RS485-LB/LS -- Waterproof RS485/UART to LoRaWAN Converter

last modified by Bei Jinggeng

on 2024/09/03 16:17

Table of Contents

Table of Contents:

1. Introduction

1.1 What is RS485-LB/LS RS485/UART to LoRaWAN Converter

The Dragino RS485-LB/LS is a **RS485 / UART to LoRaWAN Converter** for Internet of Things solutions. User can connect RS485 or UART sensor to RS485-LB/LS converter, and configure RS485-LB/LS to periodically read sensor data and upload via LoRaWAN network to IoT server.

RS485-LB/LS can interface to RS485 sensor, 3.3v/5v UART sensor or interrupt sensor. RS485-LB/LS provides **a 3.3v output** and **a 5v output** to power external sensors. Both output voltages are controllable to minimize the total system power consumption.

RS485-LB/LS is IP67 **waterproof** and powered by **8500mAh Li-SOCl2 battery** or **solar powered + li-on battery,** it is designed for long term use for several years.

RS485-LB/LS runs standard **LoRaWAN 1.0.3 in Class A**. It can reach long transfer range and easy to integrate with LoRaWAN compatible gateway and IoT server.

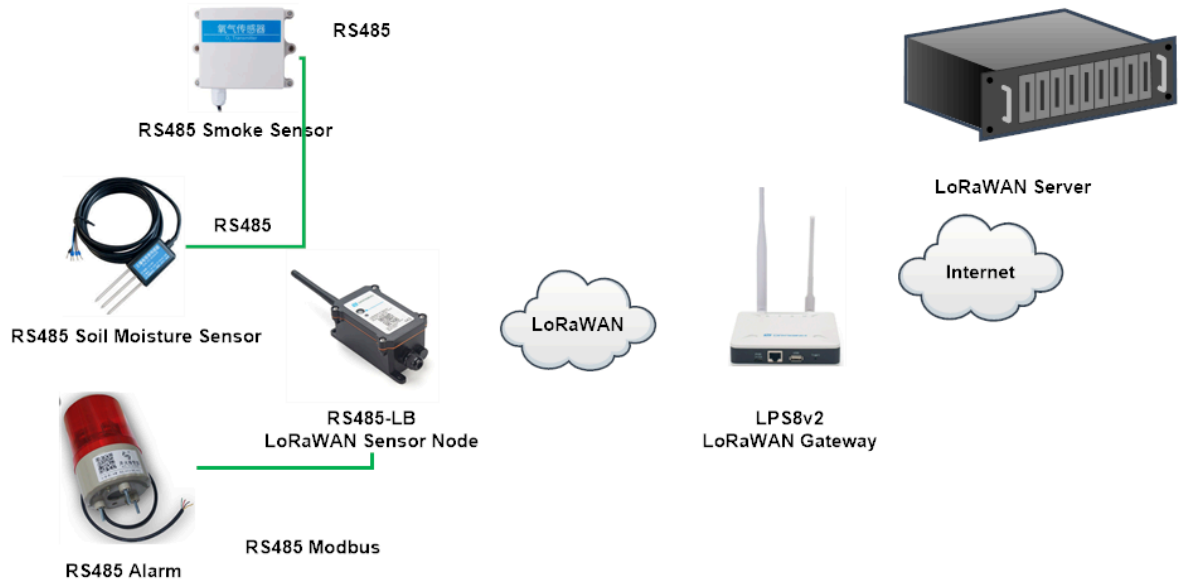
For data uplink, RS485-LB/LS sends user-defined commands to RS485 devices and gets the return from the RS485 devices. RS485-LB/LS will process these returns data according to user-define rules to get the final payload and upload to LoRaWAN server.

For data downlink, RS485-LB/LS runs in LoRaWAN Class A. When there is downlink commands from LoRaWAN server, RS485-LB/LS will forward the commands from LoRaWAN server to RS485 devices.

RS485-LB/LS **Supports BLE configure and wireless OTA update** which make user easy to use.

Each RS485-LB/LS pre-load with a set of unique keys for LoRaWAN registration, register these keys to LoRaWAN server and it will auto connect after power on.

RS485-LB in a LoRaWAN Network



1.2 Specifications

Common DC Characteristics:

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

I/O Interface:

- Battery controllable output (2.6v ~ 3.6v depends on battery)
- +5v controllable output
- 1 x RS485 Interface
- 1 x UART Interface , 3.3v or 5v
- 1 x Interrupt or Digital IN pins
- 1 x I2C Interface
- 1 x one wire interface

LoRa Spec:

- Frequency Range, Band 1 (HF): 862 ~ 1020 Mhz
- Max +22 dBm constant RF output vs.
- RX sensitivity: down to -139 dBm.
- Excellent blocking immunity

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

- Sleep Mode: 5uA @ 3.3v
- LoRa Transmit Mode: 125mA @ 20dBm, 82mA @ 14dBm

1.3 Features

- LoRaWAN 1.0.3 Class A
- Frequency Bands: CN470/EU433/KR920/US915/EU868/AS923/AU915/IN865/RU864/MA869
- Ultra-low power consumption
- Support multiply RS485 devices by flexible rules
- Support Modbus protocol
- Support Interrupt uplink
- Supports connecting a UART sensors with 3.3V or 5V
- Support Bluetooth v5.1 and LoRaWAN remote configure
- Support wireless OTA update firmware
- AT Commands to change parameters
- Uplink on periodically
- Downlink to change configure
- 8500mAh Li/SOCl2 Battery (RS485-LB)
- Solar panel + 3000mAh Li-on battery (RS485-LS)

1.4 Applications

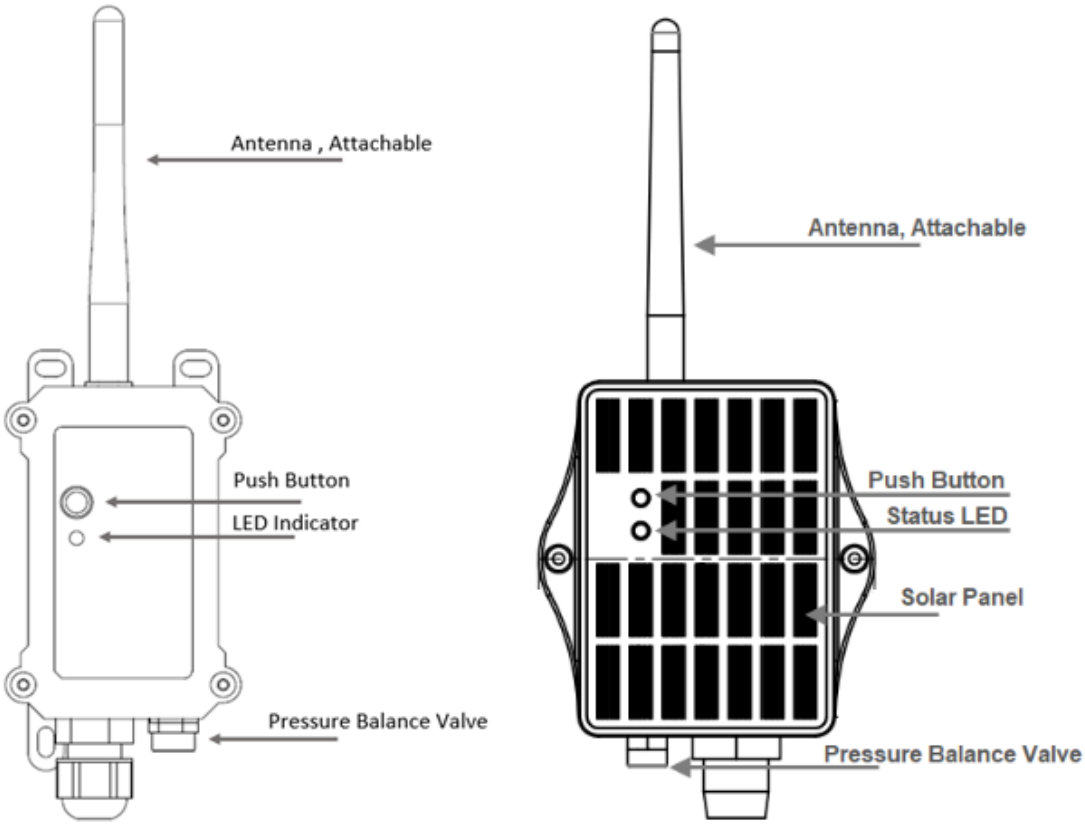
- Smart Buildings & Home Automation
- Logistics and Supply Chain Management
- Smart Metering
- Smart Agriculture
- Smart Cities
- Smart Factory

1.5 Sleep mode and working mode

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

Working Mode: In this mode, Sensor will work as LoRaWAN Sensor to Join LoRaWAN 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



Behavior on ACT	Function	Action
Pressing ACT between 1s < time < 3s	Send an uplink	If sensor is already Joined to LoRaWAN 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.
Pressing ACT for more than 3s	Active Device	Green led will fast blink 5 times, device will enter OTA mode for 3 seconds. And then start to JOIN LoRaWAN network. Green led will solidly turn on for 5 seconds after joined in network. Once sensor is active, BLE module will be active and user can connect via BLE to configure device, no matter if device join or not join LoRaWAN network.
Fast press ACT 5 times.	Deactivate Device	Red led will solid on for 5 seconds. Means device is in Deep Sleep Mode.

1.7 BLE connection

RS485-LB/LS supports BLE remote configure.

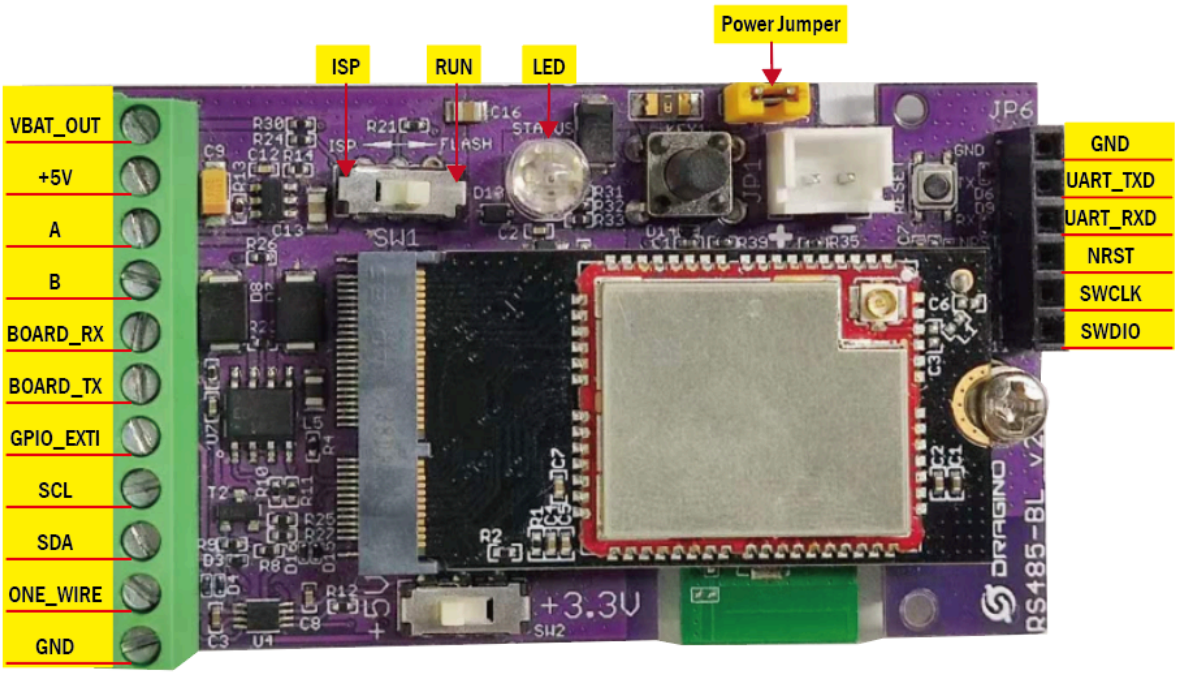
BLE can be used to configure the parameter of sensor or see the console output from sensor. BLE will be only activate on below case:

- 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

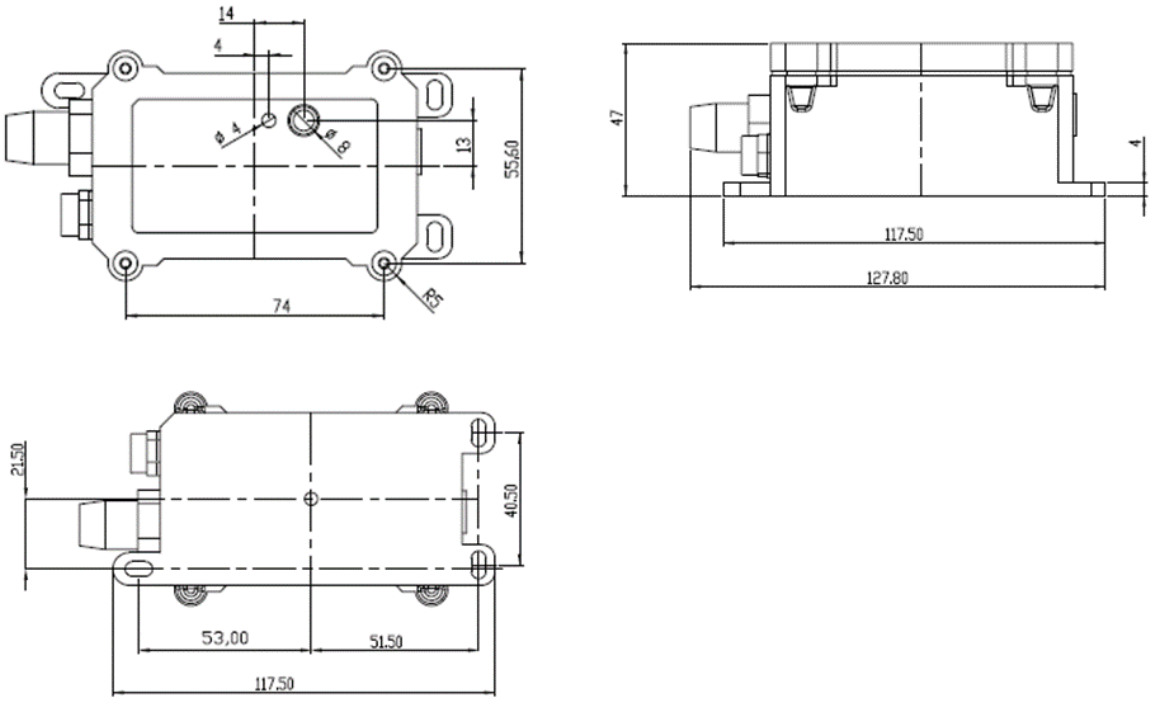


1.8.1 SW2 Jumper (Define UART level to external Sensor)

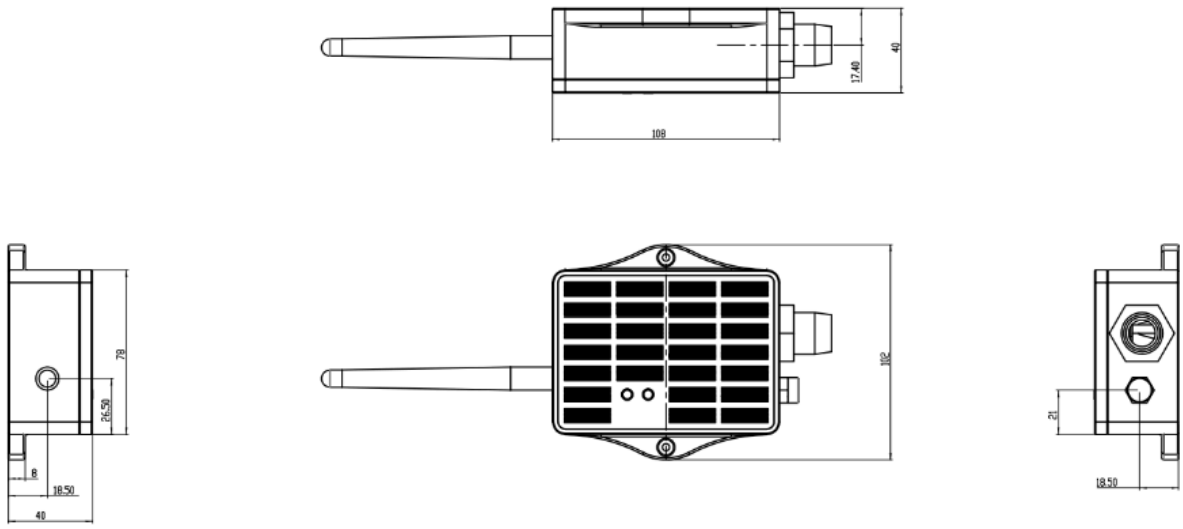
SW2 defines the voltage level of BOARD_RX and BOARD_TX pins. It should match the external sensor voltage level

