

Table of Contents

1. Introduction	6
1.1 What is LTC1/LTC2-NB/NS NB-IoT Temperature Transmitter	6
1.2 Features	6
1.3 Specification	7
1.4 Applications	8
1.5 Sleep mode and working mode	8
1.6 Button & LEDs	8
1.6.1 For LTC1/LTC2-NB/NS Host	8
1.6.2 For LTC1 sampling module	10
1.7 BLE connection	10
1.8 Pin Definitions , Switch & SIM Direction	11
1.8.1 Jumper JP2	11
1.8.2 BOOT MODE / SW1	11
1.8.3 Reset Button	11
1.8.4 SIM Card Direction	11
1.9 Mechanical	12
1.9.1 for NB version	12
1.9.2 for NS version	12
1.10 Probe Variant	12
2. Use LTC1/LTC2-NB/NS to communicate with IoT Server	14
2.1 Send data to IoT server via NB-IoT network	14
2.2 Payload Types	15
2.2.1 General Json Format(Type=5)	16
2.2.2 General Json Format(Type=5)(Since as v1.0.2)	18
2.2.3 HEX format Payload(Type=0)	19
2.2.4 ThingsBoard Payload(Type=3)	22
2.2.5 ThingSpeak Payload(Type=1)	23

3. Configure LTC1/LTC2-NB/NS	24
3.1 Configure Methods	24
3.2 Serial Access Password	24
3.3 AT Commands Set	27
3.4 Test Uplink and Change Update Interval	29
3.5 Set the receiving time	29
3.6 Reset	29
3.7 Trigger an uplink by external interrupt	30
3.8 Set the QoS level	30
3.9 Clock logging	30
Poll sensor value	32
Datalog Uplink payload (Since firmware v1.0.2)	33
3.10 Set the TLS mode	34
3.11 Set the search network time	35
3.12 Set alarm mode	35
3.13 Alarm check time	35
3.14 Set Alarm Threshold	36
3.15 Factory data reset	36
3.16 Set CoAP option	36
3.17 Example Query saved historical records	37
3.18 Uplink log query	37
3.19 Print last few data entries	38
3.20 Print data entries base on page	39
3.21 Clear Flash Record	40
3.22 Set the downlink debugging mode	41
3.23 Domain name resolution settings	41
4. Battery & Power Consumption	42
5. Firmware update	42
6. FAQ	43
6.1 How can I access t BC660K-GL AT Commands?	43
6.2 How do users connect custom PT100 sensors?	43
6.3 How do customers connect a two-wire PT100?	45
6.4 Why is there no LED response when I press the button on the solar panel model?	47
6.5 How to fix the FT -- Flat Type probe to surface?	47
7. Order Info	48

8. Packing Info 48

9. Support 49



Table of Contents:

- [1. Introduction](#)
 - [1.1 What is LTC1/LTC2-NB/NS NB-IoT Temperature Transmitter](#)
 - [1.2 Features](#)
 - [1.3 Specification](#)

- [1.4 Applications](#)
- [1.5 Sleep mode and working mode](#)
- [1.6 Button & LEDs](#)
 - [1.6.1 For LTC1/LTC2-NB/NS Host](#)
 - [1.6.2 For LTC1 sampling module](#)
- [1.7 BLE connection](#)
- [1.8 Pin Definitions , Switch & SIM Direction](#)
 - [1.8.1 Jumper JP2](#)
 - [1.8.2 BOOT MODE / SW1](#)
 - [1.8.3 Reset Button](#)
 - [1.8.4 SIM Card Direction](#)
- [1.9 Mechanical](#)
 - [1.9.1 for NB version](#)
 - [1.9.2 for NS version](#)
- [1.10 Probe Variant](#)
- [2. Use LTC1/LTC2-NB/NS to communicate with IoT Server](#)
 - [2.1 Send data to IoT server via NB-IoT network](#)
 - [2.2 Payload Types](#)
 - [2.2.1 General Json Format\(Type=5\).](#)
 - [2.2.2 General Json Format\(Type=5\)\(Since as v1.0.2\).](#)
 - [2.2.3 HEX format Payload\(Type=0\).](#)
 - [2.2.4 ThingsBoard Payload\(Type=3\).](#)
 - [2.2.5 ThingSpeak Payload\(Type=1\).](#)
- [3. Configure LTC1/LTC2-NB/NS](#)
 - [3.1 Configure Methods](#)
 - [3.2 Serial Access Password](#)
 - [3.3 AT Commands Set](#)
 - [3.4 Test Uplink and Change Update Interval](#)
 - [3.5 Set the receiving time](#)
 - [3.6 Reset](#)
 - [3.7 Trigger an uplink by external interrupt](#)
 - [3.8 Set the QoS level](#)
 - [3.9 Clock logging](#)
 - [Poll sensor value](#)
 - [Datalog Uplink payload \(Since firmware v1.0.2\).](#)
 - [3.10 Set the TLS mode](#)
 - [3.11 Set the search network time](#)
 - [3.12 Set alarm mode](#)
 - [3.13 Alarm check time](#)
 - [3.14 Set Alarm Threshold](#)
 - [3.15 Factory data reset](#)
 - [3.16 Set CoAP option](#)
 - [3.17 Example Query saved historical records](#)
 - [3.18 Uplink log query.](#)
 - [3.19 Print last few data entries](#)
 - [3.20 Print data entries base on page](#)
 - [3.21 Clear Flash Record](#)
 - [3.22 Set the downlink debugging mode](#)
 - [3.23 Domain name resolution settings](#)

- [4. Battery & Power Consumption](#)
- [5. Firmware update](#)
- [6. FAQ](#)
 - [6.1 How can I access t BC660K-GL AT Commands?](#)
 - [6.2 How do users connect custom PT100 sensors?](#)
 - [6.3 How do customers connect a two-wire PT100?](#)
 - [6.4 Why is there no LED response when I press the button on the solar panel model?](#)
 - [6.5 How to fix the FT -- Flat Type probe to surface?](#)
- [7. Order Info](#)
- [8. Packing Info](#)
- [9. Support](#)

1. Introduction

1.1 What is LTC1/LTC2-NB/NS NB-IoT Temperature Transmitter

The Dragino LTC1/LTC2-NB/NS Industrial NB-IoT Temperature Transmitter is designed to monitor temperature for different environment. It supports to read **PT100 probe** and convert the value to temperature and then upload to IoT server via NB-IoT network*.

LTC2-NB/NS has **two internal 24-bit ADC interfaces**.

LTC1-NB/NS has **one internal 24-bit ADC interfaces**.

LTC1/LTC2-NB/NS **supports BLE configure** and **wireless OTA update** which make user easy to use.

LTC1/LTC2-NB/NS supports different uplink methods include **TCP, MQTT, UDP, MQTTs or CoAP** for different application requirement. and Support Uplinks to various IoT Servers.

LTC1/LTC2-NB/NS is powered by **8500mAh Li-SOCI2 battery or solar powered + Li-ion battery**, It is designed for long term use up to several years. (Real-world battery life depends on the use environment, update period and uplink method. Please check related Power Analyze report).

*make sure you have NB-IoT coverage locally.

1.2 Features

- NB-IoT Bands: B1/B2/B3/B4/B5/B8/B12/B13/B17/B18/B19/B20/B25/B28/B66/B70/B85 @H-FDD
- Ultra-low power consumption
- For LTC1-NB/NS: max: 1 x monitor temperature channels
- For LTC1-NB/NS: Press the button to immediately take a sample and upload a data packet.
- For LTC2-NB/NS: max: 2 x monitor temperature channels

- Support 3 -wire PT-100
- Temperature alarm
- Multiply Sampling and one uplink
- Uplink via MQTT, MQTTs, TCP, UDP or CoAP
- Support Bluetooth v5.1 remote configure and update firmware
- Uplink on periodically
- Downlink to change configure
- 8500mAh Battery for long term use (LTC1/LTC2-NB)
- Solar panel + 3000mAh Li-ion battery (LTC1/LTC2-NS)

1.3 Specification

Common DC Characteristics:

- Supply Voltage: Built-in Battery , 2.5v ~ 3.6v
- Operating Temperature: -40 ~ 85°C

Temperature Sensor:

- Range: -40 to + 80°C
- Accuracy: ± 0.2 @ 0-90 °C
- Resolution: 0.1°C
- Long Term Shift: <0.03 °C/yr

NB-IoT Spec:

NB-IoT Module: BC660K-GL

Support Bands:

- B1 @H-FDD: 2100MHz
- B2 @H-FDD: 1900MHz
- B3 @H-FDD: 1800MHz
- B4 @H-FDD: 2100MHz
- B5 @H-FDD: 860MHz
- B8 @H-FDD: 900MHz
- B12 @H-FDD: 720MHz
- B13 @H-FDD: 740MHz
- B17 @H-FDD: 730MHz
- B18 @H-FDD: 870MHz
- B19 @H-FDD: 870MHz
- B20 @H-FDD: 790MHz
- B25 @H-FDD: 1900MHz
- B28 @H-FDD: 750MHz
- B66 @H-FDD: 2000MHz
- B70 @H-FDD: 2000MHz
- B85 @H-FDD: 700MHz

Battery:

- Li/SOCI2 un-chargeable battery
- Capacity: 8500mAh
- Self Discharge: <1% / Year @ 25°C

- Max continuously current: 130mA
- Max boost current: 2A, 1 second

Power Consumption

- STOP Mode: 10uA @ 3.3v
- Max transmit power: 350mA@3.3v

1.4 Applications

- Logistics and Supply Chain Management
- Food management
- Cold chains solution
- Industrial Monitoring and Control

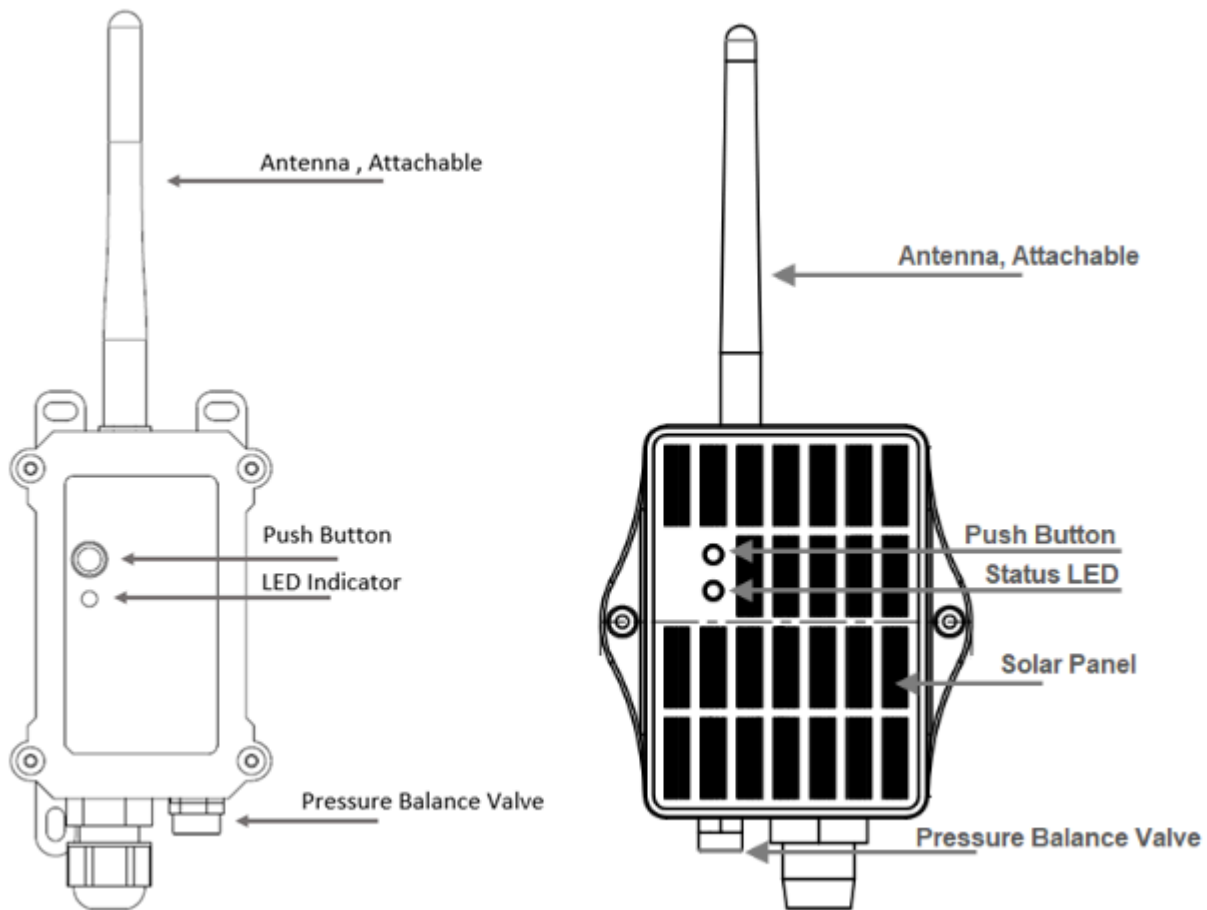
1.5 Sleep mode and working mode

Deep Sleep Mode: Sensor doesn't have any NB-IoT activate. This mode is used for storage and shipping to save battery life.

Working Mode: In this mode, Sensor will work as NB-IoT Sensor to Join NB-IoT network and send out sensor data to server. Between each sampling/tx/rx periodically, sensor will be in IDLE mode), in IDLE mode, sensor has the same power consumption as Deep Sleep mode.

1.6 Button & LEDs

1.6.1 For LTC1/LTC2-NB/NS Host



Behavior on ACT	Function	Action
  1~3s	Send an uplink	If sensor has already attached to NB-IoT network, sensor will send an uplink packet, blue led will blink once. Meanwhile, BLE module will be active and user can connect via BLE to configure device.
  >3s	Active Device	Green led will fast blink 5 times, device will enter OTA mode for 3 seconds. And then start to attach NB-IoT network. Once sensor is active, BLE module will be active and user can connect via BLE to configure device, no matter if device attach NB-IoT network or not
 x5	Deactivate Device	Red led will solid on for 5 seconds. Means device is in Deep Sleep Mode.

1.6.2 For LTC1 sampling module



Behavior on Button	Function	Action
 x1	Sample and upload	Sample data once and upload it to the platform immediately

Note:

When AT+LTCLED=1,1 is configured, the green light on the module will flash for 500 ms during sampling; and when the module receives an ACK from the server, the blue light will remain on for 2 seconds. When AT+LTCLED=0,0 is configured, the green light on the module will not flash during sampling; and when the module receives an ACK from the server, the blue light will not turn on.

