

Msg Len is the number of bytes in the settings data. The settings data starts with a **Command** byte and then the command **Value**. The Value can be blank.

Settings Data Details

| Name | Msg Len | Command | Value | Reboot Required? |
|-------------------------|---------|-------------|---|------------------|
| Reboot | 1 | 0xFF | | |
| Public Network | 2 | 0x02 | 0/1 (Disable/Enable) | Yes |
| AppEUI | 9 | 0x05 | 8 Bytes for the EUI | Yes |
| AppKey | 17 | 0x06 | 16 bytes for the Key | Yes |
| Auto Data Rate (ADR) | 2 | 0x07 | 0/1 (Disable/Enable) | |
| Duty Cycle | 2 | 0x08 | 0/1 (Disable/Enable) | |
| Message Confirmation | 2 | 0x09 | 0/1 (Disable/Enable) | |
| Transmit Port | 2 | 0x0A | 1 to 223 | |
| Default Data Rate Index | 2 | 0x0B | 1 to 6 (Requires ADR disabled) | |
| Transmit Interval Index | 2 | 0x0C | 1 to 10 | |
| Transmit Power Index | 2 | 0x0D | 1 to 6 | |
| Receive Port | 2 | 0x0E | 0 to 223 (0 indicates All Ports) | |

The following are used in the AQM/Air, Zone and ZonePlus (with Light Sensor)

| Name | Msg Len | Command | Value | Scaling |
|----------------------|---------|-------------|--------------------------|--|
| Lux Scale Parameter | 3 | 0x20 | 0 to 65535 | /1000 (0xFFFF represents 65.535) |
| Lux Offset Parameter | 3 | 0x21 | 0 to 65535 | None (0xFFFF represents 65535) |

The following are used in the AQM/Air

| Name | Msg Len | Command | Value |
|--|---------|---------|-------------------|
| Case Fan Run Time | 3 | 0x22 | 10 to 600 Seconds |
| HPM Particulate Fan Run Time (Discontinued) | 3 | 0x23 | 10 to 60 Seconds |

The following are used in devices with CO₂ sensor

| Name | Msg Len | Command | Value |
|---|---------|---------|----------------------|
| Enable/Disable Auto-Calibration | 2 | 0x24 | 0/1 (Disable/Enable) |
| Set Target CO ₂ Level | 3 | 0x25 | 100 to 1000 ppm |
| Set to Known CO ₂ Level | 3 | 0x26 | 10 to 2000 ppm |
| Reset to factory Calibration Only Sunrise model | 1 | 0x27 | |
| Set Regular Auto-Cal Interval | 3 | 0x28 | 24 to 8760 hours |
| Set the Out-of-Bounds limits Only GSS model | 3 | 0x29 | 10 to 5000 ppm |
| Set initial auto-cal interval Only GSS model | 3 | 0x2A | 1 to 8760 hours |

Downlink Message Examples

Reboot

Payload Data:

| | | |
|----|----|----|
| A5 | 01 | FF |
|----|----|----|

Enable Message Confirmation

Payload Data:

| | | |
|----|----|-------|
| A5 | 02 | 09 01 |
|----|----|-------|

Downlink Message Index Tables

The Indexes for some settings depend on the region the unit is programmed for.

0x0B - Data Rate Index

| Index | EU868 | Index | US915 Hybrid |
|-------|----------------|-------|----------------|
| 0 | DR0 SF12 BW125 | 0 | DR0 SF10 BW125 |
| 1 | DR1 SF11 BW125 | 1 | DR1 SF9 BW125 |
| 2 | DR2 SF10 BW125 | 2 | DR2 SF8 BW125 |
| 3 | DR3 SF9 BW125 | 3 | DR3 SF7 BW125 |

| Index | EU868 | Index | US915 Hybrid |
|-------|---------------|-------|---------------|
| 4 | DR4 SF8 BW125 | 4 | DR4 SF8 BW500 |
| 5 | DR5 SF7 BW125 | | |

0x0C - Transmit Interval Index

| Index | Transmit Interval | Message |
|-------|-------------------|-------------|
| 1 | 30 s | A5 02 0C 01 |
| 2 | 1 min | A5 02 0C 02 |
| 3 | 2 min | A5 02 0C 03 |
| 4 | 5 min | A5 02 0C 04 |
| 5 | 10 min | A5 02 0C 05 |
| 6 | 15 min | A5 02 0C 06 |
| 7 | 20 min | A5 02 0C 07 |
| 8 | 30 min | A5 02 0C 08 |
| 9 | 1 hour | A5 02 0C 09 |
| 10 | 2 hours | A5 02 0C 0A |
| 11 | 3 hours | A5 02 0C 0B |

0x0D - Transmit Power Index

| Index | EU868 | Index | US195 Hybrid |
|-------|--------|-------|--------------|
| 1 | 16 dBm | 6 | 20 dBm |
| 2 | 14 dBm | 7 | 18 dBm |
| 3 | 11 dBm | 8 | 16 dBm |
| 4 | 9 dBm | 9 | 14 dBm |
| 5 | 8 dBm | 10 | 12 dBm |
| 6 | 6 dBm | 11 | 10 dBm |
| 7 | 4 dBm | | |
| 8 | 2 dBm | | |

Settings for Lux Sensor

To scale the lux reading to compensate for the enclosure light pipe, a scaling factor is applied to the sensor value:

$$\text{Adjusted_Reading} = (\text{Sensor_Value} \times \text{Scale}) + \text{Offset}$$

Defaults are:

- Scale = **2.0** (AQM/AIR), **1.678** (Zone and ZonePlus)
- Offset = **0** (All devices)

For example, set Scale to **12.345** (12345 in hexadecimal is **0x3039**)

Message is: **A5 03 20 30 39**

Settings for CO₂ Sensors

To Enable Auto-Calibration:

Message is: **A5 02 24 01**

To set the auto-calibration target to 450ppm

Message is: **A5 03 25 01 C2**

To set the sensor to known CO₂ concentration of 780ppm (**0x030C**)

Message is: **A5 03 26 03 0C**

To reset the sensor back to factory calibration (Sunrise Only)

Message is: **A5 01 27**

To set the auto-calibration interval to 10 days (240 hours, 0x00F0)

Message is: **A5 03 28 00 F0**

Uplink Replies to Downlink Messages

Successfully changed the Message Confirmation Option - **ACK**

Return code: **A5 06 09**

Failed to change the Transmit Port - **NACK**

Return code: **A5 15 0A**

Example code for decoding the **Uplink replies to Downlink messages** is included in the folders on this site.