1.9 Mechanical

1.9.1 for LB version



1.9.2 for LS version



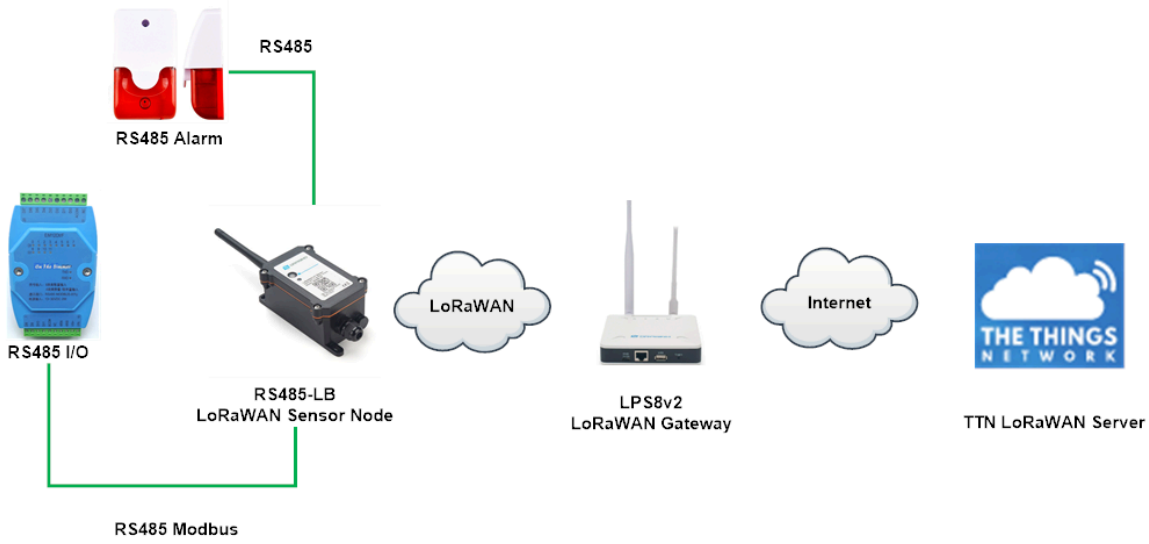
2. Operation Mode

2.1 How it works?

The RS485-LB/LS is configured as LoRaWAN OTAA Class A mode by default. It has OTAA keys to join network. To connect a local LoRaWAN network, user just need to input the OTAA keys in the network server and power on the RS485-LB/LS. It will auto join the network via OTAA.

2.2 Quick guide to connect to LoRaWAN server (OTAA)

Following is an example for how to join the TTN v3 LoRaWAN Network. Below is the network structure; we use the LPS8v2 as a LoRaWAN gateway in this example.

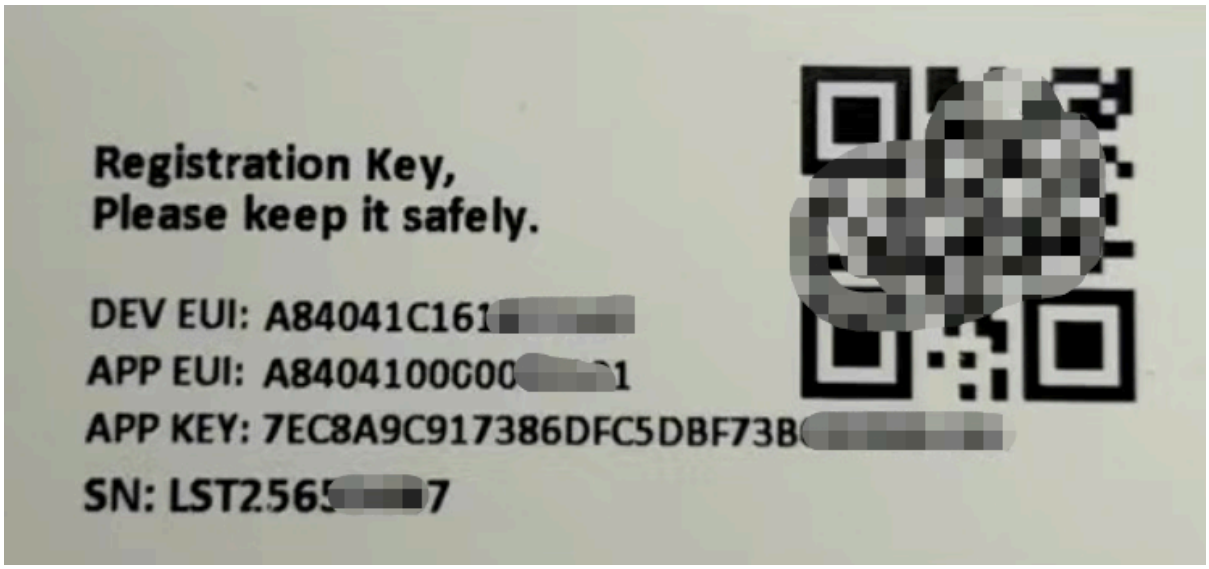


The RS485-LB/LS in this example connected to two RS485 devices for demonstration, user can connect to other RS485 devices via the same method.

The LPS8v2 is already set to connected to TTN network , so what we need to now is configure the TTN server.

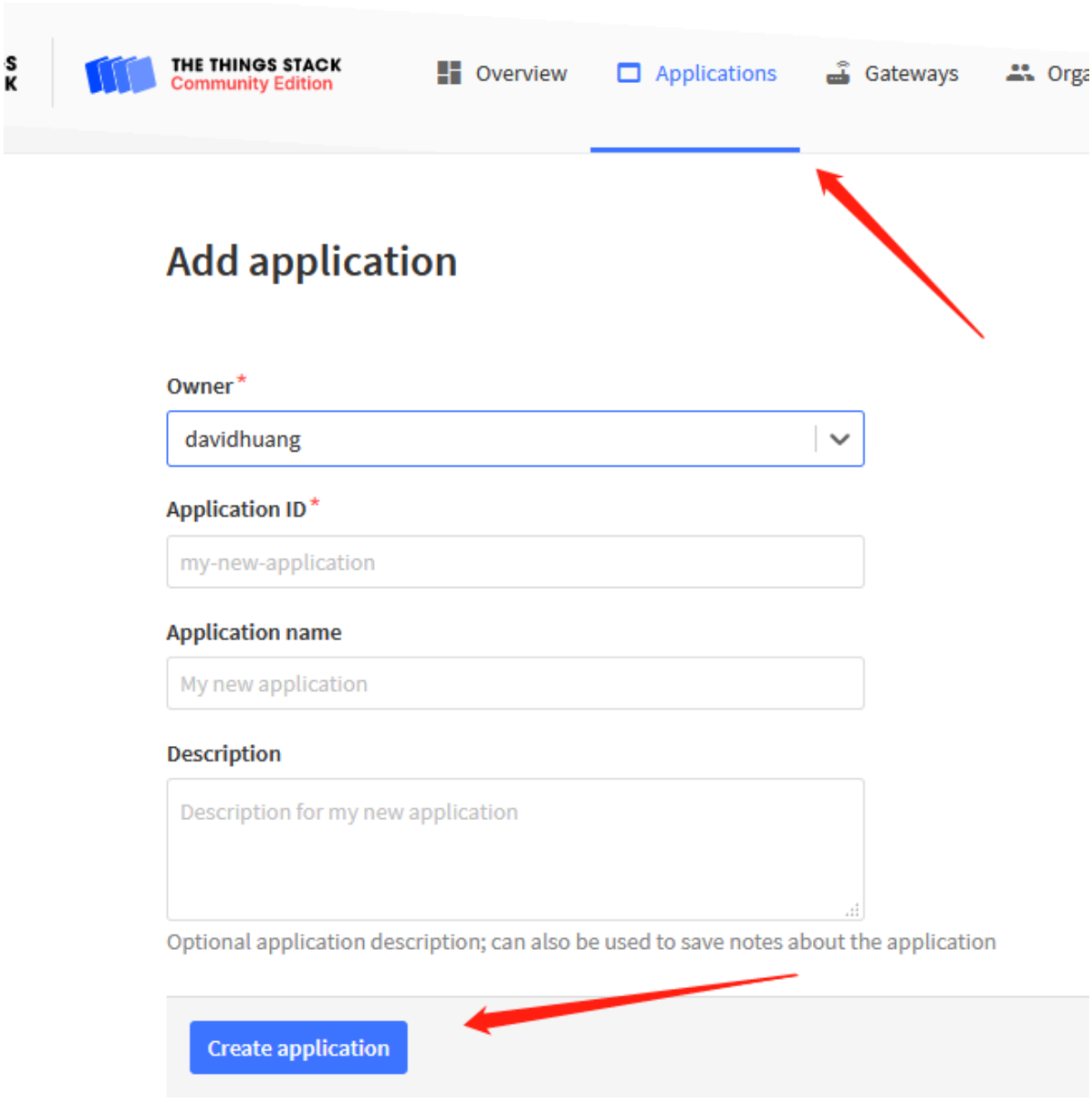
Step 1: Create a device in TTN V3 with the OTAA keys from RS485-LB/LS.

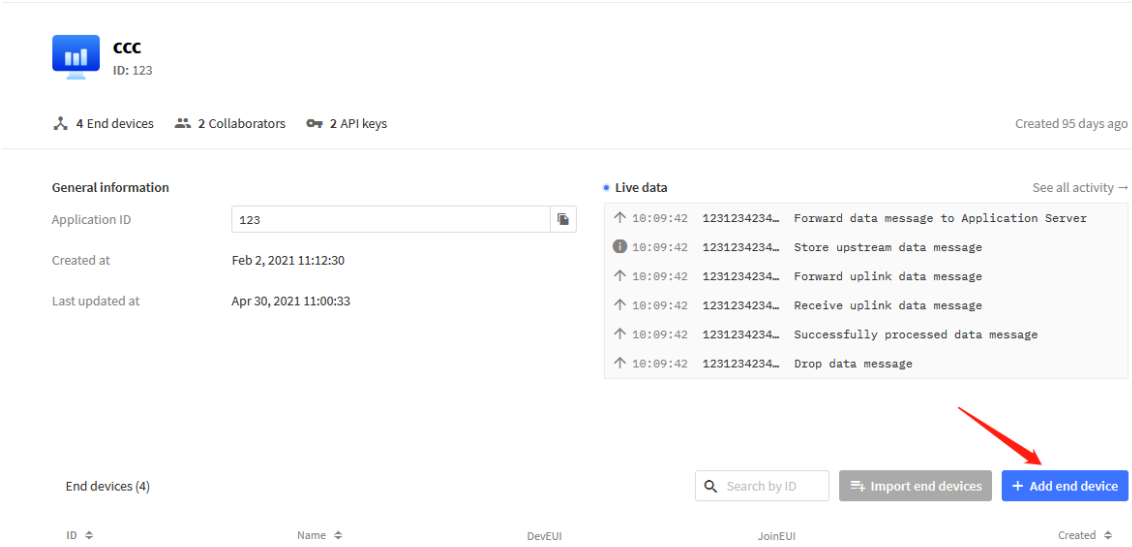
Each RS485-LB/LS is shipped with a sticker with unique device EUI:

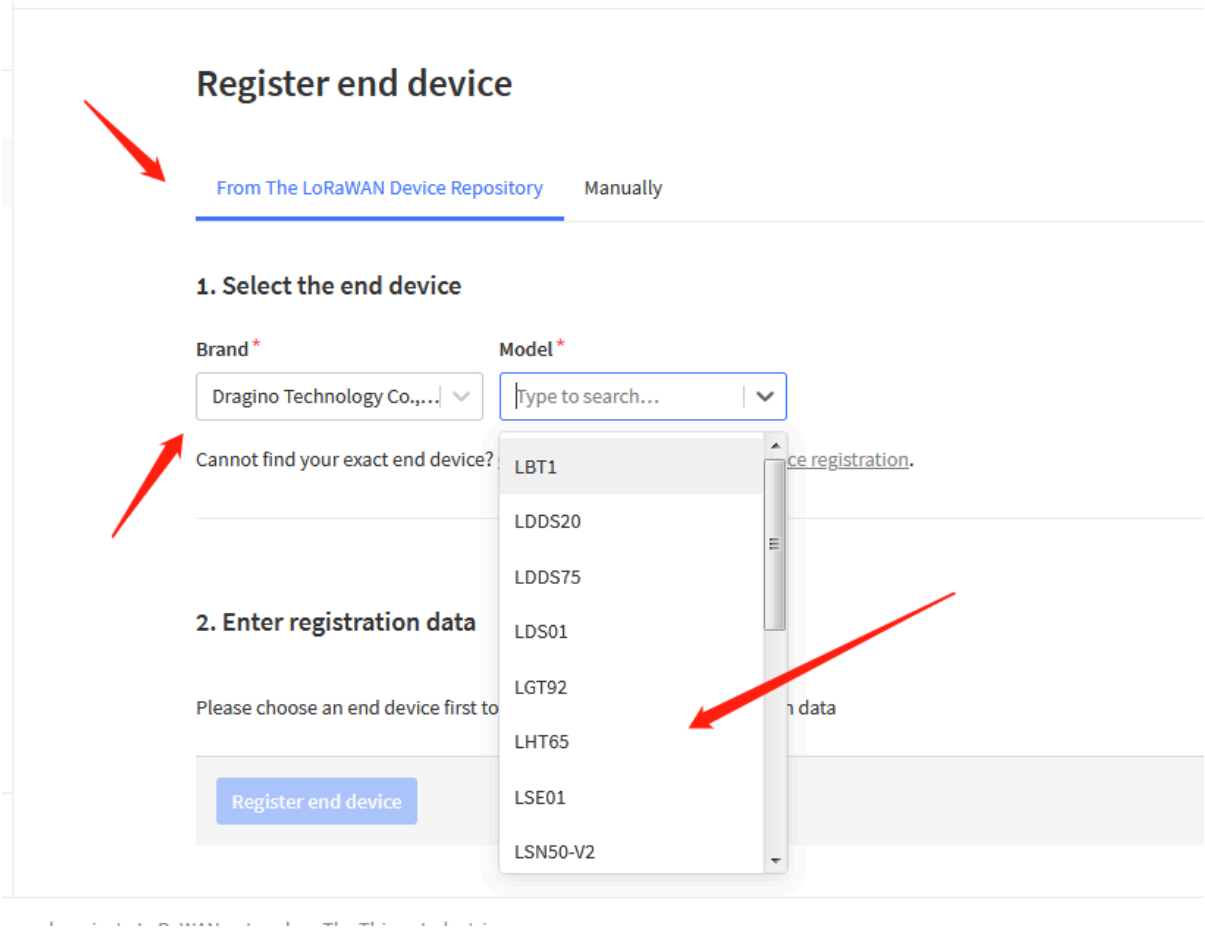


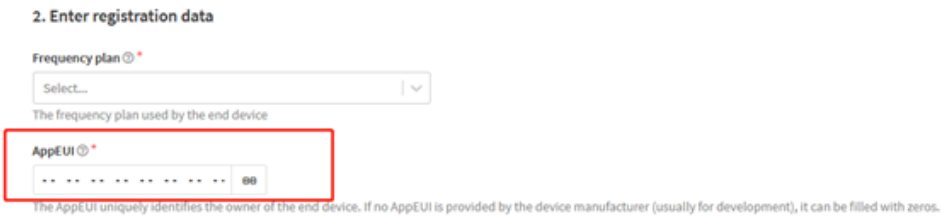
User can enter this key in their LoRaWAN Server portal. Below is TTN V3 screen shot:

Add APP EUI in the application.

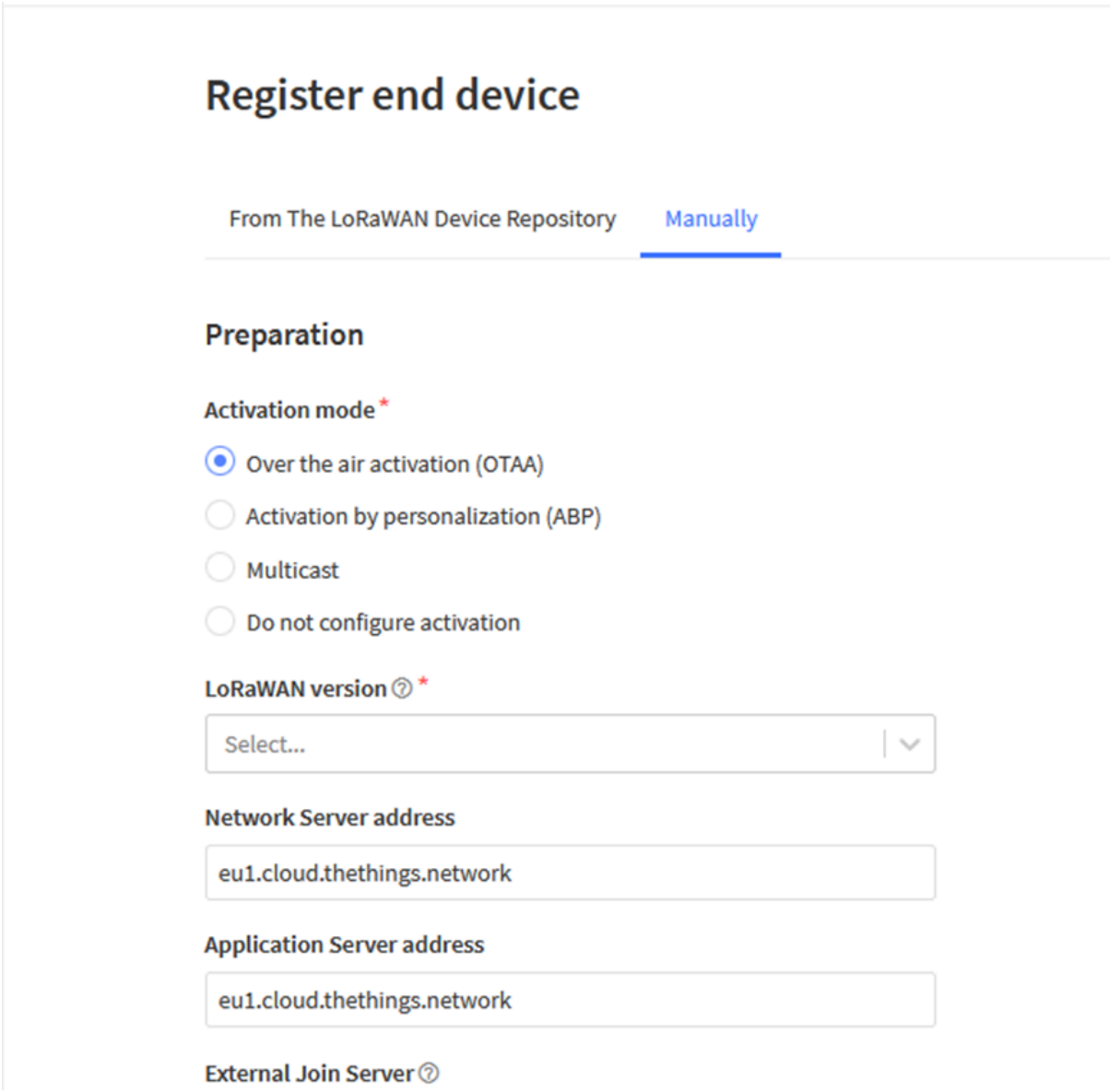



ccc
ID: 123
4 End devices
2 Collaborators
2 API keys
Created 95 days ago
General information
Application ID
123
Created at
Feb 2, 2021 11:12:30
Last updated at
Apr 30, 2021 11:00:33
Live data
See all activity →
10:09:42 1231234234… Forward data message to Application Server
10:09:42 1231234234… Store upstream data message
10:09:42 1231234234… Forward uplink data message
10:09:42 1231234234… Receive uplink data message
10:09:42 1231234234… Successfully processed data message
10:09:42 1231234234… Drop data message
End devices (4)
Search by ID
Import end devices
+ Add end device
ID
Name
DevEUI
JoinEUI
Created



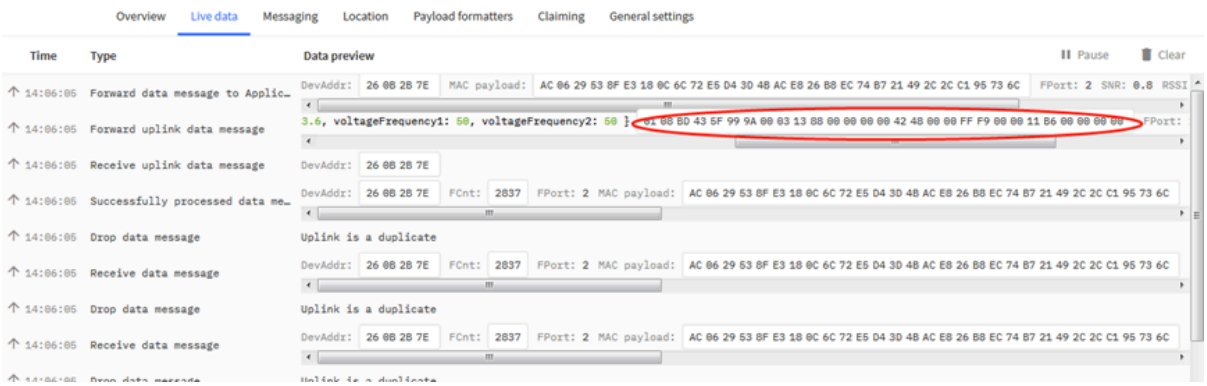


You can also choose to create the device manually.



Add APP KEY and DEV EUI

Step 2: Power on RS485-LB/LS and it will auto join to the TTN V3 network. After join success, it will start to upload message to TTN V3 and user can see in the panel.



2.3 Uplink Payload

2.3.1 Device Status, FPORT=5

Users can use the downlink command(**0x26 01**) to ask LDS12-LB to send device configure detail, include device configure status. RS485-LB/LS will uplink a payload via FPort=5 to server.

The Payload format is as below.

Size(bytes)	1	2	1	1	2
Value	Sensor Model	Firmware Version	Frequency Band	Sub-band	BAT

Sensor Model: For RS485-LB/LS, this value is 0x30

Firmware Version: 0x0100, Means: v1.0.0 version

Frequency Band:

0x01: EU868

0x02: US915

0x03: IN865

0x04: AU915

0x05: KZ865

0x06: RU864

0x07: AS923

0x08: AS923-1

0x09: AS923-2

0x0a: AS923-3

0x0b: CN470

0x0c: EU433

0x0d: KR920

0x0e: MA869

Sub-Band:

AU915 and US915:value 0x00 ~ 0x08

CN470: value 0x0B ~ 0x0C

Other Bands: Always 0x00

Battery Info:

Check the battery voltage.

Ex1: 0x0B45 = 2885mV

Ex2: 0x0B49 = 2889mV

2.3.2 Uplink Payload, FPORT=2

Size(bytes)	2	1	Length depends on the return from the commands

Value	Battery(mV) & Interrupt _Flag	PAYLOAD_VER	If the valid payload is too long and exceed the maximum support payload length in server, server will show payload not provided in the LoRaWAN server.

Below is the decoder for the first 3 bytes. The rest bytes are dynamic depends on different RS485 sensors.

Battery(mV)

Check the battery voltage for RS485-LB/LS.

Ex1: 0x0B45 = 2885mV

Ex2: 0x0B49 = 2889mV

Interrupt_Flag

Ex1: 0x0B45>>15&0x01=0x00 : Normal uplink packet.

Ex2: 0x8B49>>15&0x01=0x01 : Interrupt Uplink Packet.

PAYLOAD_VER

RS485-LB/LS can connect to different sensors. User can set the PAYVER_VER field to tell server how to decode the current payload.

2.4 Payload Decoder file

In TTN, use can add a custom payload so it shows friendly reading

In the page **Applications --> Payload Formats --> Custom --> decoder** to add the decoder from: https://github.com/dragino/dragino-end-node-decoder

2.5 Frequency Plans

The RS485-LB/LS uses OTAA mode and below frequency plans by default. Each frequency band use different firmware, user update the firmware to the corresponding band for their country.

http://wiki.dragino.com/xwiki/bin/view/Main/End%20Device%20Frequency%20Band/

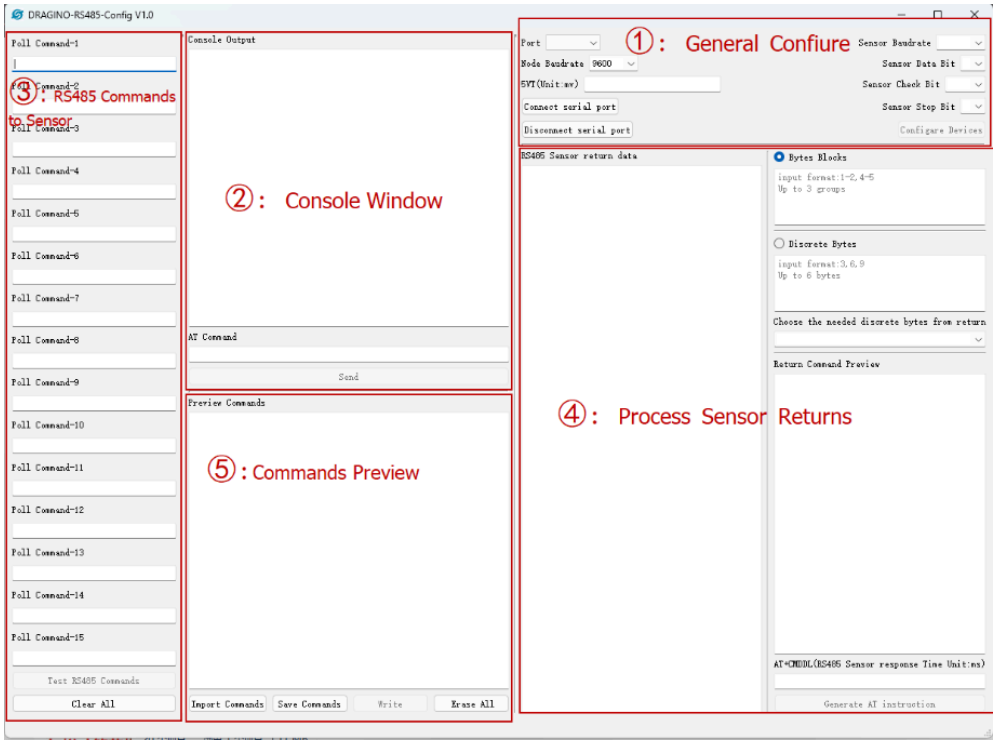
2.6 Configure Device to Read RS485 Sensors

There are plenty of RS485 and TTL level devices in the market and each device has different commands to read the valid data. To support these devices in most flexible, RS485-LB/LS supports flexible command set. User can use Dragino RS485 Tool, AT Commands or LoRaWAN Downlink Command to configure how RS485-LB/LS should read the sensor and how to handle the return from RS485 or TTL sensors.

2.6.1 Method 1 -- via RS485 Configure Tool

Use the RS485 Configure tool is the recommand method. Please see the instruction of how to use the tool:

- **RS485 Configure Tool Instruction**



2.6.2 Method 2 -- via AT Commands

2.6.2.1 Configure UART settings for RS485 or TTL communication

RS485-LB/LS can connect to either RS485 sensors or TTL sensor. User need to specify what type of sensor need to connect.

1. RS485-MODBUS mode:

AT+MOD=1 // Support RS485-MODBUS type sensors. User can connect multiply RS485 , Modbus sensors to the A / B pins.

2. TTL mode:

AT+MOD=2 // Support TTL Level sensors, User can connect one TTL Sensor to the TXD/RXD/GND pins.

RS485-LB/LS default UART settings is **9600, no parity, stop bit 1,data bit 8**. If the sensor has a different settings, user can change the RS485-LB/LS setting to match.