1.7 BLE connection

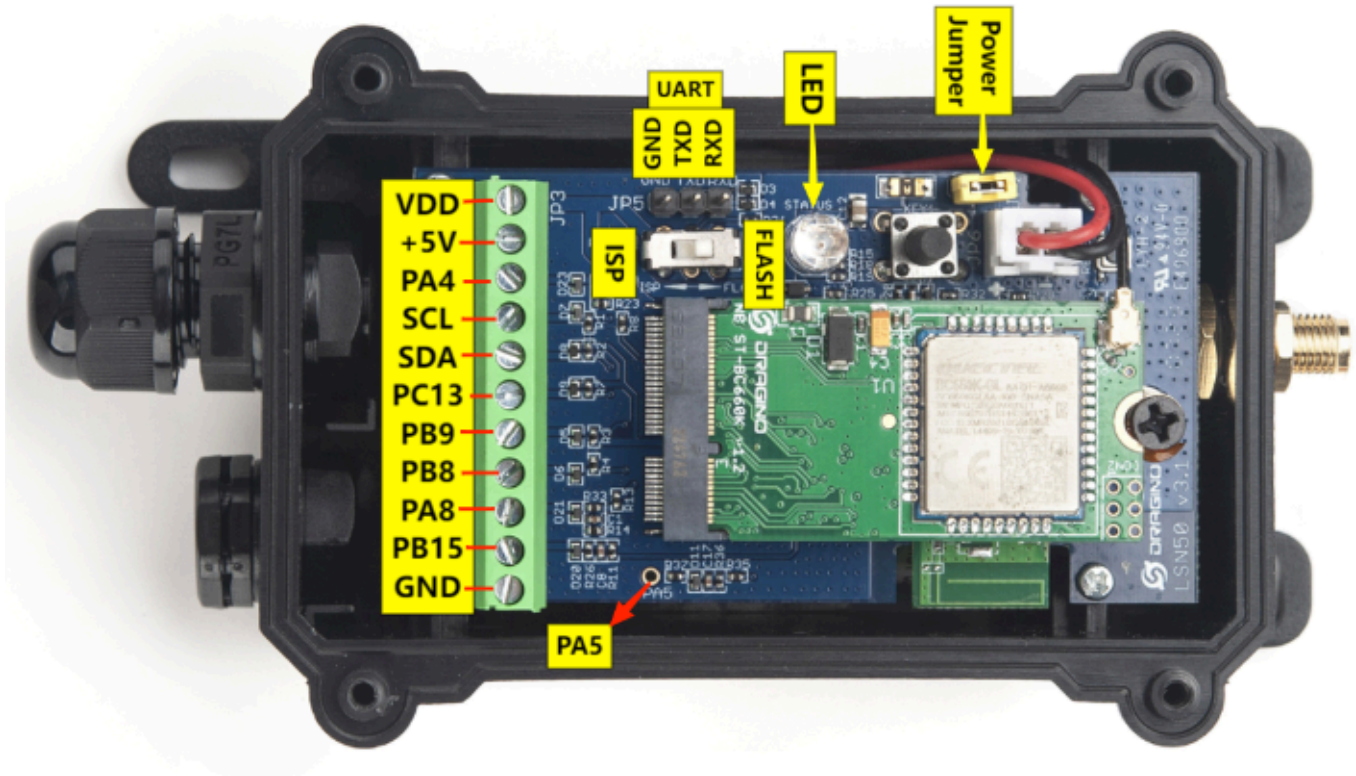
LTC1/LTC2-NB/NS support BLE remote configure and firmware update.

BLE can be used to configure the parameter of sensor or see the console output from sensor. BLE will be only activate on below case:

- Press button to send an uplink
- Press button to active device.
- Device Power on or reset.

If there is no activity connection on BLE in 60 seconds, sensor will shut down BLE module to enter low power mode.

1.8 Pin Definitions , Switch & SIM Direction



1.8.1 Jumper JP2

Power on Device when put this jumper.

Power off device when take out this jumper

1.8.2 BOOT MODE / SW1

1) **ISP**: upgrade mode, device won't have any signal in this mode. but ready for upgrade firmware. LED won't work. Firmware won't run.

2) **Flash**: work mode, device starts to work and send out console output for further debug

1.8.3 Reset Button

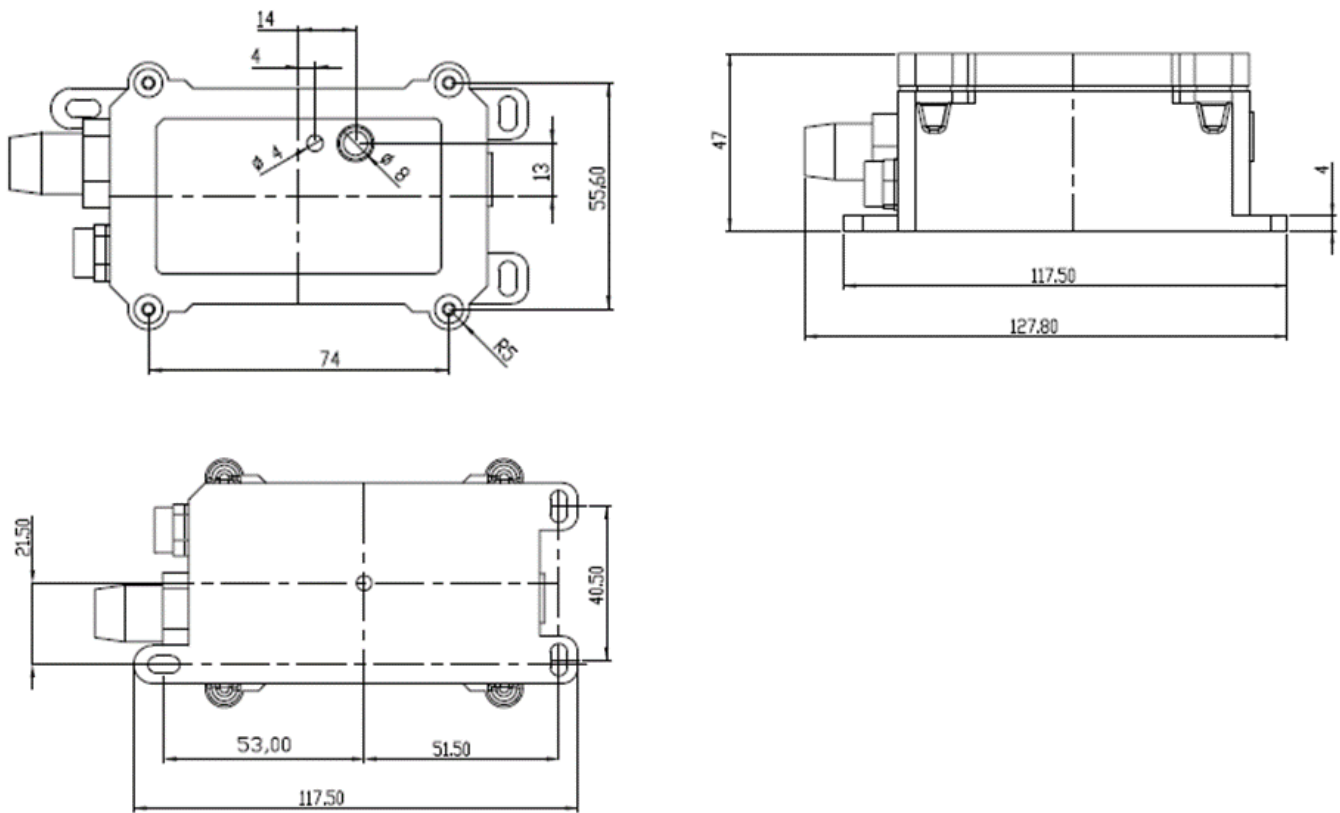
Press to reboot the device.

1.8.4 SIM Card Direction

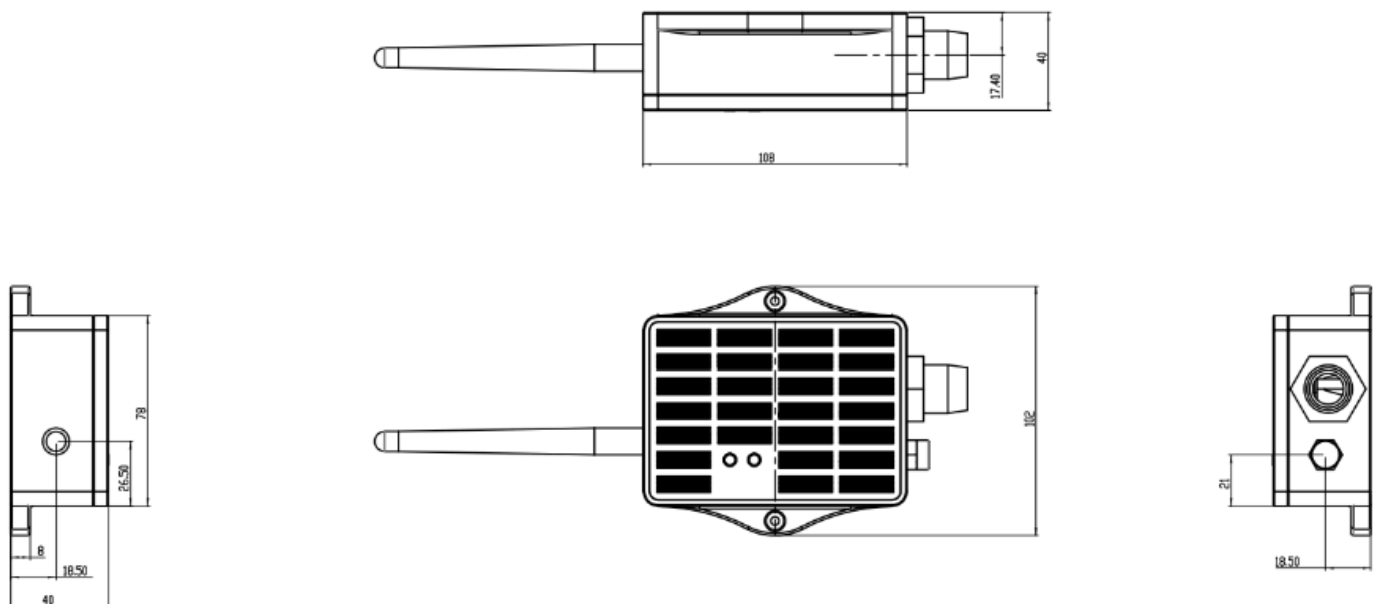
See this link. [How to insert SIM Card](#).

1.9 Mechanical

1.9.1 for NB version



1.9.2 for NS version



1.10 Probe Variant

LTC1/LTC2-NB/NS provide default probe version. See below for the variant:

Model	Photo	Description
DR-SI		Standard IP68 Probe Version: • LTC1/LTC2-NB/NS with 1 x Standard IP68 PT100 probe. • Installation: Insert • Cable Length : 2m • PT100 Class : Class A • Probe Dimension: $\varnothing 5 \times 30\text{mm}$ • Measure Range: $-60 \sim 200\text{ }^{\circ}\text{C}$ • Suitable Environment: General environment
DR-LT		Low Temperature Version: • LTC1/LTC2-NB/NS with 1 x Low Temperature PT100 probe. • Installation: Insert • Cable Length : 2m • PT100 Class : Class A • Probe Dimension: $\varnothing 5 \times 30\text{mm}$ • Measure Range: $-196 \sim 150\text{ }^{\circ}\text{C}$ • Suitable Environment: Low temperature measurement, such as COVID vaccine transport
DR-HT		High Temperature Version: • LTC1/LTC2-NB/NS with 1 x high temperature PT100 probe. • Installation: Insert • Cable Length : 3m • PT100 Class : Class A • Probe Dimension: $\varnothing 4.5 \times 30\text{mm}$ • Measure Range: $-70 \sim 450\text{ }^{\circ}\text{C}$ • Suitable Environment: High Temperature
DR-UHT		Ultra High Temperature Version: • LTC1/LTC2-NB/NS with 1 x Ultra high temperature PT100 probe. • Installation: Insert • Cable Length : 3m • PT100 Class : Class A • Probe Dimension: $\varnothing 6 \times 50\text{mm}$ • Measure Range: $-40 \sim 600\text{ }^{\circ}\text{C}$ • Suitable Environment: Ultra High Temperature

DR-FSA		Food Safety Version: • LTC1/LTC2-NB/NS with 1 x Food Safety PT100 probe. • Installation: Insert • Cable Length : 2m • PT100 Class : Class A • Probe Dimension: $\varnothing 4 \times 150\text{mm}$ • Measure Range: $-40 \sim 260\text{ }^{\circ}\text{C}$ • Suitable Environment: Food temperature measurement
DR-FT		Flat Type Version: • LTC1/LTC2-NB/NS with 1 x Flat Type PT100 probe. • Installation: Attached • Cable Length : 2m • PT100 Class : Class A • Probe Dimension: $3.6 \times 7.5 \times 20\text{mm}$ • Measure Range: $-60 \sim 200\text{ }^{\circ}\text{C}$ • Suitable Environment: Attached to the measure point.

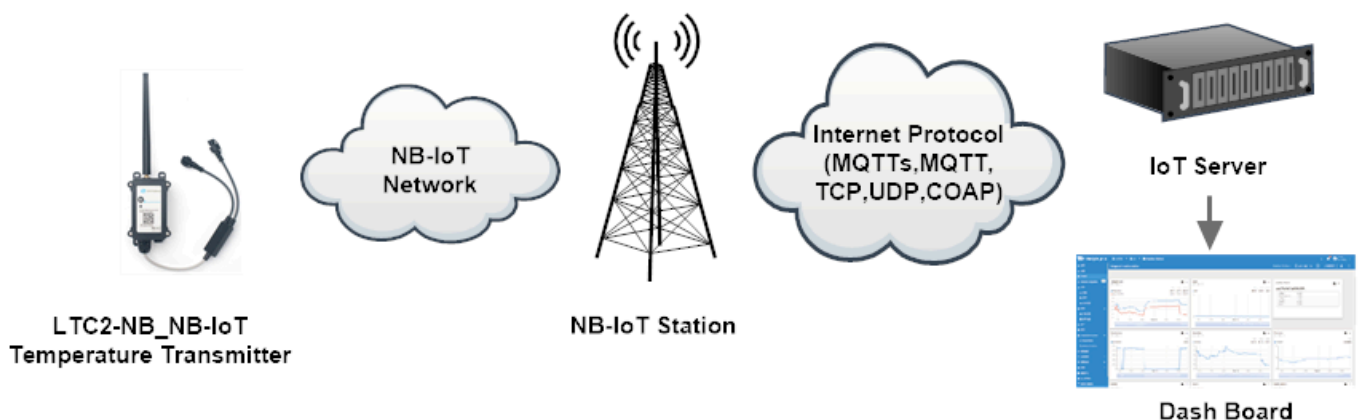
2. Use LTC1/LTC2-NB/NS to communicate with IoT Server

2.1 Send data to IoT server via NB-IoT network

The LTC1/LTC2-NB/NS is equipped with a NB-IoT module, the pre-loaded firmware in LTC1/LTC2-NB/NS will get environment data from sensors and send the value to local NB-IoT network via the NB-IoT module. The NB-IoT network will forward this value to IoT server via the protocol defined by LTC1/LTC2-NB/NS.