AT Commands	Description	Example
AT+BAUDR	Set the baud rate. Default Value is: 9600.	AT+BAUDR=9600 Options: (1200,2400,4800,14400,19200,115200)
AT+PARITY	Set UART parity. Default Value is: no parity.	AT+PARITY=0 Option: 0: no parity, 1: odd parity, 2: even parity
AT+STOPBIT	Set serial stopbit Default Value is: 1bit.	AT+STOPBIT=1 for 1 bit AT+STOPBIT=2 for 2 bits
AT+DATABIT	Set serial databit.	AT+DATABIT=7 for 7 bits

Default Value is: 8bits.	AT+DATABIT=8 for 8 bits

Example (Soil three-parameter detector) :

Wiring the UART sensor

GND <--------> GND
TX <--------> RX
RX <--------> TX
VCC <--------> 3.3/5V

Set the correct configuration:

AT+BAUDR=9600

AT+PARITY=0

AT+STOPBIT=1

AT+DATABIT=8

If the sensor needs 5v. Need to move the switch position to 5v and then use the command **AT+5VT=30000**

Configuration read command:

AT+CFGDEV=FE 03 00 00 00 03 11 C4,0

FE: Station address

03: Function code

00 00: Register start address

00 03: Number of registers

11 04: Check code

```
AT+CFGDEV=FE 03 00 00 00 03 11 C4,0

AT+CFGDEV=fe 03 00 00 00 03 11 c4 ,0
RETURN DATA:
fe 03 06 00 00 09 49 00 00 b6 cb

OK
```

Use AT+COMMAND1 to set it as a command, and use AT+DATACUT1 to intercept the bytes I need

```
AT+SETMAXNBTRANS=1,0
AT+DISFCNTCHECK=0
AT+DISMACANS=0
AT+COMMAND1=fe 03 00 00 00 03 11 c4 ,0      AT+SEARCH1=0,0
AT+DATACUT1=11,2,4~9      AT+CMDDL1=0
AT+COMMAND2=0,0      AT+SEARCH2=0,0      AT+DATACUT2=0,0,0      AT+CMDDL2=0
AT+COMMAND3=0,0      AT+SEARCH3=0,0      AT+DATACUT3=0,0,0      AT+CMDDL3=0
AT+COMMAND4=0,0      AT+SEARCH4=0,0      AT+DATACUT4=0,0,0      AT+CMDDL4=0
AT+COMMAND5=0,0      AT+SEARCH5=0,0      AT+DATACUT5=0,0,0      AT+CMDDL5=0
AT+COMMAND6=0,0      AT+SEARCH6=0,0      AT+DATACUT6=0,0,0      AT+CMDDL6=0
AT+COMMAND7=0,0      AT+SEARCH7=0,0      AT+DATACUT7=0,0,0      AT+CMDDL7=0
AT+COMMAND8=0,0      AT+SEARCH8=0,0      AT+DATACUT8=0,0,0      AT+CMDDL8=0
AT+COMMAND9=0,0      AT+SEARCH9=0,0      AT+DATACUT9=0,0,0      AT+CMDDL9=0
AT+COMMANDA=0,0      AT+SEARCHA=0,0      AT+DATACUTA=0,0,0      AT+CMDDLA=0
AT+COMMANDB=0,0      AT+SEARCHB=0,0      AT+DATACUTB=0,0,0      AT+CMDDLB=0
AT+COMMANDC=0,0      AT+SEARCHC=0,0      AT+DATACUTC=0,0,0      AT+CMDDLC=0
```

upload payload:

```
[1870882]***** UpLinkCounter= 31 *****
[1870884]TX on freq 904900000 Hz at DR 3
[1870945]txDone
[1875932]RX on freq 926300000 Hz at DR 13
[1875957]rxTimeOut
[1876948]RX on freq 923300000 Hz at DR 8
[1877008]rxTimeOut

CMD1      = fe 03 00 00 00 03 11 c4
RETURN1   = fe 03 06 00 00 09 49 00 00 b6 cb
Payload   = 0d 32 01 00 00 09 49 00 00
```

2.6.2.2 Configure sensors

Some sensors might need to configure before normal operation. User can configure such sensor via PC or through RS485-LB/LS AT Commands **AT+CFGDEV**.

When user issue an **AT+CFGDEV** command, Each **AT+CFGDEV** equals to send a command to the RS485 or TTL sensors. This command will only run when user input it and won't run during each sampling.

AT Commands	Description	Example
AT+CFGDEV	This command is used to configure the RS485/TTL devices; they won't be used during sampling. AT+CFGDEV=xx xx xx xx xx xx xx xx xx xx xx xx, mm: 0: no CRC, 1: add CRC-16/MODBUS in the end of this command	AT+CFGDEV=xx xx xx xx xx xx xx xx xx xx xx xx,m

Detail of AT+CFGDEV command see AT+CFGDEV detail.

2.6.2.3 Configure read commands for each sampling

RS485-LB/LS is a battery powered device; it will sleep most of time. And wake up on each period and read RS485 / TTL sensor data and uplink.

During each sampling, we need to confirm what commands we need to send to the sensors to read data. After the RS485/TTL sensors send back the value, it normally includes some bytes and we only need a few from them for a shorten payload.

To save the LoRaWAN network bandwidth, we might need to read data from different sensors and combine their valid value into a short payload.

This section describes how to achieve above goals.

During each sampling, the RS485-LB/LS can support 15 commands to read sensors. And combine the return to one or several uplink payloads.

Command from RS485-LB/LS to Sensor:

RS485-LB/LS can send out pre-set max 15 strings via **AT+COMMAD1**, **ATCOMMAND2**,..., to **AT+COMMANDF** . All commands are of same grammar.

Handle return from sensors to RS485-LB/LS:

After RS485-LB/LS send out a string to sensor, RS485-LB/LS will wait for the return from RS485 or TTL sensor. And user can specify how to handle the return, by **AT+DATACUT or AT+SEARCH commands**

- **AT+DATACUT**

When the return value from sensor have fix length and we know which position the valid value we should get, we can use AT+DATACUT command.

- **AT+SEARCH**

When the return value from sensor is dynamic length and we are not sure which bytes the valid data is, instead, we know what value the valid value following. We can use AT+SEARCH to search the valid value in the return string.

Define wait timeout:

Some RS485 device might has longer delay on reply, so user can use AT+CMDDL to set the timeout for getting reply after the RS485 command is sent. For example, AT+CMDDL1=1000 to send the open time to 1000ms

After we got the valid value from each RS485 commands, we need to combine them together with the command **AT +DATAUP**.

Examples:

Below are examples for the how above AT Commands works.

AT+COMMANDx : This command will be sent to RS485/TTL devices during each sampling, Max command length is 14 bytes. The grammar is:

AT+COMMANDx=xx xx xx xx xx xx xx xx xx xx xx xx,m

xx xx xx xx xx xx xx xx xx xx xx xx: The RS485 command to be sent

m: 0: no CRC, 1: add CRC-16/MODBUS in the end of this command

For example, if we have a RS485 sensor. The command to get sensor value is: 01 03 0B B8 00 02 46 0A. Where 01 03 0B B8 00 02 is the Modbus command to read the register 0B B8 where stored the sensor value. The 46 0A is the CRC-16/MODBUS which calculate manually.

In the RS485-LB/LS, we should use this command AT+COMMAND1=01 03 0B B8 00 02,1 for the same.

If a single command exceeds 14 bytes, you can use the command splicing function.

When AT+CMDDLx=1, the commands of AT+COMMANDx and AT+COMMAND(x+1) will be merged.

Examples: To send 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F data it should be configured:

AT+COMMAND1=00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D,0

AT+COMMAND1=1

AT+COMMAND2=0E 0F,0

AT+SEARCHx: This command defines how to handle the return from AT+COMMANDx.

AT+SEARCHx=aa,xx xx xx xx xx

- **aa: 1: prefix match mode; 2: prefix and suffix match mode**
- **xx xx xx xx xx: match string. Max 5 bytes for prefix and 5 bytes for suffix**

Examples:

1) For a return string from AT+COMMAND1: 16 0c 1e 56 34 2e 30 58 5f 36 41 30 31 00 49

If we set AT+SEARCH1=1,1E 56 34. (max 5 bytes for prefix)

The valid data will be all bytes after 1E 56 34 , so it is **2e 30 58 5f 36 41 30 31 00 49**

```
CMD1    = 11 01 1e d0
SEARCH1  = 1e 56 34
RETURN1  = 2e 30 58 5f 36 41 30 31 00 49
Payload  = 8d 2d 01 2e 30 58 5f 36 41 30 31 00 49
```

2) For a return string from AT+COMMAND1: 16 0c 1e 56 34 2e 30 58 5f 36 41 30 31 00 49

If we set AT+SEARCH1=2, 1E 56 34+31 00 49

Device will search the bytes between 1E 56 34 and 31 00 49. So it is **2e 30 58 5f 36 41 30**

```
CMD1    = 11 01 1e d0
SEARCH1  = 1e 56 34 + 31 00 49
RETURN1  = 2e 30 58 5f 36 41 30
Payload  = 8d 2f 01 2e 30 58 5f 36 41 30
```

AT+DATACUTx : This command defines how to handle the return from AT+COMMANDx, max return length is 100 bytes.

AT+DATACUTx=a,b,c

- **a: length for the return of AT+COMMAND**
- **b:1: grab valid value by byte, max 6 bytes. 2: grab valid value by bytes section, max 3 sections.**
- **c: define the position for valid value.**

Examples:

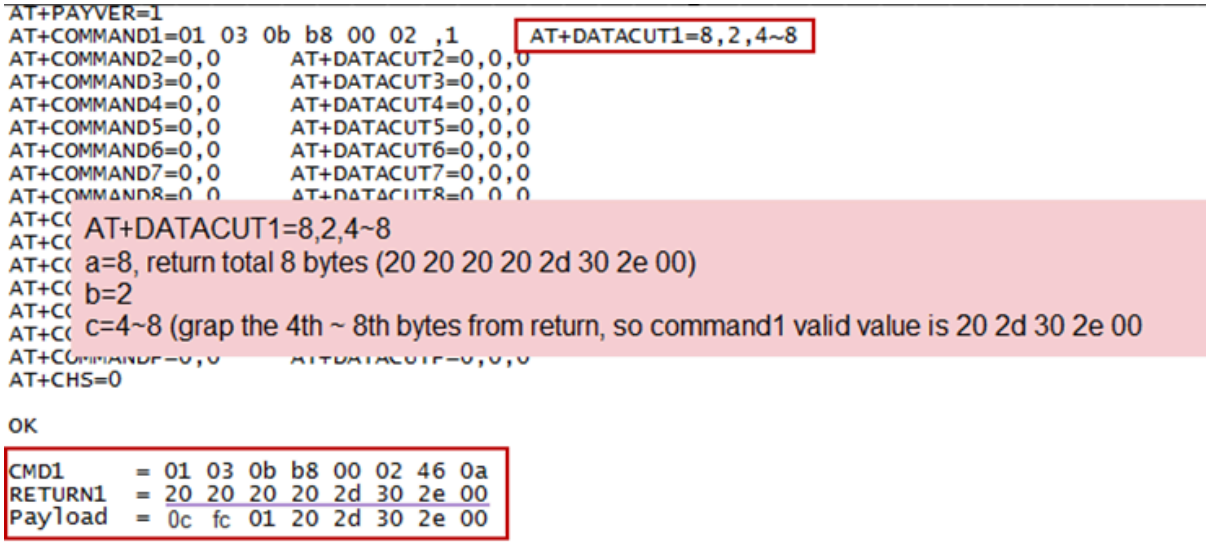
- **Grab bytes:**

```
AT+PAYVER=1
AT+COMMAND1=01 03 0b b8 00 02 ,1      AT+DATACUT1=10,1,9+4+6+8+1+3
AT+COMMAND2=0,0       AT+DATACUT2=0,0,0
AT+COMMAND3=0,0       AT+DATACUT3=0,0,0
AT+COMMAND4=0,0       AT+DATACUT4=0,0,0
AT+COMMAND5=0,0       AT+DATACUT5=0,0,0
```

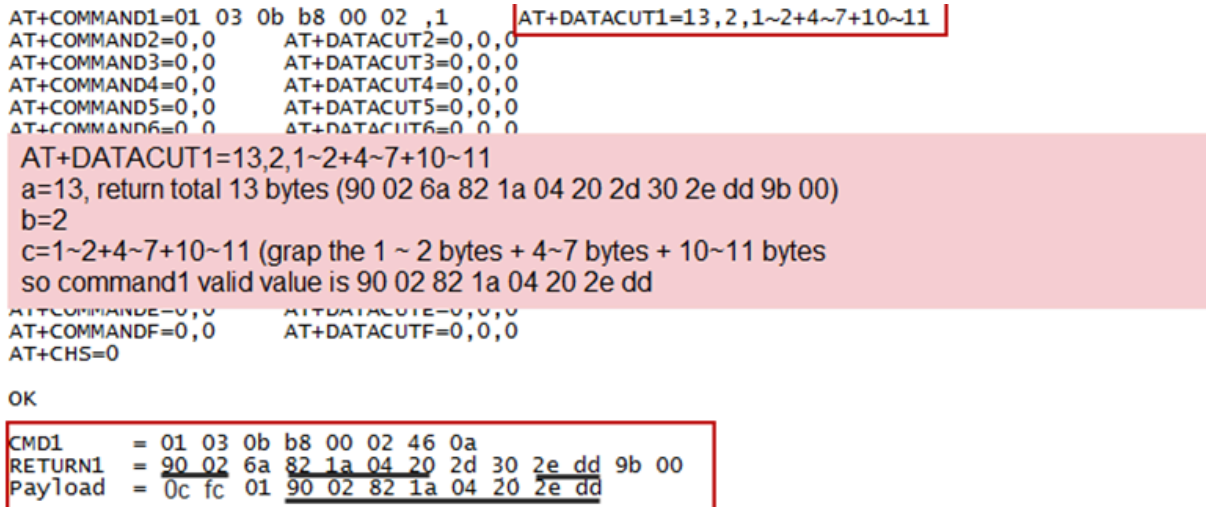
AT+DATACUT1=10,1,9+4+6+8+1+3
a=10, return total 10 bytes (20 20 20 20 2d 30 2e 32 20 75)
b=1 grab byte.
c=9+4+6+8+1+3 (grap the 9th , 4th, 6th, 8th, 1th, 3rd byte and link them together by grab sequence
so command1 valid value is 20 20 30 32 20 20

```
AT+COMMANDE=0,0       AT+DATACUTE=0,0,0
AT+COMMANDF=0,0       AT+DATACUTF=0,0,0
AT+CHS=0

OK

CMD1     = 01 03 0b b8 00 02 46 0a
RETURN1  = 20 20 20 20 2d 30 2e 32 20 75
Payload  = 0c fc 01 20 20 30 32 20 20
```

- **Grab a section.**



- **Grab different sections.**



Note:

AT+SEARCHx and **AT+DATACUTx** can be used together, if both commands are set, RS485-LB/LS will first process AT+SEARCHx on the return string and get a temporary string, and then process AT+DATACUTx on this temporary string to get the final payload. In this case, AT+DATACUTx need to set to format **AT +DATACUTx=0,xx,xx** where the return bytes set to **0**.

Example:

AT+COMMAND1=11 01 1E D0,0

AT+SEARCH1=1,1E 56 34

AT+DATACUT1=0,2,1~5

Return string from AT+COMMAND1: 16 0c 1e 56 34 2e 30 58 5f 36 41 30 31 00 49

String after SEARCH command: 2e 30 58 5f 36 41 30 31 00 49

Valid payload after DataCUT command: 2e 30 58 5f 36

```
CMD1     = 11 01 1e d0
SEARCH1  = 1e 56 34
RETURN1  = 2e 30 58 5f 36 41 30 31 00 49
Payload  = 8d 2d 01 2e 30 58 5f 36
```

2.6.2.4 Compose the uplink payload

Through AT+COMMANDx and AT+DATACUTx we got valid value from each RS485 commands, Assume these valid value are RETURN1, RETURN2, .., to RETURNx. The next step is how to compose the LoRa Uplink Payload by these RETURNs. The command is **AT+DATAUP.**

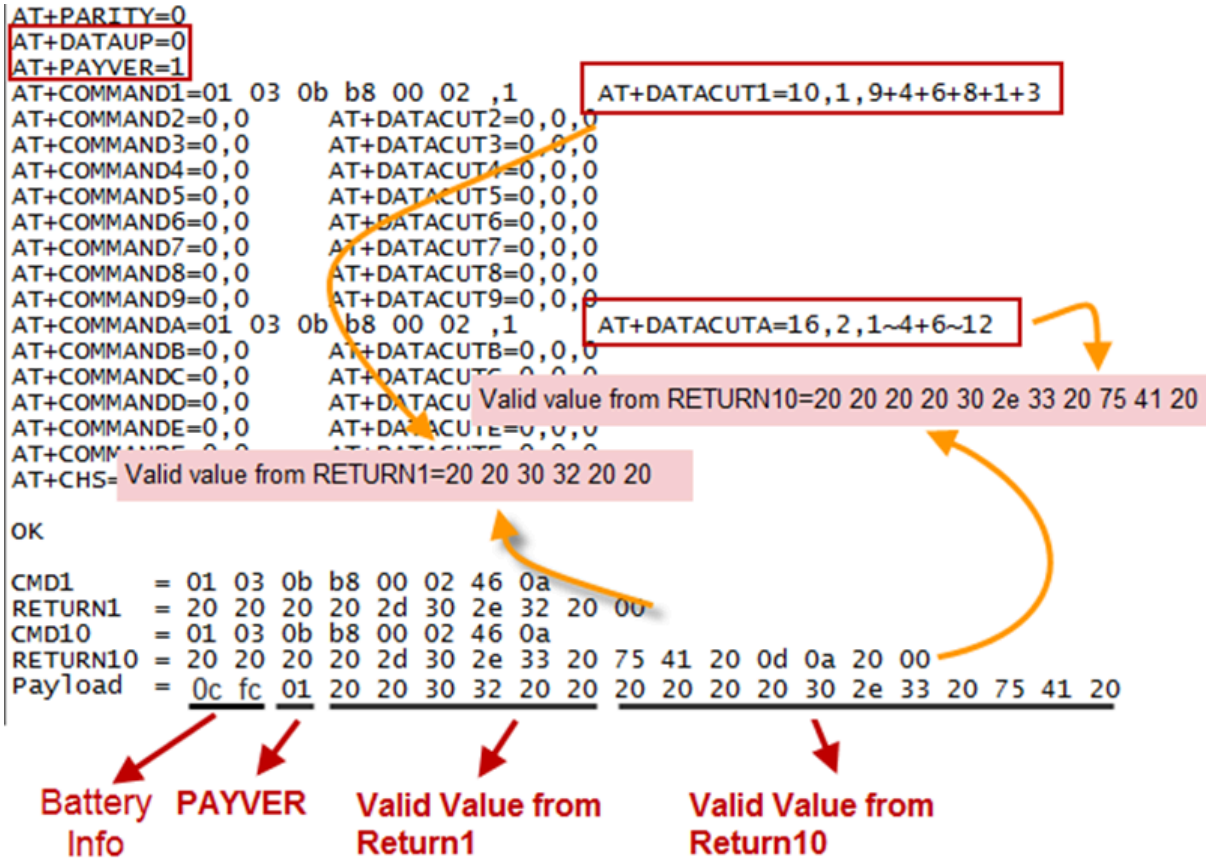
Examples: AT+DATAUP=0

Compose the uplink payload with value returns in sequence and send with **A SIGNLE UPLINK**.

Final Payload is

Battery Info+PAYVER + VALID Value from RETURN1 + Valid Value from RETURN2 + ... + RETURNx

Where PAYVER is defined by AT+PAYVER, below is an example screen shot.



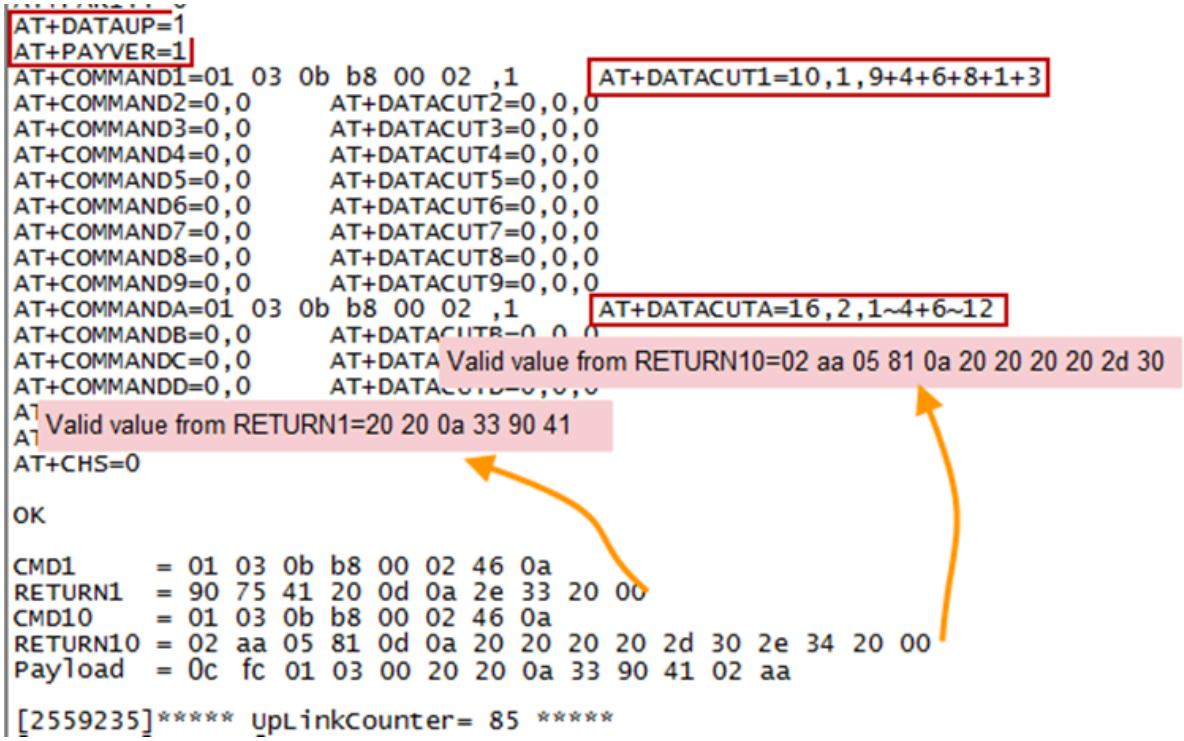
Examples: AT+DATAUP=1

Compose the uplink payload with value returns in sequence and send with **Multiply UPLINKs**.

Final Payload is

Battery Info+PAYVER + PAYLOAD COUNT + PAYLOAD# + DATA

1. Battery Info (2 bytes): Battery voltage
2. PAYVER (1 byte): Defined by AT+PAYVER
3. PAYLOAD COUNT (1 byte): Total how many uplinks of this sampling.
4. PAYLOAD# (1 byte): Number of this uplink. (from 0,1,2,3...,to PAYLOAD COUNT)
5. DATA: Valid value: max 6 bytes(US915 version here, Notice*!) for each uplink so each uplink <= 11 bytes. For the last uplink, DATA will might less than 6 bytes



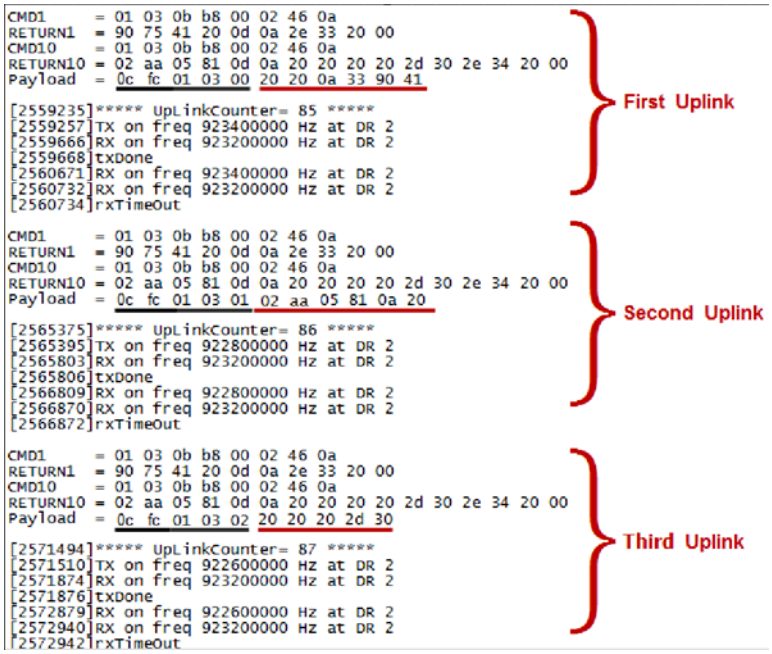
So totally there will be 3 uplinks for this sampling, each uplink includes 6 bytes DATA

DATA1=RETURN1 Valid Value = 20 20 0a 33 90 41

DATA2=1st ~ 6th byte of Valid value of RETURN10= 02 aa 05 81 0a 20

DATA3=7th ~ 11th bytes of Valid value of RETURN10 = 20 20 20 2d 30

Below are the uplink payloads:



Notice: the Max bytes is according to the max support bytes in different Frequency Bands for lowest SF. As below:

* For AU915/AS923 bands, if UplinkDwell time=0, max 51 bytes for each uplink (so 51 -5 = 46 max valid date)

* For AU915/AS923 bands, if UplinkDwell time=1, max 11 bytes for each uplink (so 11 -5 = 6 max valid date).

* For US915 band, max 11 bytes for each uplink (so 11 -5 = 6 max valid date).

* For all other bands: max 51 bytes for each uplink (so 51 -5 = 46 max valid date).

* **When AT+DATAUP=1, the maximum number of segments is 15, and the maximum total number of bytes is 1500;**

When AT+DATAUP=1 and AT+ADR=0, the maximum number of bytes of each payload is determined by the DR value. (Since v1.4.0)

- **If the data is empty, return to the** `display (Since v1.4.0)`

1) When **AT+MOD=1**, if the data intercepted by **AT+DATACUT** or **AT+MBFUN** is empty, it will display **NULL**, and the payload will be filled with **n FFs**.

```
CMD1     = 01 03 00 00 00 2f 04 16
RETURN1  = NULL
Payload  = 01 02 00 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff

[68528]***** UpLinkCounter= 4 *****
[68530]TX on freq 867900000 Hz at DR 5
[68652]RX on freq 869525000 Hz at DR 0
[68654]txDone
[73643]RX on freq 867900000 Hz at DR 5
[73670]RX on freq 869525000 Hz at DR 0
[73672]rxTimeOut

CMD1     = 01 03 00 00 00 2f 04 16
RETURN1  = NULL
Payload  = 01 02 01 ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff

[77352]***** UpLinkCounter= 5 *****
[77354]TX on freq 867500000 Hz at DR 5
[77476]RX on freq 869525000 Hz at DR 0
[77478]txDone
[82467]RX on freq 867500000 Hz at DR 5
[82495]RX on freq 869525000 Hz at DR 0
```

2) When **AT+MOD=2**, if the data intercepted by **AT+DATACUT** or **AT+MBFUN** is empty, it will display **NULL**, and the payload will be filled with **n 00s**.

```
CMD1      = 01 03 00 00 00 06 c5 c8
RETURN1   = NULL
CMD2      = 01 03 00 00 00 08 44 0c
RETURN2   = NULL
CMD15     = 01 03 00 00 00 04 44 09
RETURN15  = NULL
Payload   = 0e 59 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

[9561]***** UpLinkCounter= 0 *****
[9562]TX on freq 868500000 Hz at DR 0
[12032]txDone
[17062]RX on freq 868500000 Hz at DR 0

ADR Message:
TX Datarate change to 4
TxPower change to 0
NbRep change to 1

[18217]rxDone
Rssi= -50
```

2.6.3 Uplink on demand

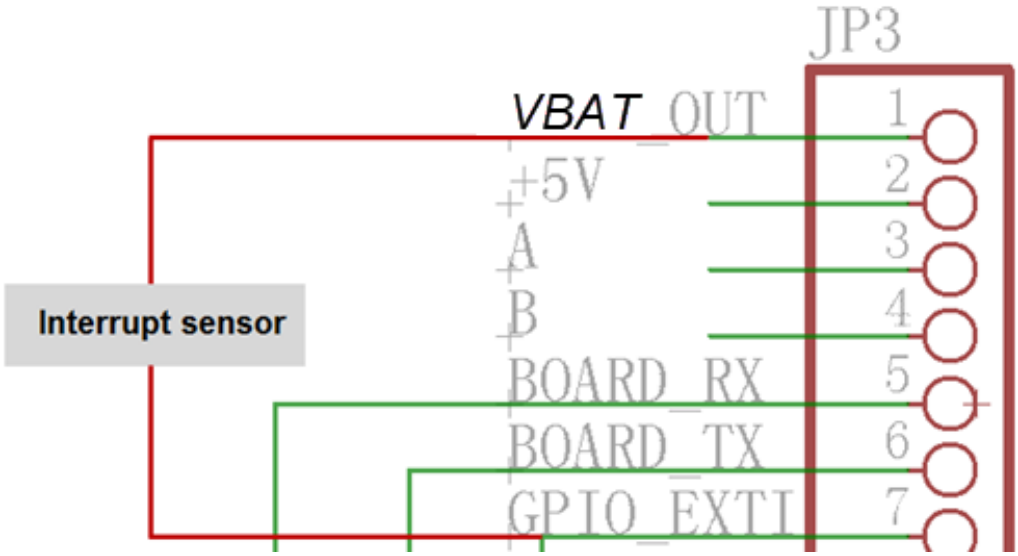
Except uplink periodically, RS485-LB/LS is able to uplink on demand. The server sends downlink command to RS485-LB/LS and RS485 will uplink data base on the command.