Below shows the network structure:

LTC2-NB in a NB-IoT Network

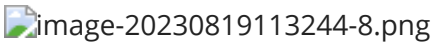
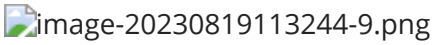
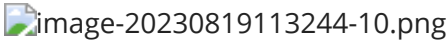
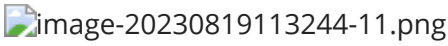


There are two version: **-GE** and **-1T** version of LTC1/LTC2-NB/NS.

GE Version: This version doesn't include SIM card or point to any IoT server. User needs to use AT Commands to configure below two steps to set LTC1/LTC2-NB/NS send data to IoT server.

- Install NB-IoT SIM card and configure APN. See instruction of [Attach Network](#).
- Set up sensor to point to IoT Server. See instruction of [Configure to Connect Different Servers](#).

Below shows result of different server as a glance.

Servers	Dash Board	Comments
Node-Red		
DataCake		
Tago.IO		
General UDP	Raw Payload. Need Developer to design Dash Board	
General MQTT	Raw Payload. Need Developer to design Dash Board	
ThingSpeak		
ThingsBoard		

1T Version: This version has 1NCE SIM card pre-installed and configure to send value to ThingsEye. User Just need to select the sensor type in ThingsEyeand Activate LTC1LTC2-NB/NS and user will be able to see data in ThingsEye. See here for [ThingsEye Config Instruction](#).

2.2 Payload Types

To meet different server requirement, LTC1/LTC2-NB/NS supports different payload type.

Includes:

- [General JSON format payload](#). (Type=5)
- [HEX format Payload](#). (Type=0)
- [ThingSpeak Format](#). (Type=1)
- [ThingsBoard Format](#). (Type=3)

User can specify the payload type when choose the connection protocol. Example:

AT+PRO=1,0 // Use COAP Connection & hex Payload
AT+PRO=1,5 // Use COAP Connection & Json Payload
AT+PRO=2,0 // Use UDP Connection & hex Payload
AT+PRO=2,5 // Use UDP Connection & Json Payload
AT+PRO=3,0 // Use MQTT Connection & hex Payload
AT+PRO=3,5 // Use MQTT Connection & Json Payload
AT+PRO=4,0 // Use TCP Connection & hex Payload
AT+PRO=4,5 // Use TCP Connection & Json Payload

2.2.1 General Json Format(Type=5)

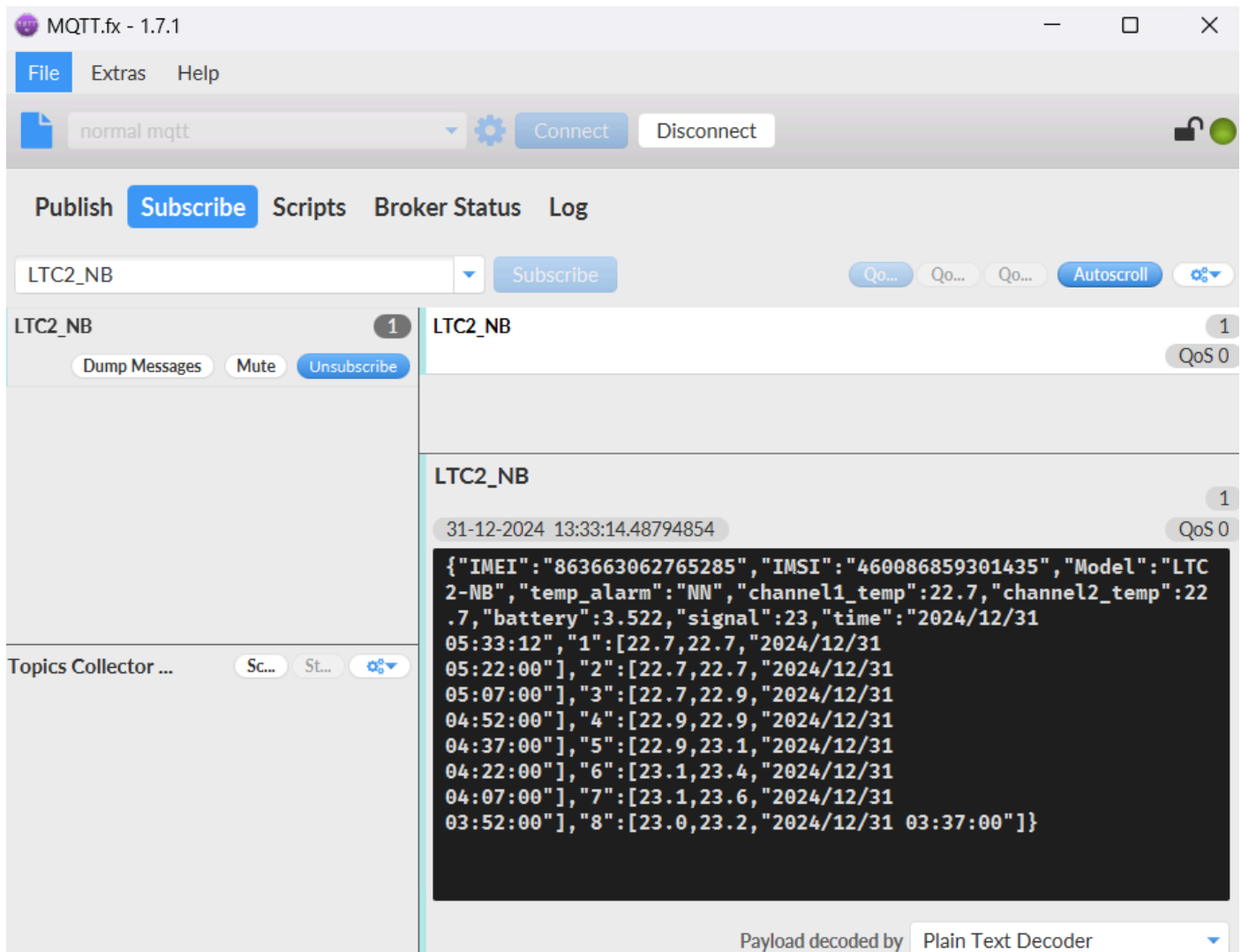
Notice:When using the LTC1-NB module, the temperature of channel 2 is invalid and is always 0.0 degrees.

This is the General Json Format. As below:

```
{
  "IMEI": "863663062765285",
  "IMSI": "460086859301435",
  "Model": "LTC2-NB",
  "temp_alarm": "NN",
  "channel1_temp": 22.7,
  "channel2_temp": 22.7,
  "battery": 3.522,
  "signal": 23,
  "time": "2024/12/31 05:33:12",
  "1": [22.7, 22.7, "2024/12/31 05:22:00"],
  "2": [22.7, 22.7, "2024/12/31 05:07:00"],
  "3": [22.7, 22.9, "2024/12/31 04:52:00"],
  "4": [22.9, 22.9, "2024/12/31 04:37:00"],
  "5": [22.9, 23.1, "2024/12/31 04:22:00"],
  "6": [23.1, 23.4, "2024/12/31 04:07:00"],
  "7": [23.1, 23.6, "2024/12/31 03:52:00"],
  "8": [23.0, 23.2, "2024/12/31 03:37:00"]}

```

Example:



Notice, from above payload:

- Temp_alarm, Channel1_temp, Channel2_temp, Battery, Signal & time are the value at uplink time.
- Json entry 1 ~ 8 are the last 1 ~ 8 sampling data as specify by **AT+CLOCKLOG=1,65535,15,8** Command. Each entry includes (from left to right): TChannel1_temp, Channel2_temp, Sampling time.

Temp_alarm:

This data field shows whether the packet was generated by a temperature alarm.

"L" stands for low temperature alarm, "H" stands for high temperature alarm.

Example:

- **NN**: No temperature alarm.
- **NH**: Channel 1 does not alarm, channel 2 high temperature alarm.
- **NL**: Channel 1 does not alarm, channel 2 low temperature alarm.
- **HH**: Both channel 1 and channel 2 have high temperature alarms.
- **LL**: Both channel 1 and channel 2 have low temperature alarms

2.2.2 General Json Format(Type=5)(Since as v1.0.2)

Notice:When using the LTC1-NB module, the temperature of channel 2 is invalid and is always 0.0 degrees.

This is the General Json Format. As below:

```
{
  "IMEI": "863663062798914",
  "IMSI": "460087023300073",
  "Model": "LTC2-NB",
  "temp_alarm": "NN",
  "channel1_temp": 27.7,
  "channel2_temp": 0.0,
  "battery": 3.357,
  "signal": 16,
  "time": "2025/09/01 06:27:00",
  "1": [27.7, 0.0, "2025/09/01 06:25:26"],
  "2": [27.7, 0.0, "2025/09/01 06:24:26"],
  "3": [27.1, 0.0, "2025/09/01 06:14:21"],
  "4": [26.4, 0.0, "2025/09/01 05:59:08"],
  "5": [26.1, 0.0, "2025/09/01 05:44:21"],
  "6": [25.6, 0.0, "2025/09/01 05:29:08"],
  "7": [25.7, 0.0, "2025/09/01 05:14:21"],
  "8": [25.9, 0.0, "2025/09/01 04:59:08"]
}
```

Example:

The screenshot shows the MQTT.fx 1.7.1 application window. The 'Subscribe' tab is selected, and the topic '2276498' is chosen. The message list shows a single message received at '01-09-2025 14:27:04.52024935'. The message payload is a JSON object with the following structure:

```
{
  "IMEI": "863663062798914",
  "IMSI": "460087023300073",
  "Model": "LTC2-NB",
  "temp_alarm": "NN",
  "channel1_temp": 27.7,
  "channel2_temp": 0.0,
  "battery": 3.357,
  "signal": 16,
  "time": "2025/09/01 06:27:00",
  "1": [27.7, 0.0, "2025/09/01 06:25:26"],
  "2": [27.7, 0.0, "2025/09/01 06:24:26"],
  "3": [27.1, 0.0, "2025/09/01 06:14:21"],
  "4": [26.4, 0.0, "2025/09/01 05:59:08"],
  "5": [26.1, 0.0, "2025/09/01 05:44:21"],
  "6": [25.6, 0.0, "2025/09/01 05:29:08"],
  "7": [25.7, 0.0, "2025/09/01 05:14:21"],
  "8": [25.9, 0.0, "2025/09/01 04:59:08"]
}
```

The 'Topics Collector (0)' tab is also visible at the bottom of the interface.

Notice, from above payload:

- Temp_alarm, Channel1_temp, Channel2_temp, Battery, Signal & time are the value at uplink time.
- Json entry 1 ~ 8 are the last 1 ~ 8 sampling data as specify by **AT+CLOCKLOG=1,65535,15,8** Command. Each entry includes (from left to right):

TChannel1_temp, Channel2_temp, Sampling time.

Temp_alarm:

This data field shows whether the packet was generated by a temperature alarm.

"L" stands for low temperature alarm, "H" stands for high temperature alarm.

Example:

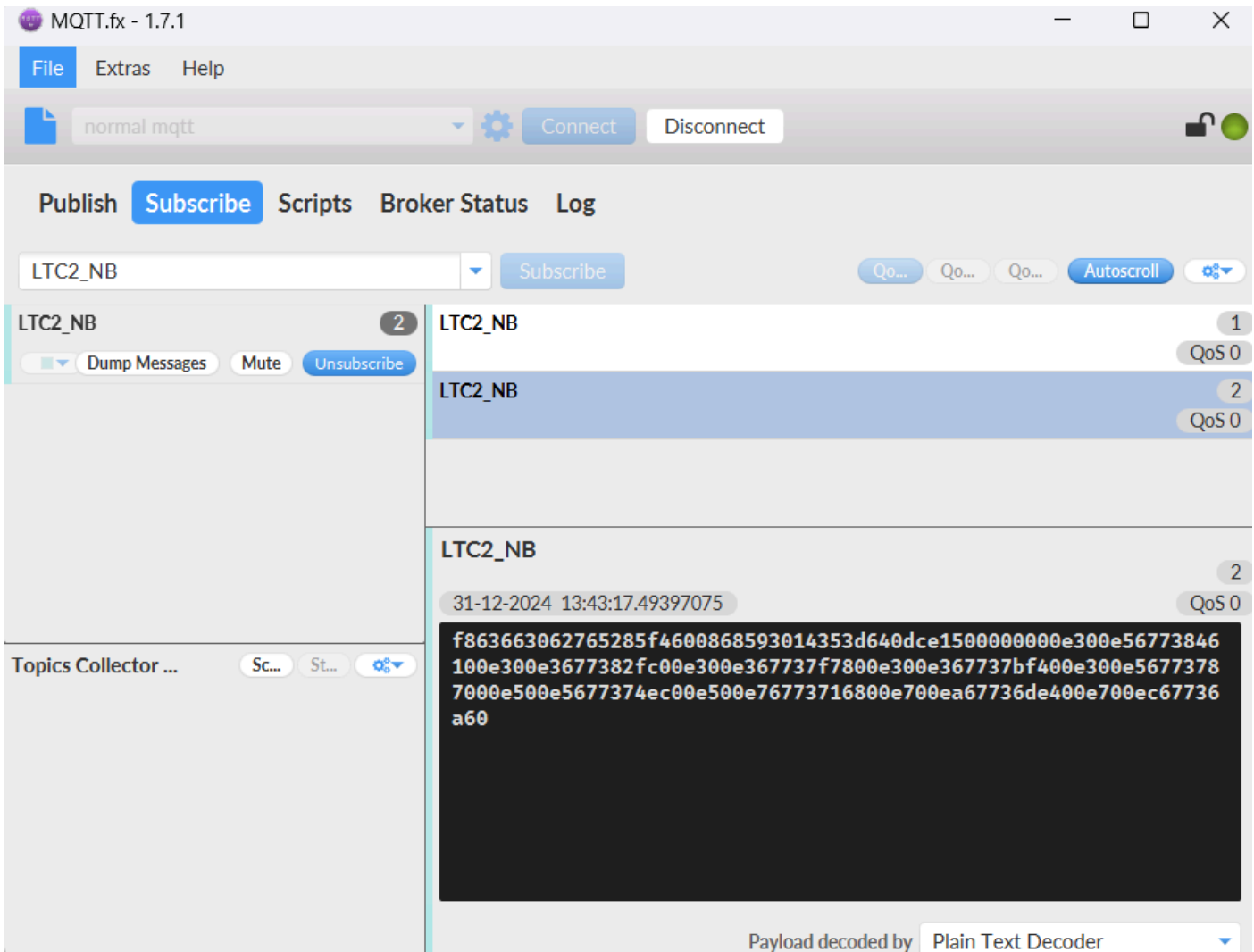
- **NN:** No temperature alarm.
- **NH:** Channel 1 does not alarm, channel 2 high temperature alarm.
- **NL:** Channel 1 does not alarm, channel 2 low temperature alarm.
- **HH:** Both channel 1 and channel 2 have high temperature alarms.
- **LL:** Both channel 1 and channel 2 have low temperature alarms

2.2.3 HEX format Payload(Type=0)

This is the HEX Format. As below:

f863663062765285f4600868593014353d640dce150000000e300e56773846100e300e3677382fc00e300e36

HEX Format for LTC2-NB(AT+CLOCKLOG=1,65535,15,8)												
f863663062765285		f460086859301435		3d64	0dce	15	00	00	00	00e3	00e5	67738461
fIMEI		fMSI		Version	BAT	signal	Interrupt	interrupt_level	temp_alarm	channel1_temp	channel2_temp	Timestamp
8 Bytes		8 Bytes		16 Bytes								
00e3	00e3	677382fc	00e300e367737f78	00e300e367737bf4				00e300e567737870				
channel1_temp	channel2_temp	Timestamp	last 2nd data	last 3rd data				last 4th data				
8 Bytes		8 Bytes		8 Bytes				8 Bytes				
00e500e5677374ec		00e500e767737168		00e700ea67736de4				00e700ec67736a60				
last 5th data		last 6th data		last 7th data				last 8th data				
8 Bytes		8 Bytes		8 Bytes				8 Bytes				



Device ID(f+IMEI): f863663062765285 =863663062765285

SIM Card ID(f+IMSI): f460086859301435 =460086859301435

Version:

These bytes include the hardware and software version.