Downlink control command:

0xA8 command: Send a command to RS485-LB/LS and uplink the output from sensors.

2.6.4 Uplink on Interrupt

Put the interrupt sensor between 3.3v_out and GPIO ext.



AT+INTMOD=0 Disable Interrupt

AT+INTMOD=1 Interrupt trigger by rising or falling edge.

AT+INTMOD=2 Interrupt trigger by falling edge. (Default Value)

AT+INTMOD=3 Interrupt trigger by rising edge.

3. Configure RS485-LB/LS

3.1 Configure Methods

RS485-LB/LS supports below configure method:

- AT Command via Bluetooth Connection (**Recommended**): BLE Configure Instruction.
- AT Command via UART Connection : See UART Connection.
- LoRaWAN Downlink. Instruction for different platforms: See IoT LoRaWAN Server section.

3.2 General Commands

These commands are to configure:

- General system settings like: uplink interval.
- LoRaWAN protocol & radio related command.

They are same for all Dragino Devices which support DLWS-005 LoRaWAN Stack. These commands can be found on the wiki:

http://wiki.dragino.com/xwiki/bin/view/Main/End%20Device%20AT%20Commands%20and%20Downlink%20Command/

3.3 Commands special design for RS485-LB/LS

These commands only valid for RS485-LB/LS, as below:

3.3.1 Choose Device Type (RS485 or TTL)

RS485-LB/LS can connect to either RS485 sensors or TTL sensor. User need to specify what type of sensor need to connect.

- **AT Command**

AT+MOD=1 // Set to support RS485-MODBUS type sensors. User can connect multiply RS485 , Modbus sensors to the A / B pins.

AT+MOD=2 // Set to support TTL Level sensors, User can connect one TTL Sensor to the TXD/RXD/GND pins.

- **Downlink Payload**

0A aa --> same as AT+MOD=aa

3.3.2 RS485 Debug Command (AT+CFGDEV)

This command is used to configure the RS485 or TTL sensors; they won't be used during sampling. Max Length of AT+CFGDEV is **40 bytes**.

- **AT Command**

AT+CFGDEV=xx xx xx xx xx xx xx xx xx xx xx xx,m m: 0: no CRC; 1: add CRC-16/MODBUS in the end of this command.

- **Downlink Payload**

Format: **A8 MM NN XX XX XX XX YY**

Where:

- MM: 1: add CRC-16/MODBUS ; 0: no CRC
- NN: The length of RS485 command
- XX XX XX XX: RS485 command total NN bytes
- YY: How many bytes will be uplink from the return of this RS485 command, if YY=0, RS485-LB/LS will execute the downlink command without uplink; if YY>0, RS485-LB/LS will uplink total YY bytes from the output of this RS485 command

Example 1:

To connect a Modbus Alarm with below commands.

- The command to active alarm is: 0A 05 00 04 00 01 **4C B0**. Where 0A 05 00 04 00 01 is the Modbus command to read the register 00 40 where stored the DI status. The 4C B0 is the CRC-16/MODBUS which calculate manually.
- The command to deactivate alarm is: 0A 05 00 04 00 00 **8D 70**. Where 0A 05 00 04 00 00 is the Modbus command to read the register 00 40 where stored the DI status. The 8D 70 is the CRC-16/MODBUS which calculate manually.

So if user want to use downlink command to control to RS485 Alarm, he can use:

A8 01 06 0A 05 00 04 00 01 00: to activate the RS485 Alarm

A8 01 06 0A 05 00 04 00 00 00: to deactivate the RS485 Alarm

A8 is type code and 01 means add CRC-16/MODBUS at the end, the 3rd byte is 06, means the next 6 bytes are the command to be sent to the RS485 network, the final byte 00 means this command don't need to acquire output.

Example 2:

Check TTL Sensor return:

```
AT+CFGDEV=11 01 1e d0 ,0
RETURN DATA:16 0c 1e 56 34 2e 30 58 5f 36 41 30 31 00 49

OK
```

3.3.3 Set Payload version

This is the first byte of the uplink payload. RS485-LB/LS can connect to different sensors. User can set the PAYVER field to tell server how to decode the current payload.

- **AT Command:**

AT+PAYVER: Set PAYVER field = 1

- **Downlink Payload:**

0xAE 01 --> Set PAYVER field = 0x01

0xAE 0F --> Set PAYVER field = 0x0F

3.3.4 Set RS485 Sampling Commands

AT+COMMANDx, AT+DATACUTx and AT+SEARCHx

These three commands are used to configure how the RS485-LB polling data from Modbus device. Detail of usage please see : polling RS485 device.

- **AT Command:**

AT+COMMANDx: **Configure RS485 read command to sensor.**

AT+DATACUTx: **Configure how to handle return from RS485 devices.**

AT+SEARCHx: **Configure search command**

- **Downlink Payload:**

0xAF downlink command can be used to set AT+COMMANDx or AT+DATACUTx.

Note : if user use AT+COMMANDx to add a new command, he also need to send AT+DATACUTx downlink.

Format: AF MM NN LL XX XX XX XX YY

Where:

- MM: the ATCOMMAND or AT+DATACUT to be set. Value from 01 ~ 0F,
- NN: 0: no CRC; 1: add CRC-16/MODBUS ; 2: set the AT+DATACUT value.
- LL: The length of AT+COMMAND or AT+DATACUT command
- XX XX XX XX: AT+COMMAND or AT+DATACUT command
- YY: If YY=0, RS485-LB/LS will execute the downlink command without uplink; if YY=1, RS485-LB/LS will execute an uplink after got this command.

Example:

AF 03 01 06 0A 05 00 04 00 01 00: Same as AT+COMMAND3=0A 05 00 04 00 01,1

AF 03 02 06 10 01 05 06 09 0A 00: Same as AT+DATACUT3=16,1,5+6+9+10

AF 03 02 06 0B 02 05 07 08 0A 00: Same as AT+DATACUT3=11,2,5~7+8~10

0xAB downlink command can be used for set AT+SEARCHx

Example: AB aa 01 03 xx xx xx (03 here means there are total 3 bytes after 03) So

- AB aa 01 03 xx xx xx same as AT+SEARCHaa=1,xx xx xx
- AB aa 02 03 xx xx xx 02 yy yy(03 means there are 3 bytes after 03, they are xx xx xx;02 means there are 2 bytes after 02, they are yy yy) so the commands

AB aa 02 03 xx xx xx 02 yy yy same as **AT+SEARCHaa=2,xx xx xx+yy yy**

3.3.5 Fast command to handle MODBUS device

AT+MBFUN is valid since v1.3 firmware version. The command is for fast configure to read Modbus devices. It is only valid for the devices which follow the MODBUS-RTU protocol.

This command is valid since v1.3 firmware version

AT+MBFUN has only two value:

- **AT+MBFUN=1**: Enable Modbus reading. And get response base on the MODBUS return

AT+MBFUN=1, device can auto read the Modbus function code: 01, 02, 03 or 04. AT+MBFUN has lower priority vs AT+DATACUT command. If AT+DATACUT command is configured, AT+MBFUN will be ignore.

- **AT+MBFUN=0**: Disable Modbus fast reading.

Example:

- AT+MBFUN=1 and AT+DATACUT1/AT+DATACUT2 are not configure (0,0,0).
- AT+COMMAND1= 01 03 00 10 00 08,1 --> read slave address 01 , function code 03, start address 00 01, quantity of registers 00 08.
- AT+COMMAND2= 01 02 00 40 00 10,1 --> read slave address 01 , function code 02, start address 00 40, quantity of inputs 00 10.

```
AT+COMMAND1=01 03 00 10 00 08 ,1   AT+DATACUT1=0,0,0   AT+CMDDL1=0
AT+COMMAND2=01 02 00 40 00 10 ,1   AT+DATACUT2=0,0,0   AT+CMDDL2=0
AT+COMMAND3=0,0    AT+DATACUT3=0,0,0    AT+CMDDL3=0
AT+COMMAND4=0,0    AT+DATACUT4=0,0,0    AT+CMDDL4=0
AT+COMMAND5=0,0    AT+DATACUT5=0,0,0    AT+CMDDL5=0
AT+COMMAND6=0,0    AT+DATACUT6=0,0,0    AT+CMDDL6=0
AT+COMMAND7=0,0    AT+DATACUT7=0,0,0    AT+CMDDL7=0
AT+COMMAND8=0,0    AT+DATACUT8=0,0,0    AT+CMDDL8=0
AT+COMMAND9=0,0    AT+DATACUT9=0,0,0    AT+CMDDL9=0
AT+COMMANDA=0,0    AT+DATACUTA=0,0,0    AT+CMDDLA=0
AT+COMMANDB=0,0    AT+DATACUTB=0,0,0    AT+CMDDLB=0
AT+COMMANDC=0,0    AT+DATACUTC=0,0,0    AT+CMDDLC=0
AT+COMMANDD=0,0    AT+DATACUTD=0,0,0    AT+CMDDLD=0
AT+COMMANDE=0,0    AT+DATACUTE=0,0,0    AT+CMDDLE=0
AT+COMMANDF=0,0    AT+DATACUTF=0,0,0    AT+CMDDLF=0

Start Tx events

OK
```

```
CMD1     = 01 03 00 10 00 08 45 c9
RETURN1  = 01 03 10 01 00 05 ff 00 00 00 00 01 03 00 00 00 00 00 00 86 fe   --> DATA1:8 register values
CMD2     = 01 02 00 40 00 10 78 12
RETURN2  = 01 02 02 20 00 a0 78   --> DATA2:16 register values
Payload  =  0C FC  01 01 00 05 ff 00 00 00 00 01 03 00 00 00 00 00 00 20 00   --> DATA:DATA1+DATA2

[177893]***** UpLinkCounter= 4 *****
[177895]TX on freq 867700000 Hz at DR 3
[178145]RX on freq 869525000 Hz at DR 3
[178148]txDone
[179143]RX on freq 867700000 Hz at DR 3
[179183]RX on freq 869525000 Hz at DR 3
[179185]rxTimeOut
```

- **Downlink Commands:**

A9 aa --> Same as AT+MBFUN=aa

3.3.6 RS485 command timeout

Some Modbus device has slow action to send replies. This command is used to configure the RS485-LB to use longer time to wait for their action.

Default value: 0, range: 0 ~ 10 seconds

- **AT Command:**

AT+CMDDLaa=hex(bb cc)

Example:

AT+CMDDL1=1000 to send the open time to 1000ms

- **Downlink Payload:**

0x AA aa bb cc Same as: **AT+CMDDLaa=hex(bb cc)**

Example:

0xAA 01 03 E8 --> Same as **AT+CMDDL1=1000 ms**

3.3.7 Uplink payload mode

Define to use one uplink or multiple uplinks for the sampling.

The use of this command please see: Compose Uplink payload

- **AT Command:**

AT+DATAUP=0

AT+DATAUP=1

- **Downlink Payload:**

0xAD 00 **-->** Same as AT+DATAUP=0

0xAD 01 **-->** Same as AT+DATAUP=1 //Each uplink is sent to the server one after the other as it is segmented.

- **AT Command:**

AT+DATAUP=1,Timeout

- **Downlink Payload:**

0xAD 01 00 00 14 **-->** Same as AT+DATAUP=1, 20000 //(00 00 14 is 20 seconds)

Each uplink is sent to the server at 20-second intervals when segmented.

3.3.8 Clear RS485 Command

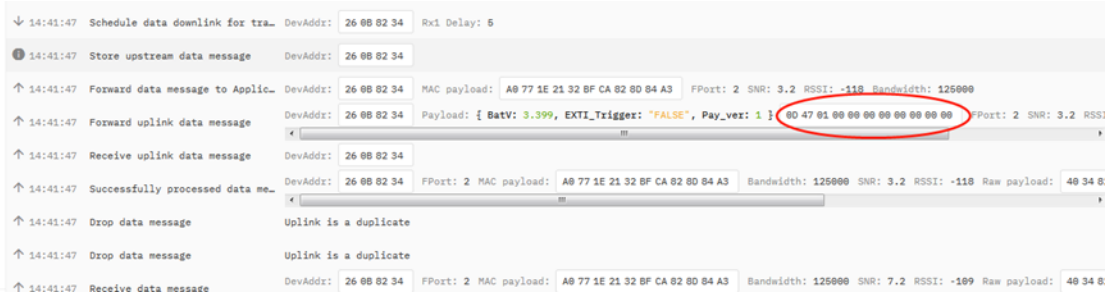
The AT+COMMANDx and AT+DATACUTx AT+SEARCHx and AT+CMDDLx settings are stored in special location, user can use below command to clear them.

- **AT Command:**

AT+CMDEAR=mm,nn mm: start position of erase ,nn: stop position of erase Etc. AT+CMDEAR=1,10 means erase AT+COMMAND1/AT+DATACUT1/AT+SEARCH1/AT+CMDDL1 to AT+COMMANDA/AT+DATACUTA/AT+SEARCHA/AT+CMDDLA.

Example screen shot after clear all RS485 commands.

The uplink screen shot is:



- **Downlink Payload:**

0x09 aa bb same as AT+CMDEAR=aa,bb

3.3.9 Set Serial Communication Parameters

Set the Rs485 serial communication parameters:

- **AT Command:**

- **Set Baud Rate:**

AT+BAUDR=9600 // Options: (200~115200), When using low baud rate or receiving multiple bytes, you need to use AT+CMDDL to increase the receive timeout (the default receive timeout is 300ms), otherwise data will be lost.

- **Set UART Parity**

AT+PARITY=0 // Option: 0: no parity, 1: odd parity, 2: even parity

- **Set STOPBIT**

AT+STOPBIT=1 // Option:1 for 1 bit ; 2 for 2 bits

- **Set DATABIT**

AT+DATABIT=8 // Option:7 for 7 bits ; 8 for 8 bits

- **Downlink Payload:**

Example:

A7 01 00 60 same as AT+BAUDR=9600

A7 01 04 80 same as AT+BAUDR=115200

A7 02 aa: Same as AT+PARITY=aa (aa value: 00 , 01 or 02)

A7 03 aa: Same as AT+STOPBIT=aa (aa value: 01 or 02)

A7 04 07: Same as AT+DATABIT=7

A7 04 08: Same as AT+DATABIT=8

3.3.10 Cut data separation processing

AT+NEWLINE command, which only takes effect when AT+DATAUP=1 or AT+DATAUP=1, timeout.

When not set, each part of AT+DATAUP is sent according to the maximum number of bytes of DR.

When setting, each part of AT+DATAUP is sent according to the value set by AT+NEWLINE.

- **AT Command:**

AT+NEWLINE=ALL The data cut out by each AT+COMMANDx command is sent separately as an uplink.

AT+NEWLINE=ALL *equal#* ***AT+NEWLINE=1+2+3+4+5+6+7+8+9+10+11+12+13+14+15***

AT+NEWLINE=a+b+c The data returned by all commands is divided into three parts, COMMAND(1~a) is the first part, COMMAND(a+1~b) is the second part,COMMAND(b+1~c) is the third part.

AT+NEWLINE=NULL Turn off the functionality of this AT command.

- **Downlink Payload:**

AT+NEWLINE=ALL ---> ***0xA5 01***

AT+NEWLINE= NULL ---> ***0xA5 00***

AT+NEWLINE= a+b+c ---> ***0xA5 number of bytes a b c***

AT+NEWLINE= 1+5+15 ---> ***0xA5 03 01 05 0F***

3.3.11 Control output power duration

User can set the output power duration before each sampling.

- **AT Command:**

Example:

AT+3V3T=1000 // 3V3 output power will open 1s before each sampling.

AT+3V3T=0 // Normally open 3V3 power supply.

AT+3V3T=65535 // Normally closed 3V3 power supply.

AT+5VT=1000 // +5V output power will open 1s before each sampling.

AT+5VT=0 // Normally closed +5V power supply.

AT+5VT=65535 // Normally open +5V power supply.

- **LoRaWAN Downlink Command:**

07 01 aa bb : Same as AT+5VT=(aa bb)

07 02 aa bb : Same as AT+3V3T=(aa bb)

07 03 01 : Same as AT+3V3T=0

07 03 00 : Same as AT+3V3T=65535

07 04 01 : Same as AT+5VT=65535

07 04 00 : Same as AT+5VT=0

3.3.12 Encrypted payload

- **AT Command:**

AT+DECRYPT=1 // The payload is uploaded without encryption

AT+DECRYPT=0 // Encrypt when uploading payload (default)

3.3.13 Get sensor value

- **AT Command:**

AT+GETSENSORVALUE=0 // The serial port gets the reading of the current sensor

AT+GETSENSORVALUE=1 // The serial port gets the current sensor reading and uploads it.

3.3.14 Resets the downlink packet count

- **AT Command:**

AT+DISFCNTCHECK=0 // When the downlink packet count sent by the server is less than the node downlink packet count or exceeds 16384, the node will no longer receive downlink packets (default)

AT+DISFCNTCHECK=1 // When the downlink packet count sent by the server is less than the node downlink packet count or exceeds 16384, the node resets the downlink packet count and keeps it consistent with the server downlink packet count.

3.3.15 When the limit bytes are exceeded, upload in batches

- **AT Command:**

AT+DISMACANS=0 // When the MACANS of the reply server plus the payload exceeds the maximum number of bytes of 11 bytes (DR0 of US915, DR2 of AS923, DR2 of AU195), the node will send a packet with a payload of 00 and a port of 4. (default)

AT+DISMACANS=1 // When the MACANS of the reply server plus the payload exceeds the maximum number of bytes of the DR, the node will ignore the MACANS and not reply, and only upload the payload part.

- **Downlink Payload**

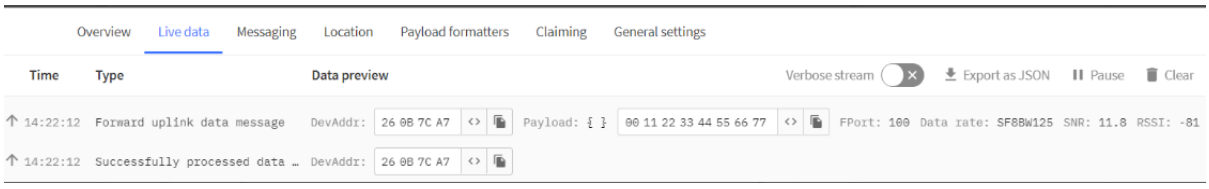
0x21 00 01 // Set the DISMACANS=1

3.3.16 Copy downlink to uplink

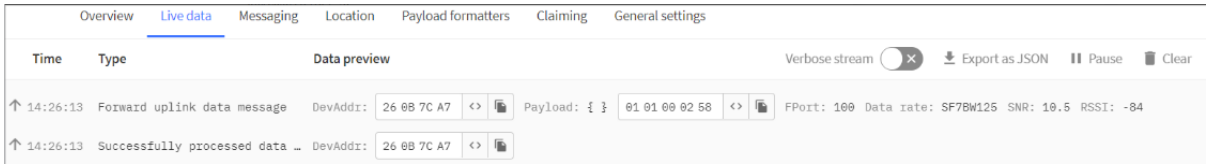
- **AT Command:**

AT+RPL=5 // After receiving the package from the server, it will immediately upload the content of the package to the server, the port number is 100.

Example: **aa xx xx xx xx** // aa indicates whether the configuration has changed, 00 is yes, 01 is no; xx xx xx xx are the bytes sent.



For example, sending 11 22 33 44 55 66 77 will return invalid configuration 00 11 22 33 44 55 66 77.



For example, if 01 00 02 58 is issued, a valid configuration of 01 01 00 02 58 will be returned.

3.3.17 Query version number and frequency band

- **Downlink Payload: 26 01** // Downlink 26 01 can query device upload frequency, frequency band, software version number, battery.

Example:

3.4 +3V3 Output

RS485-LB/LS has a Controllable +3V3 output, user can use this output to power external sensor.

The +3V3 output will be valid for every sampling. RS485-LB/LS will enable +3V3 output before all sampling and disable the +3V3 after all sampling.

The +3V3 output time can be controlled by AT Command.

AT+3V3T=1000

Means set +3v3 valid time to have 1000ms. So, the real +3v3 output will actually have 1000ms + sampling time for other sensors.

By default, the AT+3V3T=0. This is a special case, means the +3V3 output is always on at any time

3.5 +5V Output

RS485-LB/LS has a Controllable +5V output, user can use this output to power external sensor.

The +5V output will be valid for every sampling. RS485-LB/LS will enable +5V output before all sampling and disable the +5v after all sampling.

The 5V output time can be controlled by AT Command.

AT+5VT=1000

Means set 5V valid time to have 1000ms. So, the real 5V output will actually have 1000ms + sampling time for other sensors.

By default, the AT+5VT=0. If the external sensor which require 5v and require more time to get stable state, user can use this command to increase the power ON duration for this sensor.

3.6 Switch Jumper

Switch Jumper	Feature
SW1	ISP position: Upgrade firmware via UART Flash position: Configure device, check running status.
SW2	5V position: set to compatible with 5v I/O. 3.3v position: set to compatible with 3.3v I/O.,

+3.3V: is always ON

+5V: Only open before every sampling. The time is by default, it is **AT+5VT=0**.

Note: If SW2 is at +5V and AT+5VT=0, work mode 2 will not be able to send data.

3.7 Battery & Power Consumption

RS485-LB use ER26500 + SPC1520 battery pack and RS485-LS 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 .

4. Case Study

User can check this URL for some case studies: APP RS485 COMMUNICATE WITH SENSORS

5. OTA Firmware update

User can change firmware RS485-LB/LS to:

- Change Frequency band/ region.
- Update with new features.
- Fix bugs.

Firmware and changelog can be downloaded from : Firmware download link

Methods to Update Firmware:

- (Recommanded way) OTA firmware update via wireless: **http://wiki.dragino.com/xwiki/bin/view/Main/Firmware%20OTA%20Update%20for%20Sensors/**
- Update through UART TTL interface: **Instruction**.

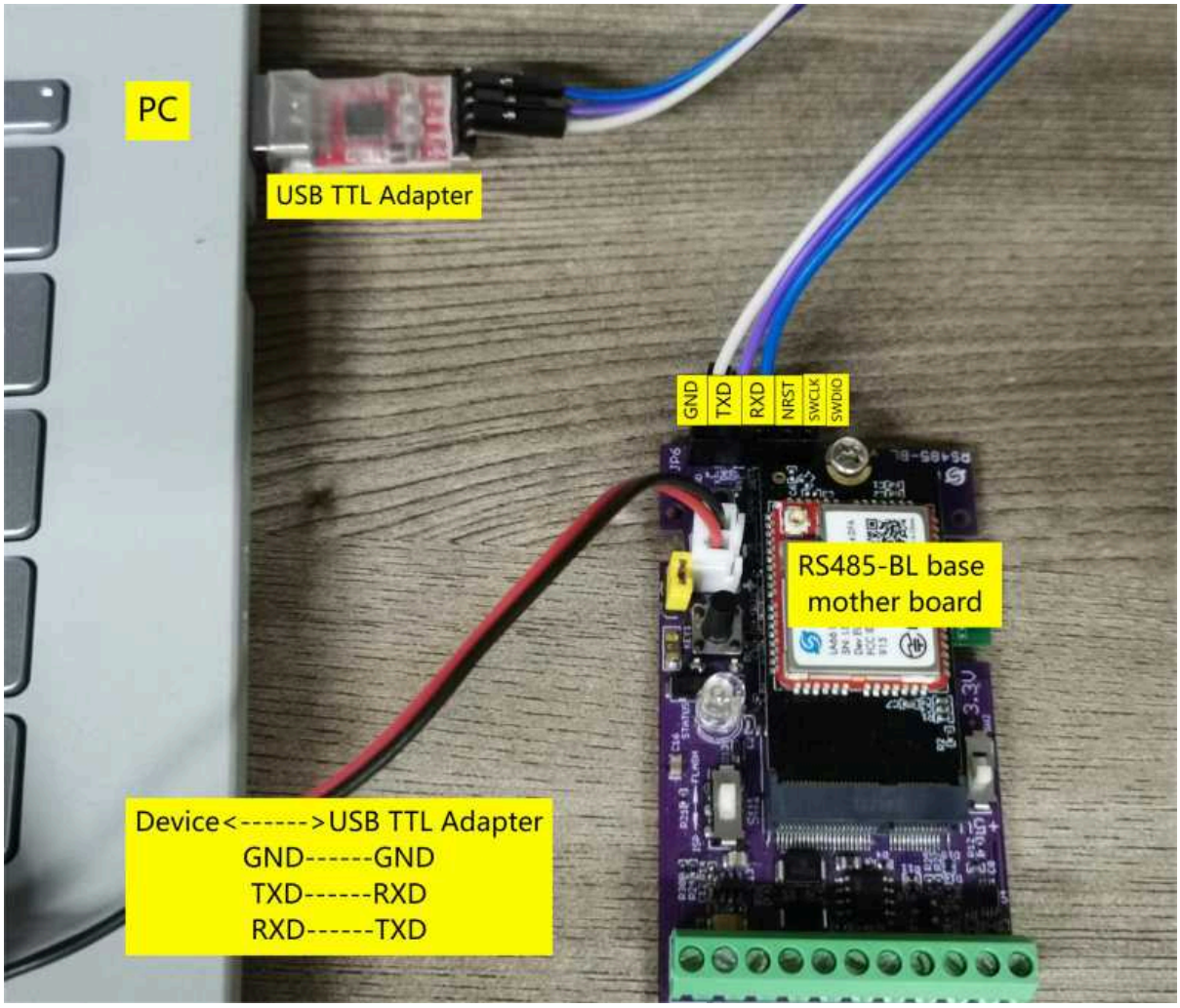
6. FAQ

6.1 How to upgrade the image?

The RS485-LB/LS LoRaWAN Controller is shipped with a 3.5mm cable, the cable is used to upload image to RS485-LB/LS to:

- Support new features
- For bug fix
- Change LoRaWAN bands.

Below shows the hardware connection for how to upload an image to RS485-LB/LS:



Update through UART TTL interface: **Instruction**.

6.2 How to change the LoRa Frequency Bands/Region?

User can follow the introduction for how to upgrade image. When download the images, choose the required image file for download.

6.3 How many RS485-Slave can RS485-LB/LS connects?

The RS485-LB/LS can support max 32 RS485 devices. Each uplink command of RS485-LB/LS can support max 16 different RS485 command. So RS485-LB/LS can support max 16 RS485 devices pre-program in the device for uplink. For other devices no pre-program, user can use the downlink message (type code 0xA8) to poll their info.

6.4 How to Use RS485-LB/LS to connect to RS232 devices?

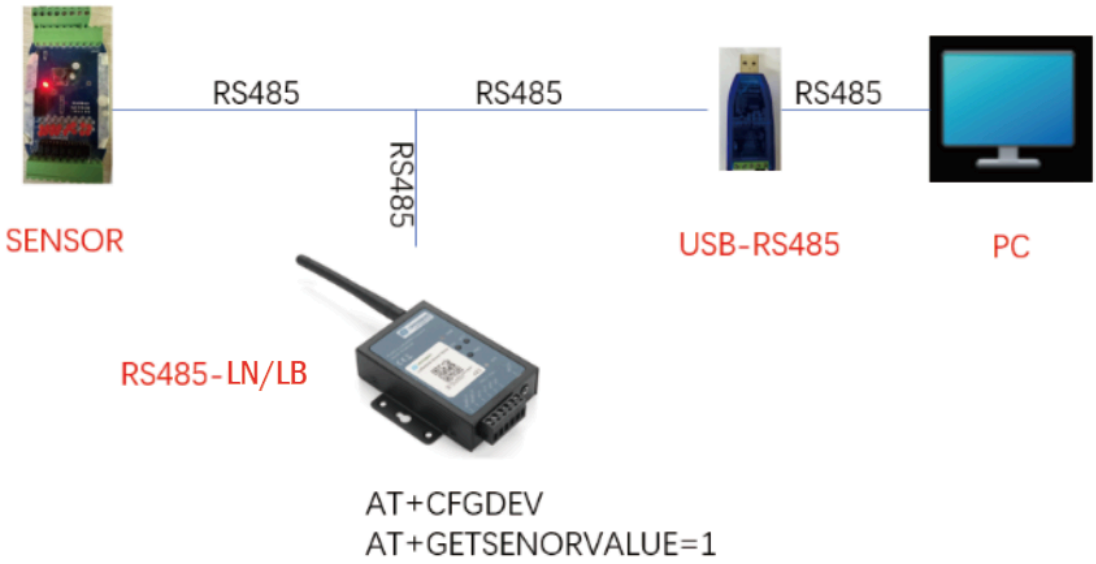
Use RS485-LB/LS or RS485-LN to connect to RS232 devices. - DRAGINO

6.5 How to judge whether there is a problem with the set COMMAND

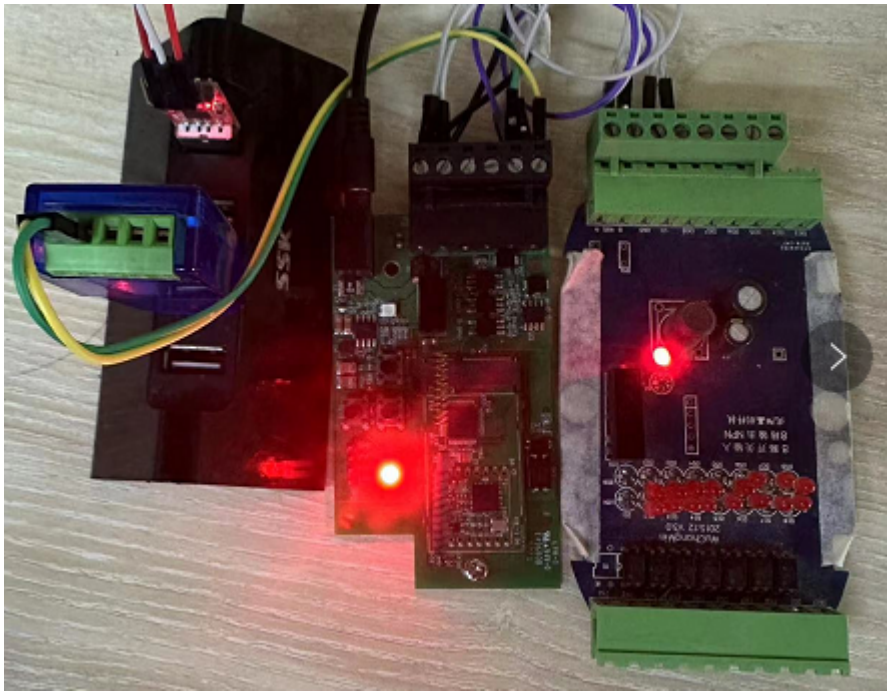
6.5.1 Introduce:

Users can use below the structure to fast debug the communication between RS485-LB/LS and RS485-LN. The principle is to put the PC in the RS485 network and sniff the packet between Modbus MTU and RS485-LB/LS/LN.
We can **use this way to:**

1. Test if Modbus-MTU works with PC commands.
2. Check if RS485-LN sent the expected command to Mobus-MTU
3. Check if Modbus-MTU return back the expected result to RS485-LN.
4. If both b) and c) has issue, we can compare PC's output and RS485-LN output.