Higher byte: Specify Sensor Model: 0x3d for LTC2-NB/NS

Lower byte: Specify the software version: 0x64=100, means firmware version 1.0.0

BAT (Battery Info):

Ex1: 0x0cbc = 3260mV

Signal Strength:

NB-IoT Network signal Strength.

Ex1: 0x15 = 21

0 -113dBm or less

1 -111dBm

2...30 -109dBm... -53dBm

31 -51dBm or greater

99 Not known or not detectable

Interrupt:

This data field shows if this packet is generated by interrupt or not.

Example:

If $\text{byte}[0] \& 0x01 = 0x00$: Normal uplink packet.

If $\text{byte}[0] \& 0x01 = 0x01$: Interrupt Uplink Packet.

Interrupt_level:

This byte shows whether the interrupt is triggered by a high or low level.

Ex1: 0x00 Interrupt triggered by falling edge (low level)

Ex2: 0x01 Interrupt triggered by rising edge (high level)

Temp_alarm:

This data field shows whether the packet was generated by a temperature alarm.

Example:

Ex1: 0x00(H) = 0000(B) No temperature alarm.

Ex2: 0x04(H) = 0100(B) Channel 1 does not alarm, channel 2 alarms.

Ex3: 0x01(H) = 0001(B) Channel 1 alarms, channel 2 does not alarm.

Ex4: 0x05(H) = 0101(B) Channel 1 and channel 2 alarms.

channel1_temp

If payload is: 0105H: $(0105 \& 8000 == 0)$, $\text{temp} = 0105H / 10 = 26.1$ degree

If payload is: FF3FH : $(FF3F \& 8000 == 1)$, $\text{temp} = (FF3FH - 65536) / 10 = -19.3$ degrees.

(FF3F & 8000: Judge whether the highest bit is 1, when the highest bit is 1, it is negative)

If the value is -327.6, it means the channel1 PT100 probe is not connected.

If the value is -983, it means the PT100 converter is not connected.

channel2_temp

Notice: When using the LTC1-NB module, the temperature of channel 2 is invalid and is always 0.0 degrees.

If payload is: 0105H: $(0105 \& 8000 == 0)$, $\text{temp} = 0105H / 10 = 26.1$ degree

If payload is: FF3FH : $(FF3F \& 8000 == 1)$, $\text{temp} = (FF3FH - 65536) / 10 = -19.3$ degrees.

(FF3F & 8000: Judge whether the highest bit is 1, when the highest bit is 1, it is negative)

If the value is -327.6, it means the channel2 PT100 probe is not connected.

If the value is -983, it means the PT100 converter is not connected.

TimeStamp:

Unit TimeStamp Example: 676b5f85(H) = 1735090053(D)

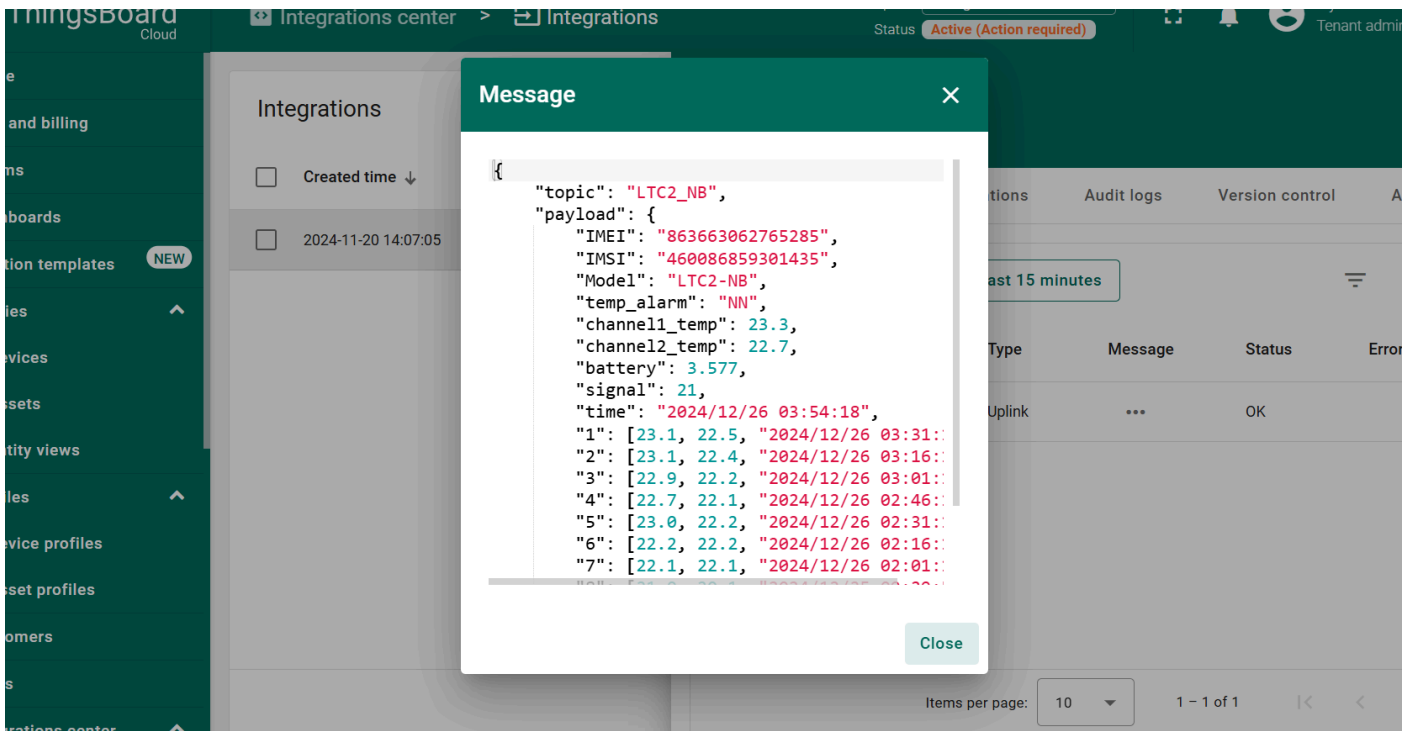
Put the decimal value into this link(<https://www.epochconverter.com/>) to get the time.

2.2.4 ThingsBoard Payload(Type=3)

Notice: When using the LTC1-NB module, the temperature of channel 2 is invalid and is always 0.0 degrees.

Type3 payload special design for ThingsBoard, it will also configure other default server to ThingsBoard.

```
{
  "topic": "LTC2_NB",
  "payload": {
    "IMEI": "863663062765285",
    "IMSI": "460086859301435",
    "Model": "LTC2-NB",
    "temp_alarm": "NN",
    "channel1_temp": 23.3,
    "channel2_temp": 22.7,
    "battery": 3.577,
    "signal": 21,
    "time": "2024/12/26 03:54:18",
    "1": [23.1, 22.5, "2024/12/26 03:31:19"],
    "2": [23.1, 22.4, "2024/12/26 03:16:19"],
    "3": [22.9, 22.2, "2024/12/26 03:01:19"],
    "4": [22.7, 22.1, "2024/12/26 02:46:19"],
    "5": [23.0, 22.2, "2024/12/26 02:31:19"],
    "6": [22.2, 22.2, "2024/12/26 02:16:19"],
    "7": [22.1, 22.1, "2024/12/26 02:01:19"],
    "8": [21.0, 20.1, "2024/12/25 09:29:50"]
  }
}
```

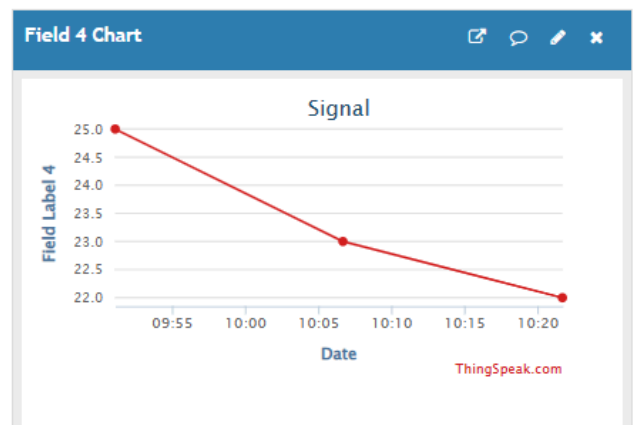
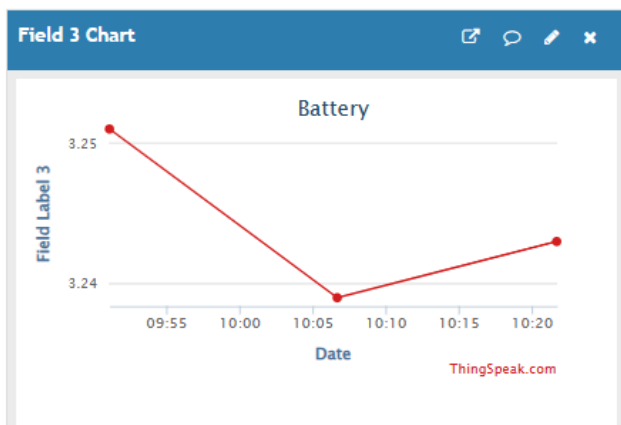
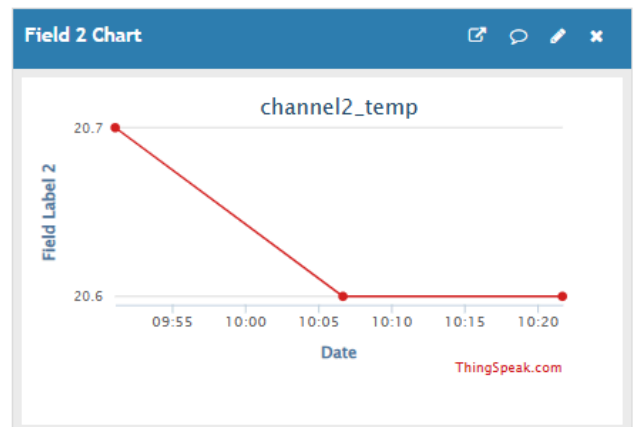
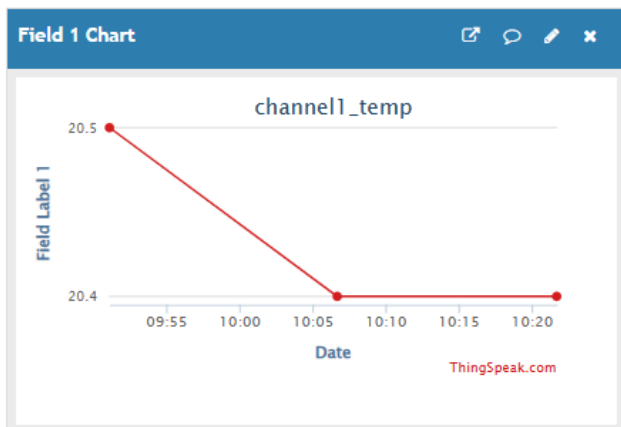


2.2.5 ThingSpeak Payload(Type=1)

Notice: When using the LTC1-NB module, the temperature of channel 2 is invalid and is always 0.0 degrees.

This payload meets ThingSpeak platform requirement. It includes only four fields. Form 1~4 are: Temperature, Humidity, Battery & Signal. This payload type only valid for ThingsSpeak Platform
As below:

`field1=channel1_temp&field2=channel2_temp&field3=BatV&field4=Singal`



3. Configure LTC1/LTC2-NB/NS

3.1 Configure Methods

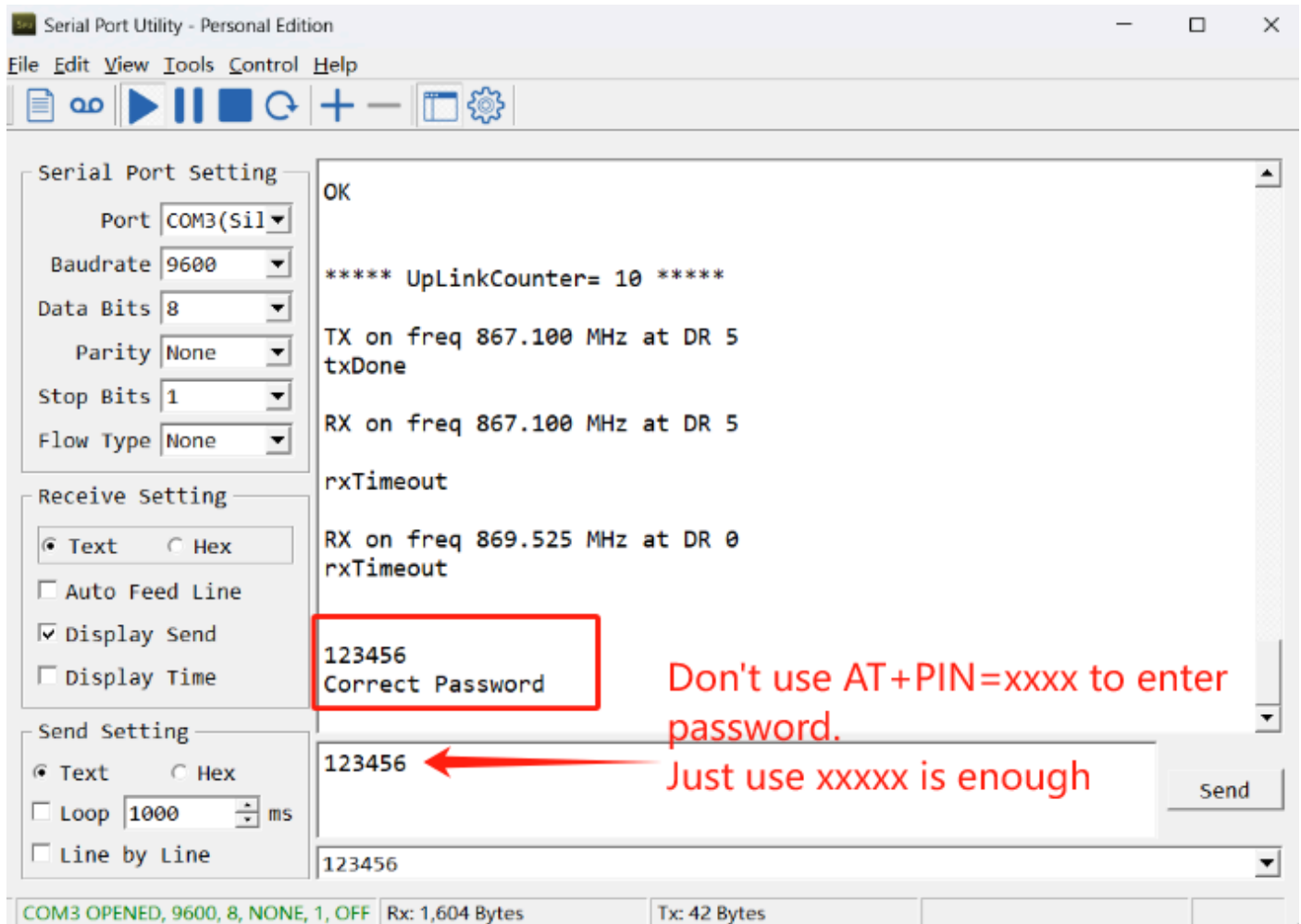
LTC1/LTC2-NB/NS supports below configure method:

- AT Command via Bluetooth Connection (**Recommended**): [BLE Configure Instruction](#).
- AT Command via UART Connection : See [UART Connection](#).

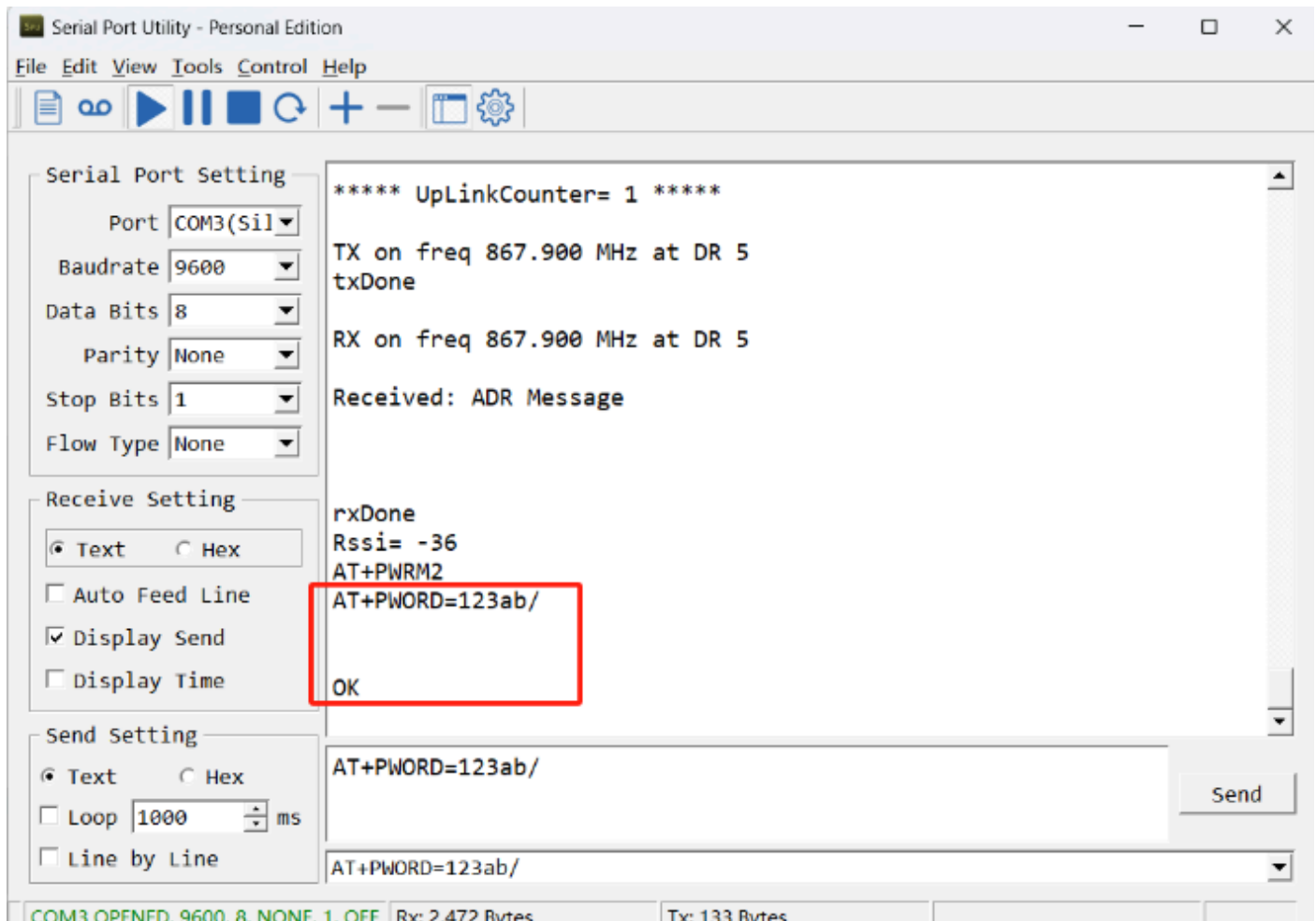
3.2 Serial Access Password

After the Bluetooth or UART connection is successful, use the Serial Access Password to enter the AT command window.

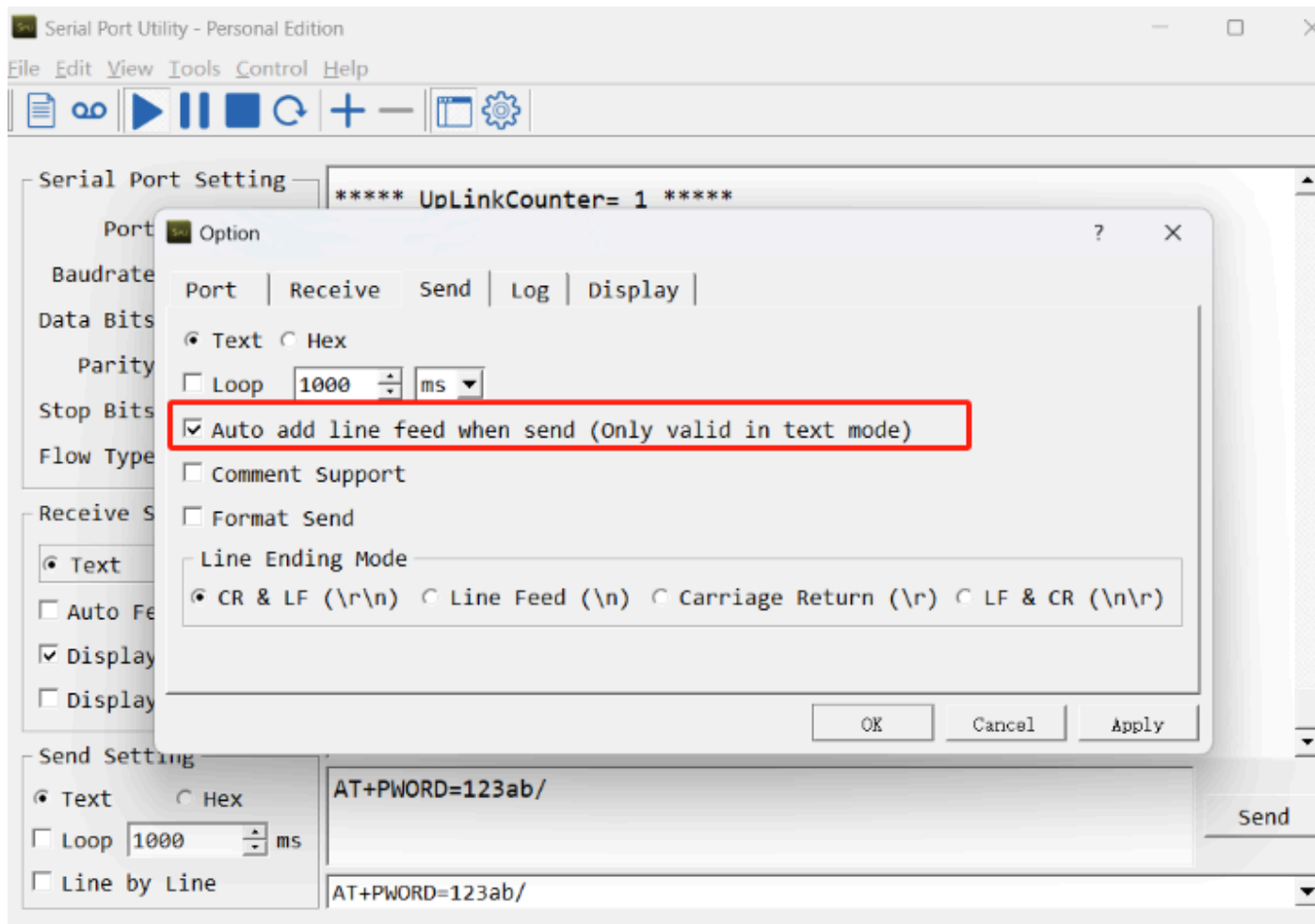
The label on the box of the node will print the initial password: AT+PIN=xxxxxx, and directly use the six-digit password to access the AT instruction window.



If you need to change the password, use **AT+PWORD=xxxxxx** (6 characters), -CB nodes only support lowercase letters.



Note: After entering the command, you need to add a line break, and you can also set automatic line breaks in the Bluetooth tool or UART connection tool.



3.3 AT Commands Set

AT+<CMD>? : Help on <CMD>

AT+<CMD> : Run <CMD>

AT+<CMD>=<value> : Set the value

AT+<CMD>=? : Get the value

General Commands

AT+MODEL : Get module information

ATZ : Trig a reset of the MCU

AT+DEUI : Get or set the Device ID

AT+SLEEP : Get or set the sleep status

AT+DEBUG : Set more info output

AT+SERVADDR: Get or Set the Server address

AT+TDC : Get or set the application data transmission interval in s

AT+APN : Get or set the APN

AT+PRO : Get or Set usage agreement (1:COAP,2:UDP,3:MQTT,4:TCP)

AT+RXDL : Get or Set the receiving time

AT+GETSENSORVALUE : Returns the current sensor measurement

AT+DNSCFG : Get or Set DNS Server

AT+CSQTIME : Get or Set the time to join the network

AT+BKDNS : Get or Set dynamic domain name resolution IP

AT+TLSMOD : Get or Set the TLS mode

AT+INTMOD : Get or Set the trigger interrupt mode (0:input,1:falling or rising,2:falling,3:rising)

AT+WMOD : Get or Set the Alarm Mode

AT+CITEMP : Get or Set the time interval to check sensor value for Alarm

AT+ARTEMP : Get or Set the Alarm Threshold

AT+CLOCKLOG: Enable or Disable Clock Logging

AT+TIMESTAMP : Get or Set UNIX timestamp in second

AT+DOWNTE: Get or set the conversion between the standard version and 1T version downlinks

MQTT Management

AT+MQOS : Set the QoS level of MQTT

AT+CLIENT : Get or Set the MQTT clientID

AT+UNAME : Get or Set the MQTT Username

AT+PWD : Get or Set the MQTT password

AT+PUBTOPIC: Get or set MQTT publishing topic

AT+SUBTOPIC: Get or set MQTT subscription topic

COAP Management

AT+URI1: Get or set CoAP option 1

AT+URI2: Get or set CoAP option 2

AT+URI3: Get or set CoAP option 3

AT+URI4: Get or set CoAP option 4

Information

AT+CFG : Print all settings

AT+FDR1 : Reset parameters to factory default values except for passwords

AT+FDR : Reset Parameters to Factory Default

AT+PASSWORD : Get or set the System password

AT+CDP : Read or Clear cached data

AT+LDATA : Get the last upload data

AT+PDTA: Print the sector data from start page to stop page

AT+PLDTA: Print the last few sets of data

AT+CLRDTA: Clear the storage, record position back to 1st

AT+GETLOG : Print serial port logs

3.4 Test Uplink and Change Update Interval

By default, Sensor will send uplinks **every 2 hours**.

User can use below commands to change the uplink interval.

AT Command: AT+TDC

Example: AT+TDC=7200 // Set Update Interval to 7200 seconds

Downlink Commands: 0x01

Format: Command Code (0x01) followed by 3 bytes.

Example: 12 hours= 43200 seconds 43200(D)=0xA8C0(H)

Downlink Payload: **01 00 A8 C0** // AT+TDC=43200, Set Update Interval to 12 hours.

Note: User can also push the button for more than 1 second to activate an uplink.

3.5 Set the receiving time

Feature: Extend the receiving time

AT Command: AT+RXDL

Example: AT+RXDL=1000 // Set the receiving time delay to 1000ms

Downlink Commands: 0x03

Format: Command Code (0x03) followed by 3 bytes.

Example: Downlink Payload: **03 00 03 E8** // AT+RXDL=1000

3.6 Reset

Feature: Trig a reset of the MCU.

AT Command: ATZ

Downlink Commands: 0x04FF

3.7 Trigger an uplink by external interrupt

LTC2-NB/NS has an external trigger interrupt function. Users can use the PB15 pin to trigger the upload of data packets.

AT command:

- **AT+INTMOD** // Set the trigger interrupt mode
- **AT+INTMOD=0** // Disable Interrupt
- **AT+INTMOD=1** // Trigger by rising and falling edge
- **AT+INTMOD=2** // Trigger by falling edge
- **AT+INTMOD=3** // Trigger by rising edge

Downlink Commands: 0x06

Format: Command Code (0x06) followed by 3 bytes.

Example1: Downlink Payload: **06 00 00 01** // AT+INTMOD=1

Example2: Downlink Payload: **06 00 00 03** // AT+INTMOD=3

3.8 Set the QoS level

This command is used to set the QoS level of MQTT.

AT command:

- **AT+MQOS=xx** // 0~2

Downlink command: 0x07

Format: Command Code (0x07) followed by 1 byte.

Ex1: Downlink payload: **0x0700** // AT+MQOS=0

Ex2: Downlink payload: **0x0701** // AT+MQOS=1

3.9 Clock logging

Sometimes when we deploy lots of end nodes in field. We want all sensors sample data at the same time, and upload these data together for analyze. In such case, we can use clock logging feature.

We can use this command to set the start time of data recording and the time interval to meet the requirements of the specific collection time of data.

AT command: AT+CLOCKLOG=a,b,c,d

a: 0: Disable Clock logging. **1:** Enable Clock Logging

b: Specify First sampling start second: range (0 ~ 3599, 65535) // **Note:** If parameter b is set to 65535, the log period starts after the node accesses the network and sends packets.

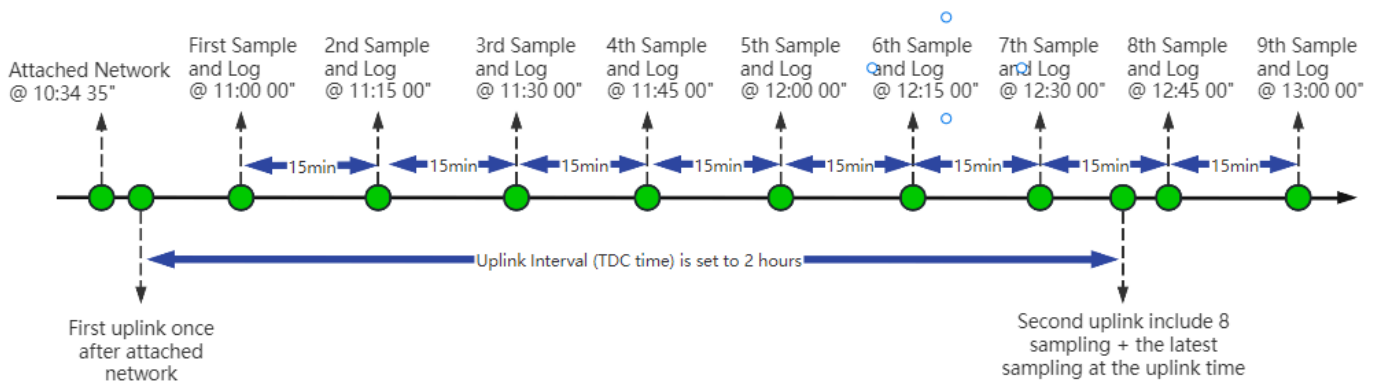
c: Specify the sampling interval: range (0 ~ 255 minutes)

d: How many entries should be uplink on every TDC (max 32)

Note: To disable clock recording, set the following parameters: AT+CLOCKLOG=1,65535,0,0

Example: AT+CLOCKLOG=1,0,15,8

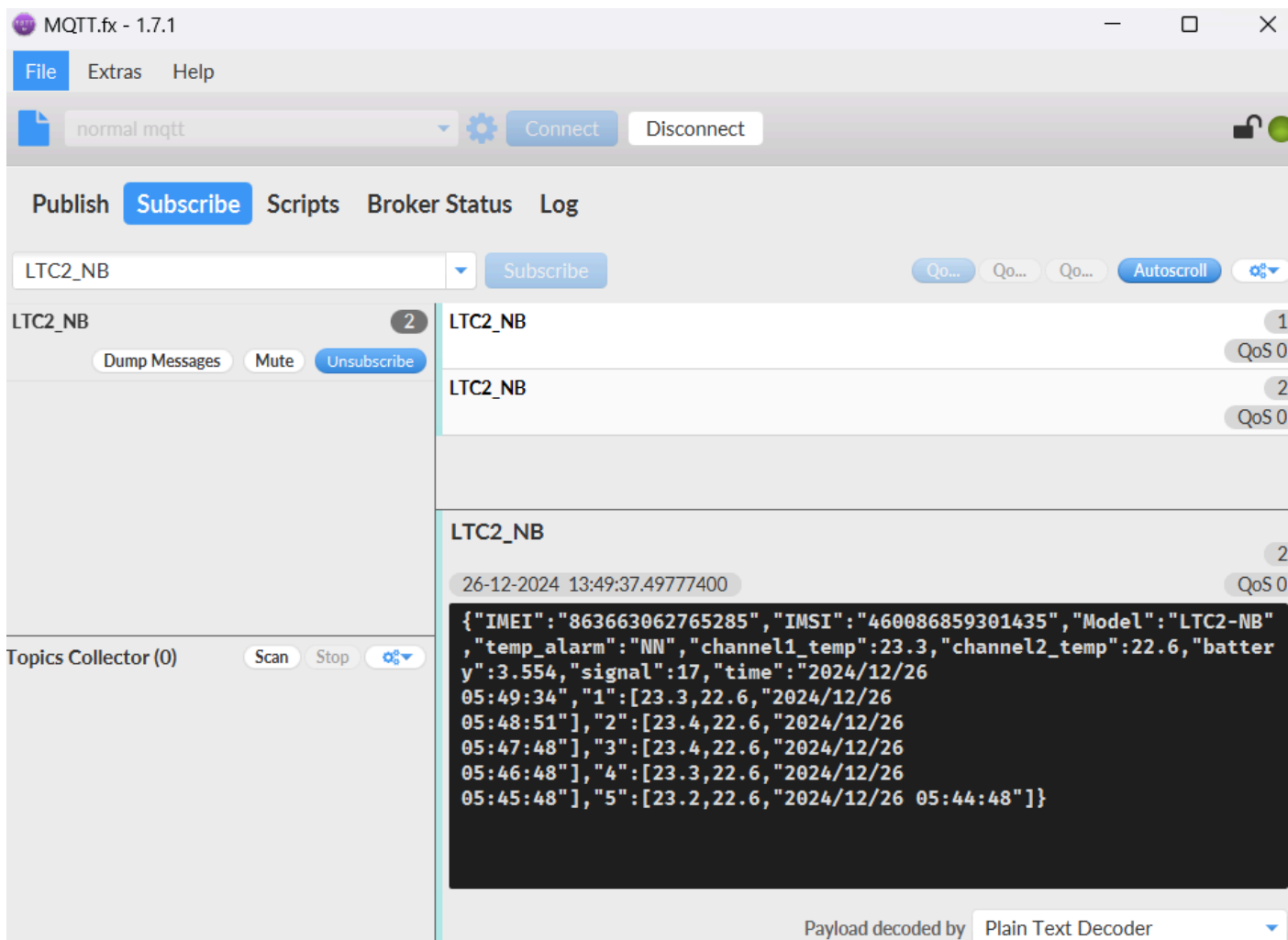
Device will log data to memory start from the 0" second (11:00 00" of first hour and then sampling and log every 15 minutes. Every TDC uplink, the uplink payload will consist: Battery info + last 8 memory record with timestamp + the latest sample at uplink time) . See below for the example.



Example:

AT+CLOCKLOG=1,65535,1,5

After the node sends the first packet, data is recorded to the memory at intervals of 1 minute. For each TDC uplink, the uplink load will include: battery information + the last 5 memory records (payload + timestamp).



Note: Users need to synchronize the server time before configuring this command. If the server time is not synchronized before this command is configured, the command takes effect only after the node is reset.

Downlink command: 0x03

Format: Command Code (0x03) followed by 5 bytes.

- Example 1:** Downlink Payload: **03 01 FFFF 0F 08** // Set SHT record time:
AT+CLOCKLOG=1,65535,15,8
- Example 2:** Downlink Payload: **03 01 04B0 0F 08** // Set SHT record time:
AT+CLOCKLOG=1,1200,15,8

Note: When entering the downlink payload, there must be no Spaces between bytes.

Poll sensor value

User can poll sensor value based on timestamps from the server. Below is the downlink command.

1 byte	4 bytes	4 bytes
31	Timestamp start	Timestamp end

Timestamp start and Timestamp end use Unix TimeStamp format as mentioned above. Devices will reply with all data log during this time period.

For example, downlink command **31 68B6 52E8 68B6 8C18**

Is to check 2025/9/2 02:14:00 to 2025/9/2 06:18:00's data

Datalog Uplink payload (Since firmware v1.0.2)

The Datalog poll reply uplink will use below payload format.

Retrieval data payload:

Size(bytes)	2	2	4
value	channel1_temp	channel2_temp	Timestamp

Function Description: This feature is only used when the clock logging feature is turned on. one uplink packet can send **64** groups of stored data totaling **512** bytes.

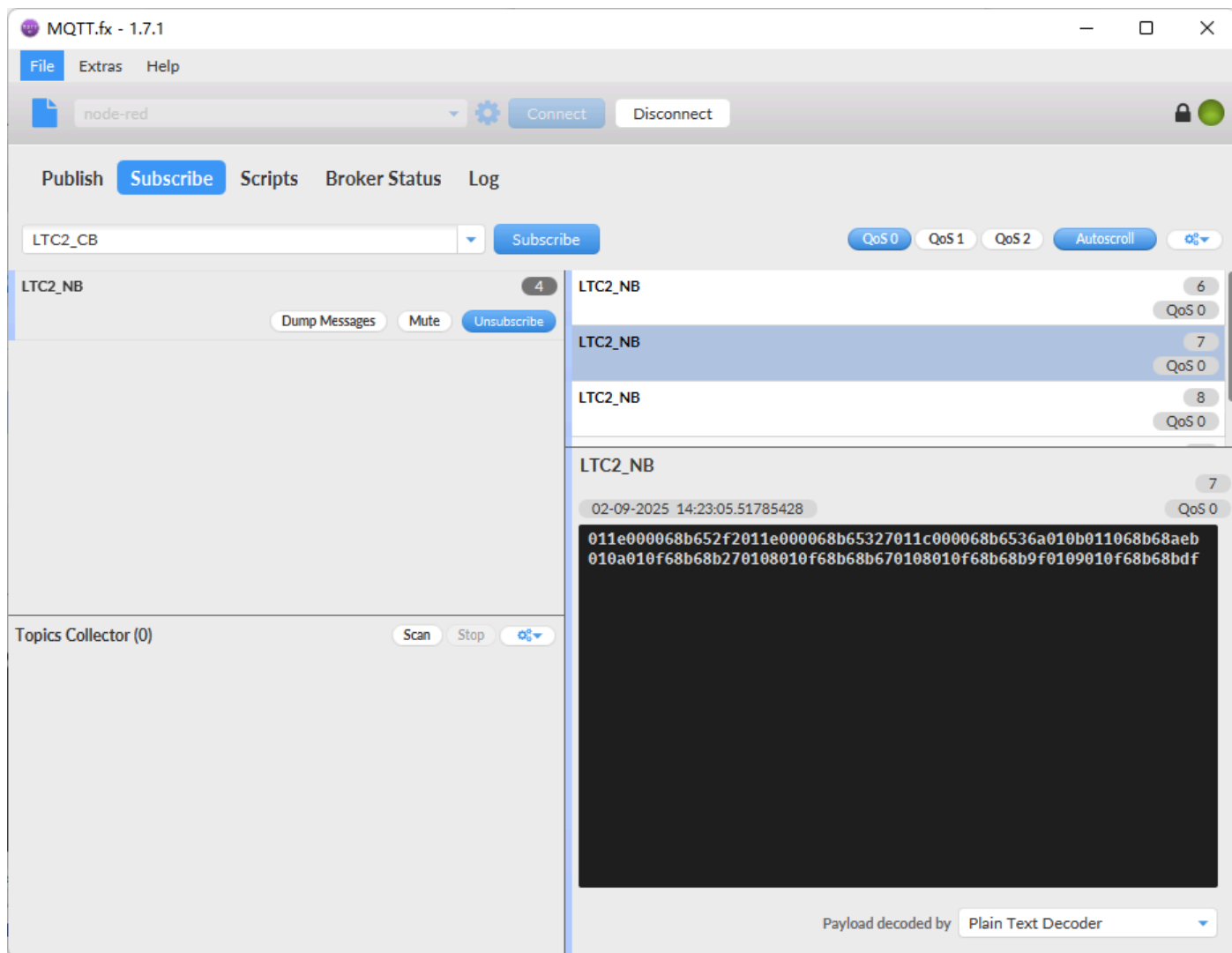
Example(For MQTT.fx):

If user sends below downlink command:

Where : Start time: 68B652E8= time 25/9/2 02:14:00

Stop time: 68B68C18 = time 25/9/2 06:18:00

LTC1/LTC2-NB will uplink this payload.



0x011e000068b652f2 011e000068b65327 011c000068b6536a 010b011068b68aeb 010a010f68b68b27
0108010f68b68b67 0108010f68b68b9f 0109010f68b68bdf

LTC2-NB(8 groups for example)					
011e	0000	68b652f2	011e000068b65327	011c000068b6536a	010b011068b68aeb
channel1_temp	channel2_temp	Timestamp	2nd data	3rd data	4th data
8 Bytes			8 Bytes	8 Bytes	8 Bytes
010a010f68b68b27		0108010f68b68b67	0108010f68b68b9f	0109010f68b68bdf	
5th data		6th data	7th data	8th data	
8 Bytes		8 Bytes	8 Bytes	8 Bytes	

channel1_temp = $0x011e/1000 = 28.6^{\circ}\text{C}$

channel2_temp = $0x0000/1000 = 0.0^{\circ}\text{C}$

Unix time is $0x68b652f2 = 1756779250s = 25/9/2 \ 02:14:00$

3.10 Set the TLS mode

Refer to this link ([MQTT Connection to send data to Tago.io](#)) to use the TLS mode.

AT Command: AT+TSLMOD

Example 1: AT+TSLMOD=0,0 // Disable TLS Mode.

Example 2: AT+TSLMOD=1,0 // No authentication

AT+TSLMOD=1,1 // Perform server authentication

AT+TSLMOD=1,2 // Perform server and client authentication if requested by the remote server

Downlink command: 0x09

Format: Command Code (0x09) followed by 2 bytes.

Example1: Downlink Payload: **09 00 00** // AT+TSLMOD=0,0

Example2: Downlink Payload: **09 01 02** // AT+TSLMOD=1,2

3.11 Set the search network time

Feature: Get or Set the time to join the network(unit: minutes).

AT Command: AT+CSQTIME

Example: AT+CSQTIME=10 // Set the search time to 10 minutes.

Downlink command: 0x13

Format: Command Code (0x13) followed by 1 byte.

Example: Downlink Payload: **13 0A** // AT+CSQTIME=10

3.12 Set alarm mode

Feature: Enable/Disable Alarm Mode.

AT Command: AT+WMOD

- AT+WMOD=0 // Disable the alarm mode, default.
- AT+WMOD=1 // Enable the alarm mode.

Downlink Command: 0xA5

- Downlink payload: **0xA500** // Same as AT+WMOD=0
- Downlink payload: **0xA501** // Same as AT+WMOD=1

3.13 Alarm check time

Feature: The time interval to check sensor value for Alarm. (Default: AT+CITEMP=5; Unit: minute)

AT Command: AT+CITEMP

Example:

- AT+CITEMP=10 // Set collection interval in 10 min, only in alarm mode.

Downlink Command: 0xA6

Format: Command Code (0xA6) followed by 2 bytes.

Example:

- Downlink payload: **0xA6000A** // Same as AT+CITEMP=10

3.14 Set Alarm Threshold

Feature: Set Alarm Threshold. (Unit: °C)

AT Command: AT+ARTEMP

The first parameter sets the low limit of channel 1, and the second parameter sets the high limit of channel 1.

The third parameter sets the low limit for channel 2, and the fourth parameter sets the high limit for channel 2.

Example:

- AT+ARTEMP=-200,800,-200,800 // Channel 1 & Channel 2 operating temp: -200°C~800°C, alarm when out of range.
- AT+ARTEMP=10,100,10,101 // Channel 1 operating temp: 10°C~100°C; Channel 2 operating temp: 10°C~101°C, alarm when out of range.

Downlink Command: 0xA7

Format: Command Code (0xA7) followed by 8 bytes.

Each two bytes after the function code 0xA7 is a parameter, corresponding to the four parameters of AT+ARTEMP in sequence.

Example:

- Downlink payload: **0xA7FF380320FF380320** // Same as AT+ARTEMP=-200,800,-200,800
- Downlink payload: **0xA7000A0064000A0065** // Same as AT+ARTEMP=10,100,10,101

Note: For negative temperature -200°C: 65536-200= 65336(D) = 0xFF38(H)

3.15 Factory data reset

Two different restore factory Settings configurations.

AT command:

- **AT+FDR** // Reset Parameters to Factory Default.
- **AT+FDR1** // Reset parameters to factory default values **except for passwords**.

3.16 Set CoAP option

This command sets the connection parameters of the COAP.

AT command:

- AT+URI1 // CoAP option name, CoAP option length, "CoAP option value"
- AT+URI2 // CoAP option name, CoAP option length, "CoAP option value"

- AT+URI3 // CoAP option name, CoAP option length, "CoAP option value"
- AT+URI4 // CoAP option name, CoAP option length, "CoAP option value"

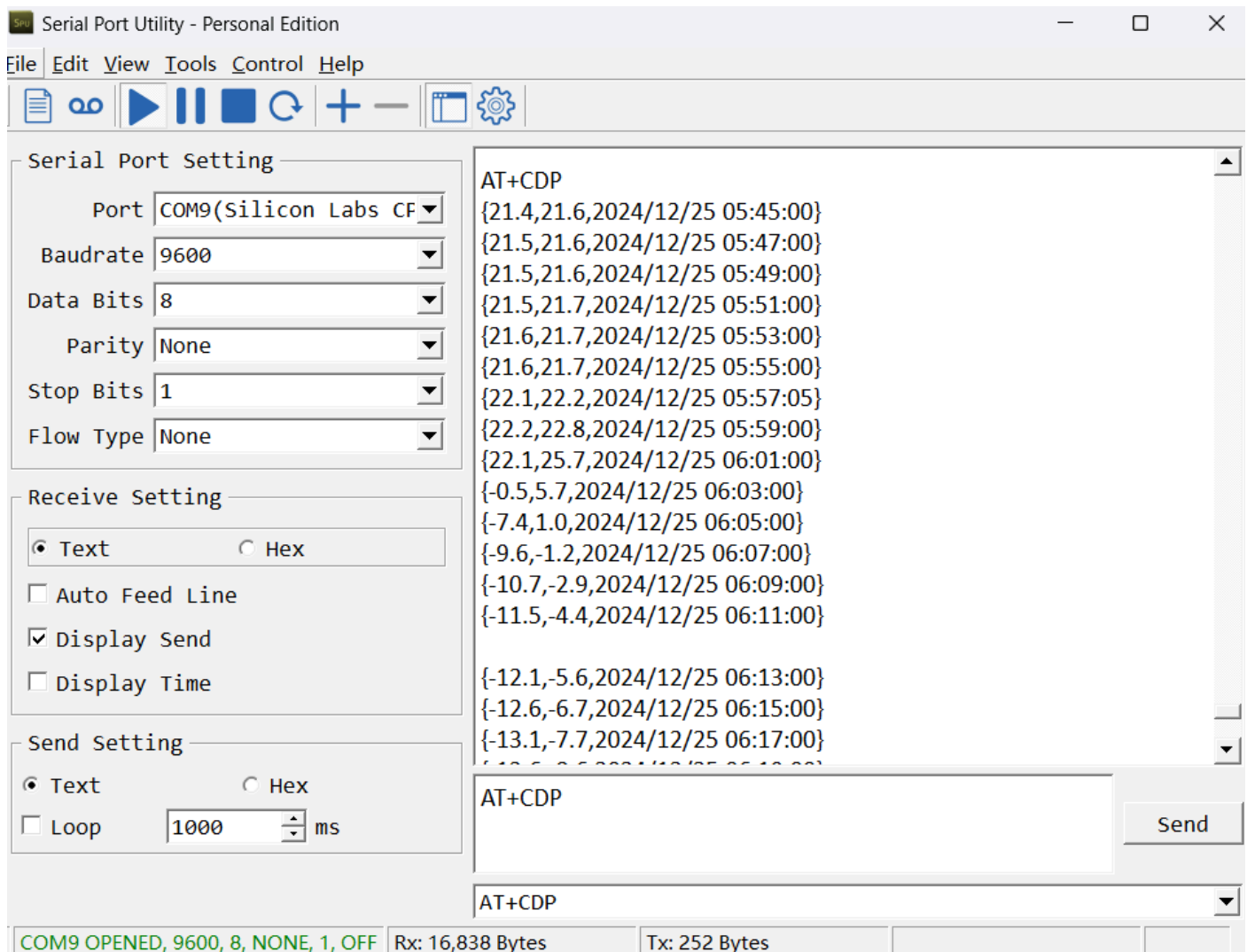
Example:

- AT+URI1=11,38,"i/aaaa241f-af4a-b780-4468-c671bb574858"

3.17 Example Query saved historical records

- **AT command:** AT+CDP

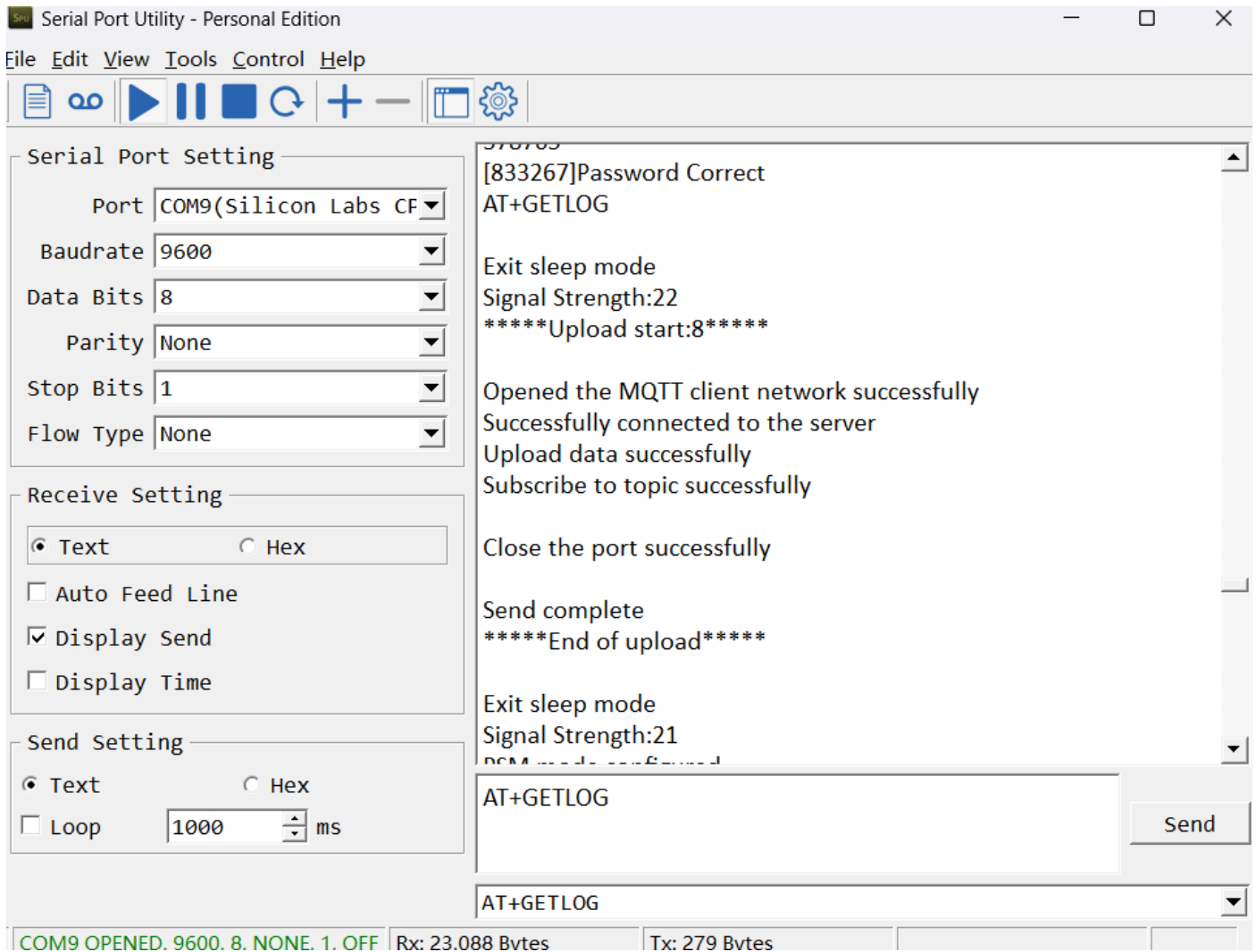
This command can be used to search the saved history, recording up to 32 groups of data, each group of historical data contains a maximum of 100 bytes.



3.18 Uplink log query

- **AT command:** AT+GETLOG

This command can be used to query upstream logs of data packets.



3.19 Print last few data entries

Feature: Print the last few data entries

AT command: AT+PLDTA

Command Example

4 bytes

AT+PLDTA=8 Print last 8 entries	Stop Tx events when read sensor data 1 25/9/2 06:41:39 channel1_temp=26.7 channel2_temp=27.2 2 25/9/2 06:42:35 channel1_temp=26.8 channel2_temp=27.2 3 25/9/2 06:43:39 channel1_temp=26.9 channel2_temp=27.3 4 25/9/2 06:44:35 channel1_temp=26.8 channel2_temp=27.4 5 25/9/2 06:45:39 channel1_temp=26.7 channel2_temp=27.4 6 25/9/2 06:46:35 channel1_temp=26.7 channel2_temp=27.3 7 25/9/2 06:47:39 channel1_temp=26.7 channel2_temp=27.4 8 25/9/2 06:48:35 channel1_temp=26.7 channel2_temp=27.3 OK
------------------------------------	--

Downlink Command:

No downlink commands for feature

3.20 Print data entries base on page

Feature: Print the sector data from start page to stop page.

AT command: AT+PDTA

Command Example	4 bytes
-----------------	---------

AT+PDTA=1,1 Print page 1 to 1	Stop Tx events when read sensor data 8028A00 25/7/17 01:46:47 channel1_temp=0.0 channel2_temp=0.0 8028A08 25/7/17 02:20:55 channel1_temp=0.0 channel2_temp=0.0 8028A10 25/7/17 02:35:55 channel1_temp=0.0 channel2_temp=0.0 8028A18 25/7/17 02:50:55 channel1_temp=580.0 channel2_temp=0.0 8028A20 25/7/17 03:05:55 channel1_temp=581.2 channel2_temp=0.0 8028A28 25/7/17 03:21:00 channel1_temp=0.0 channel2_temp=249.8 8028A30 25/7/17 03:35:55 channel1_temp=0.0 channel2_temp=349.6 8028A38 25/7/17 03:50:57 channel1_temp=0.0 channel2_temp=748.9 8028A40 25/7/17 04:05:55 channel1_temp=0.0 channel2_temp=0.0 8028A48 25/7/17 04:20:55 channel1_temp=0.0 channel2_temp=0.0 8028A50 25/7/17 04:35:55 channel1_temp=0.0 channel2_temp=0.0 8028A58 25/7/17 04:50:55 channel1_temp=0.0 channel2_temp=0.0 8028A60 25/7/17 05:05:55 channel1_temp=0.0 channel2_temp=0.0 8028A68 25/7/17 05:20:55 channel1_temp=0.0 channel2_temp=0.0 8028A70 25/7/17 05:35:55 channel1_temp=0.0 channel2_temp=0.0 8028A78 25/7/17 06:04:10 channel1_temp=0.0 channel2_temp=0.0 OK
----------------------------------	---

Downlink Command:

No downlink commands for feature

3.21 Clear Flash Record

Feature: Clear flash storage for data log feature.

AT command: AT+CLRDTA

Command Example	Function	Response
AT+CLRDTA	Clear date record	Please wait for the erase to complete Clear all stored sensor data... OK

Downlink Command: 0x32

- Example: 0x32 00 // Same as AT+CLRDTA

3.22 Set the downlink debugging mode

Feature: Set the conversion between the standard version and 1T version downlinks.

AT command: AT+DOWNTE

Command Example	Function/Parameters	Response/Explanation
AT+DOWNTE=?	Get current Settings	0,0 (default) OK
AT+DOWNTE=a,b	a: Set the conversion between the downlink of the standard version and 1T version	0: Set the downlink of the standard version. 1: Set the downlink of the 1T version(ThingsEye platform)
	b: Enable/Disable downlink debugging	0: Disable downlink debugging mode. 1: Enable downlink debugging mode, users can see the original downlink reception.

Example:

- AT+DOWNTE=0,1 // Set to standard version downlink, and enable downlink debugging.
- AT+DOWNTE=1,1 // Set to 1T version downlink, and enable downlink debugging.

Downlink Command:

No downlink commands for feature

3.23 Domain name resolution settings

Feature: Set dynamic domain name resolution IP.

AT command: AT+BKDNS

Command Example	Function/Parameters	Response/Explanation
AT+BKDNS=?	Get current Settings	0,0,NULL (default) OK

AT+BKDNS=a,b,c	a: Enable/Disable dynamic domain name resolution.	1: Disable dynamic domain name update. The ip address will be saved after the domain name is resolved, if the next domain name resolution fails, the last saved ip address will be used. 2: Enable dynamic domain name update. The ip address will be saved after domain name resolution, if the next domain name resolution fails, the last saved ip address will be used, and the domain name resolution will be updated regularly according to the time set by the customer.
	b: Set the time to update the domain name resolution at regular intervals.	Unit: hour
	c: Set the IP address manually.	The format is the same as AT+SERVADDR. If domain name resolution fails, this ip address will be used directly, if domain name resolution succeeds, parameter c will be updated to the successfully resolved IP address.

Example:

- AT+BKDNS=1,0 // Dynamic domain name resolution is disabled.
- AT+BKDNS=2,1 // The dynamic domain name resolution function is enabled and the automatic update time is set to 1 hour.
- AT+BKDNS=2,4,3.69.98.183,1883 // The dynamic domain name resolution function is enabled and the automatic update time is set to 4 hour, and manually set the ip address, if the domain name failed to resolve, it will directly use this ip to communicate. When the next domain name resolution is successful, it will be updated to the ip address of the successful resolution.

Downlink Command:

No downlink commands for feature

4. Battery & Power Consumption

LTC1/LTC2-NB use ER26500 + SPC1520 battery pack and LTC1/LTC2-NS use 3000mAh Recharable Battery with Solar Panel. See below link for detail information about the battery info and how to replace.

[Battery Info & Power Consumption Analyze](#) .

5. Firmware update

User can change device firmware to::

- Update with new features.
- Fix bugs.

Firmware and changelog can be downloaded from : [Firmware download link](#) 

Methods to Update Firmware:

- (Recommended way) OTA firmware update via BLE: [Instruction](#) .
- Update through UART TTL interface : [Instruction](#) .

6. FAQ

6.1 How can I access t BC660K-GL AT Commands?

User can access to BC660K-GL directly and send AT Commands.

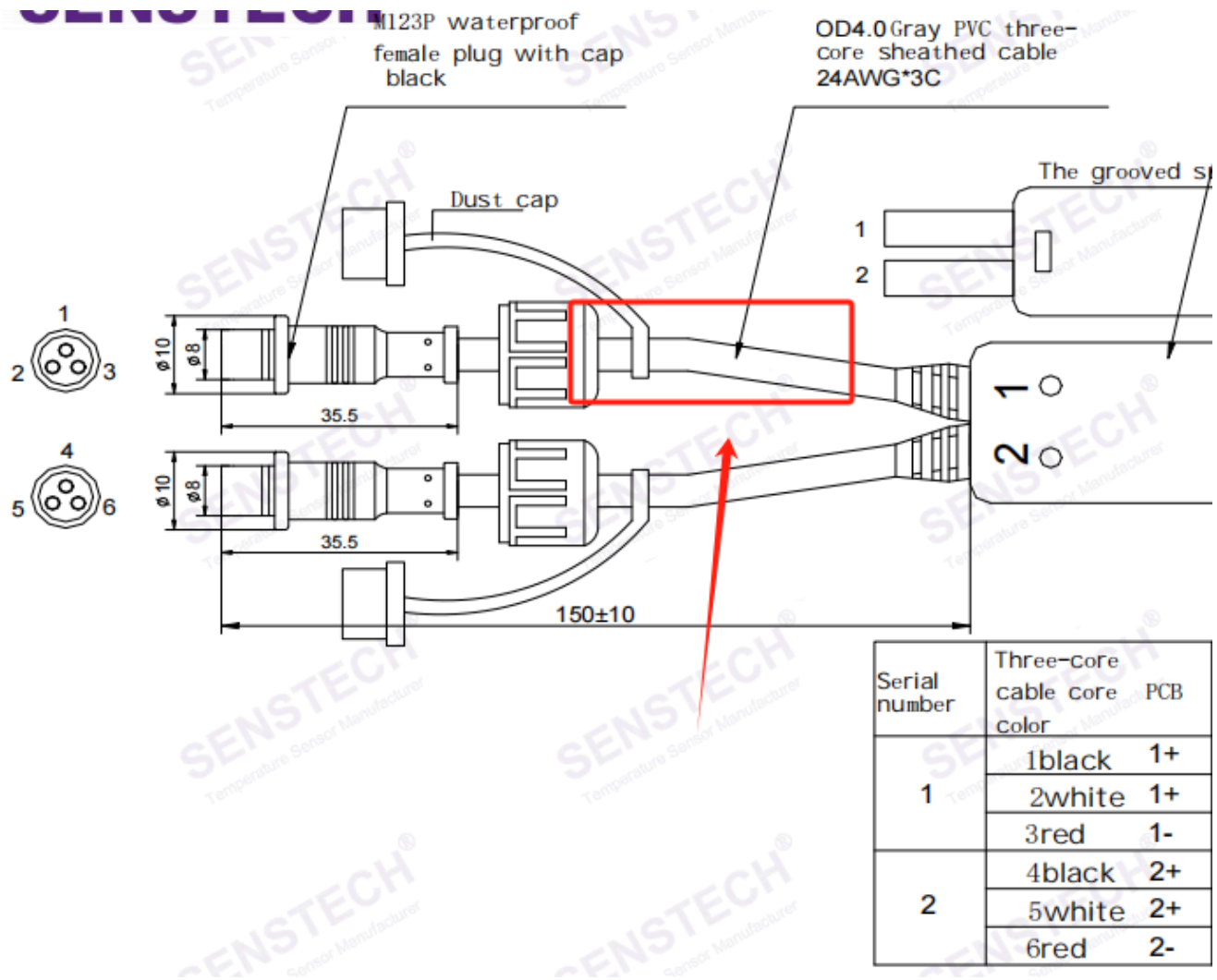
[See BC660K-GL AT Command set](#) 

6.2 How do users connect custom PT100 sensors?

1. The user needs to cut off one end of the original probe and peel off the wire core.



2. Connect according to the corresponding positive and negative poles.

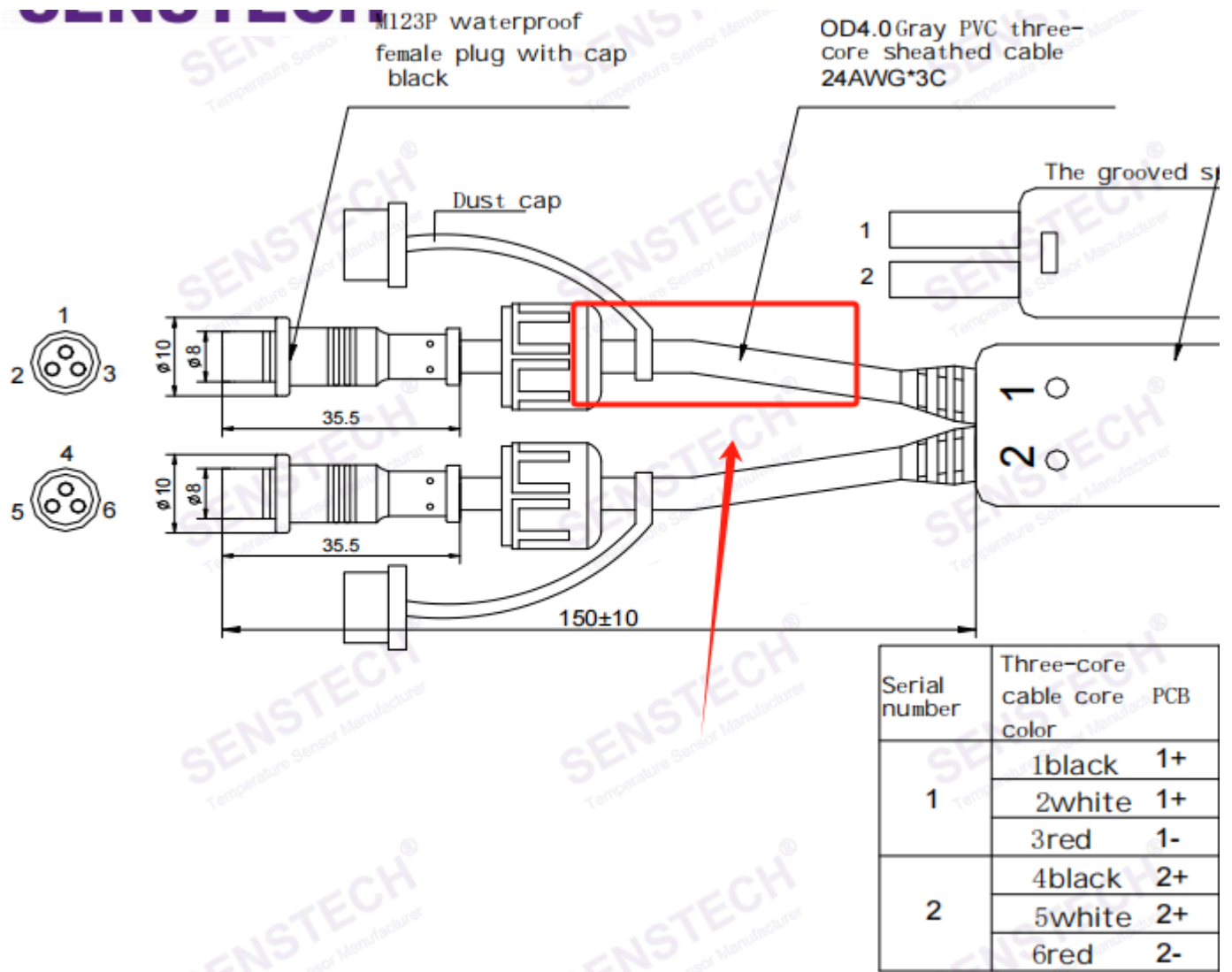


6.3 How do customers connect a two-wire PT100?

1. The user needs to cut off one end of the original probe and peel off the wire core.



2. Connect according to the corresponding positive and negative poles.



3. Wiring instructions:

positive terminal <-----> black wire.
negative terminal <-----> red wire.
Short-circuit the red and white wires.

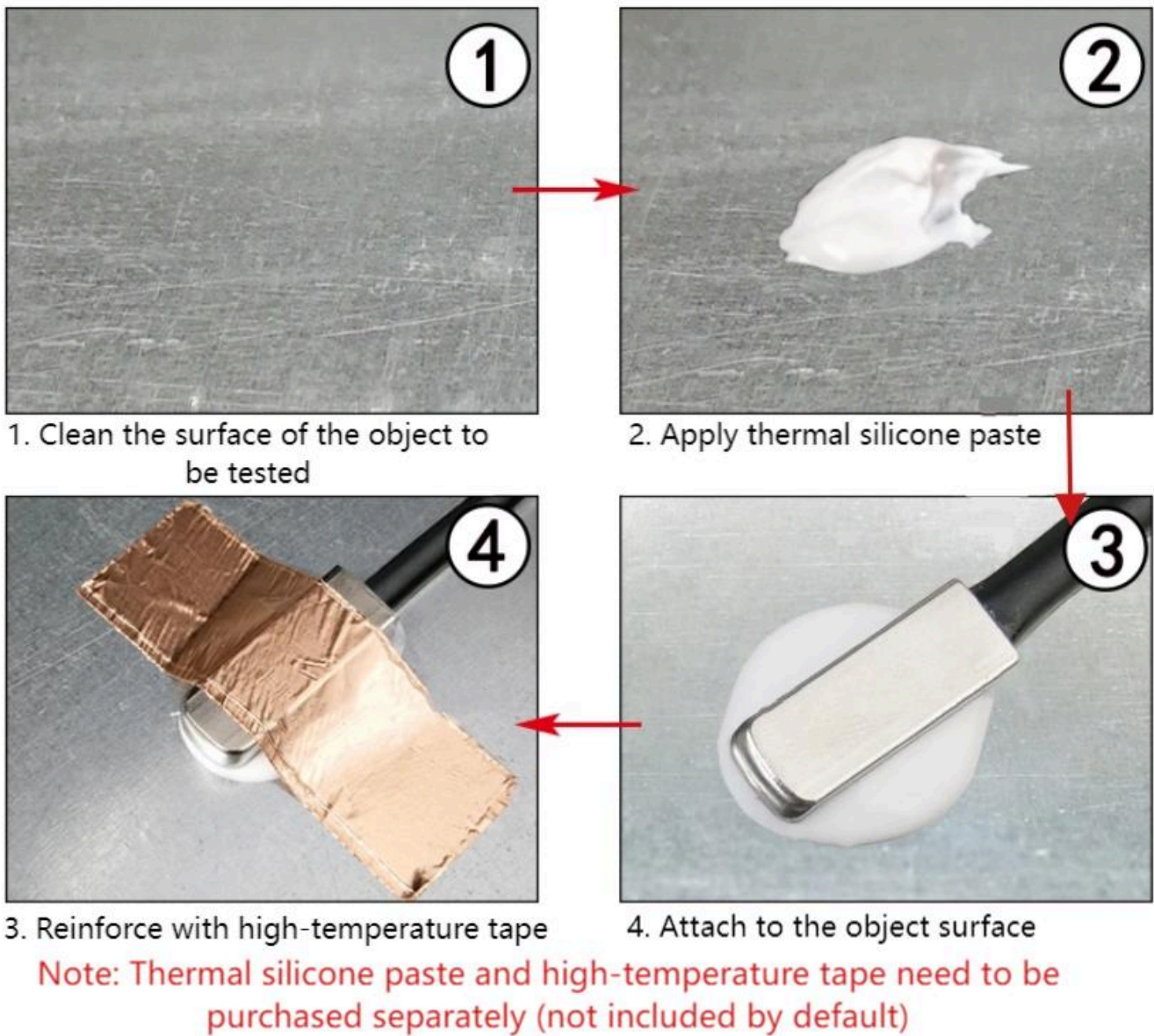
6.4 Why is there no LED response when I press the button on the solar panel model?

If the LED does not light up when you press the button, it may be because the battery has entered protection mode.

Solution: To reactivate the battery, simply expose the solar panel to direct sunlight.

For more details, please refer to: [Battery Protection State \(Apply to Solar Panel + Li-ion battery\)](#)

6.5 How to fix the FT -- Flat Type probe to surface?



7. Order Info

Notice: This part number doesn't include the Temperature sensor. Users need to purchase separately.

Part Number: **LTC1-NB/NS-XX** or **LTC2-NB/NS-XX**

XX:

- **GE:** General version (Exclude SIM card)
- **1T:** with 1NCE * 10 years 500MB SIM card and Pre-configure to ThingsEye server

Probe Options: **DR-SI**, **DR-LT**, **DR-HT**, **DR-FSA**, **DR-FT**

8. Packing Info

Package Includes:

- LTC1-NB/NS or LTC2-NB/NS NB-IoT Temperature Transmitter x 1

Dimension and weight:

- Device Size: cm
- Device Weight: g
- Package Size / pcs : cm
- Weight / pcs : g

9. Support

- Support is provided Monday to Friday, from 09:00 to 18:00 GMT+8. Due to different timezones we cannot offer live support. However, your questions will be answered as soon as possible in the before-mentioned schedule.
- Provide as much information as possible regarding your enquiry (product models, accurately describe your problem and steps to replicate it etc) and send a mail to support@dragino.com .