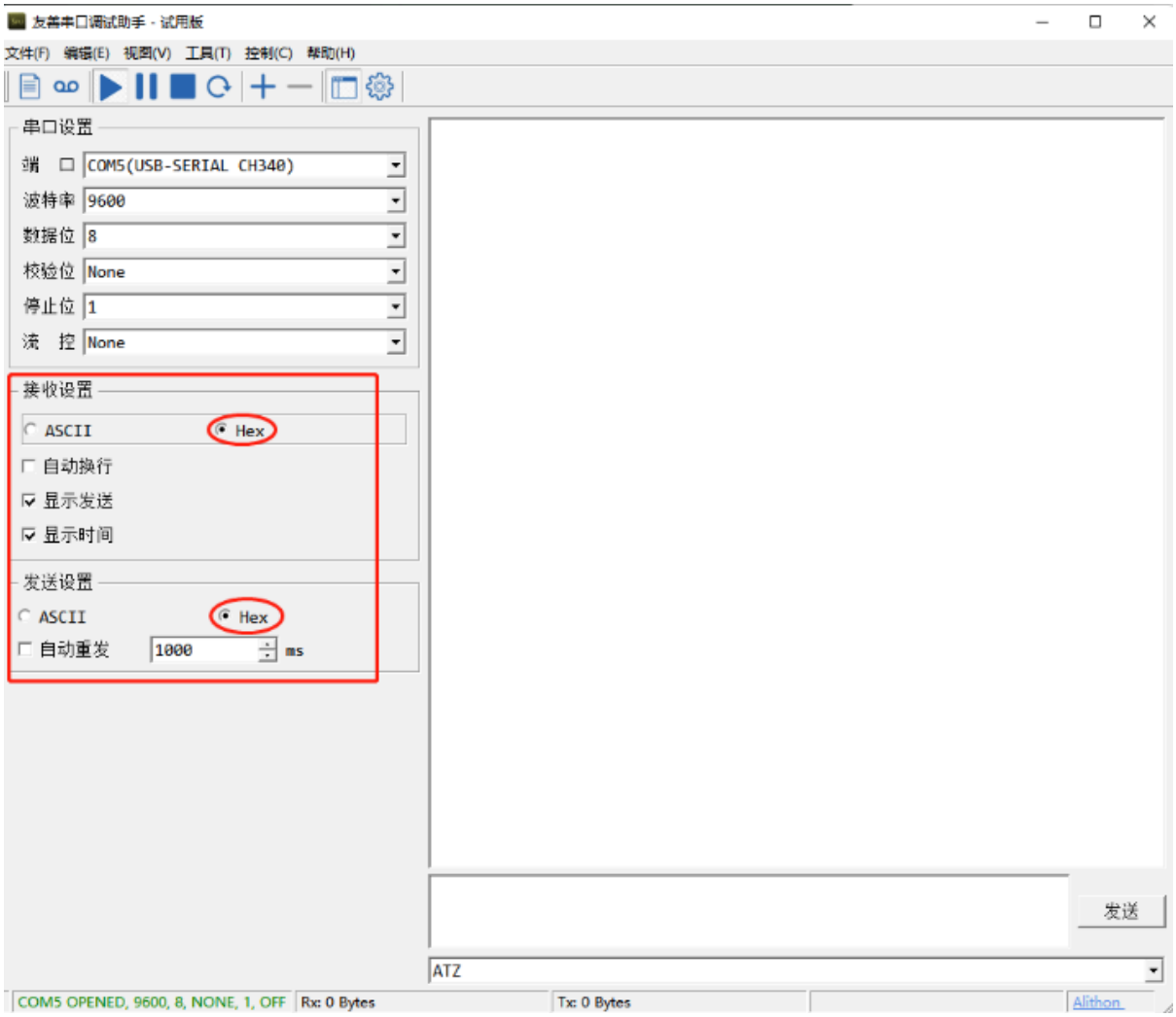


Example Connection:



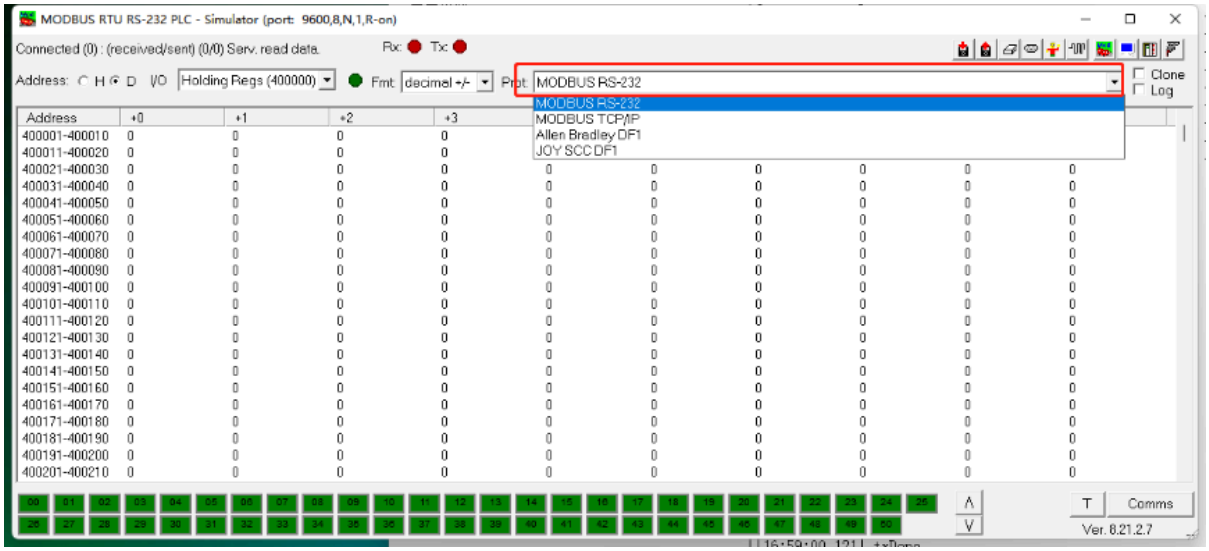
6.5.2 Set up PC to monitor RS485 network With Serial tool

Note: Receive and send set to hex mode

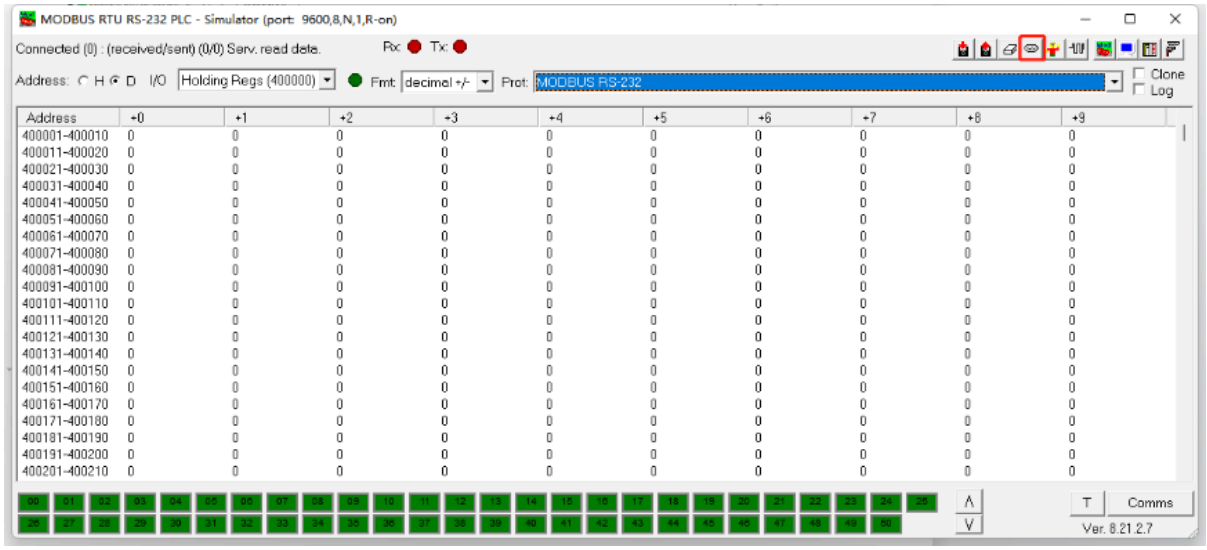


6.5.3 With ModRSsim2:

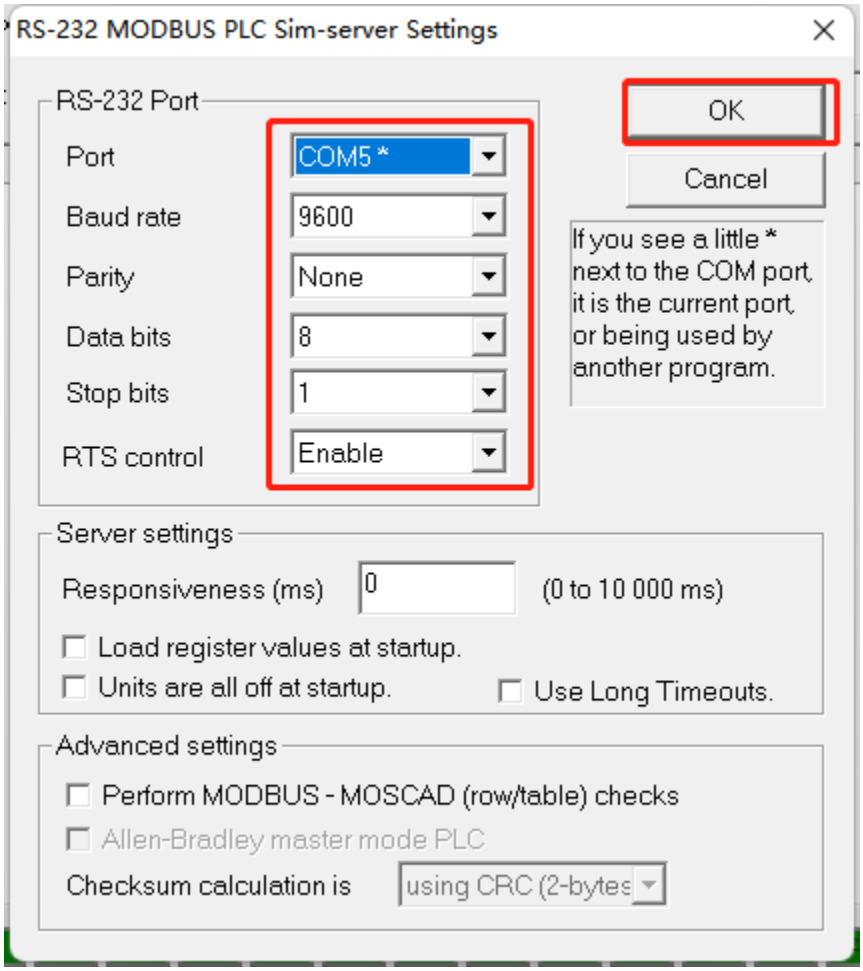
(1) Select serial port MODBUS RS-232



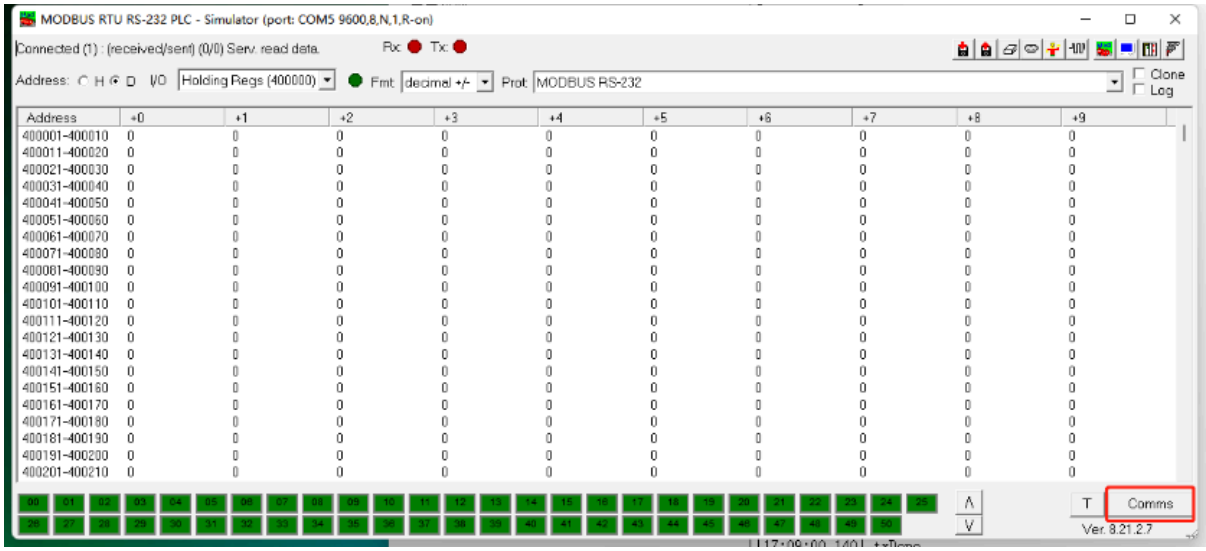
(2) Click the serial port icon



(3) After selecting the correct serial port and baud rate, click ok



(4) Click the comms.



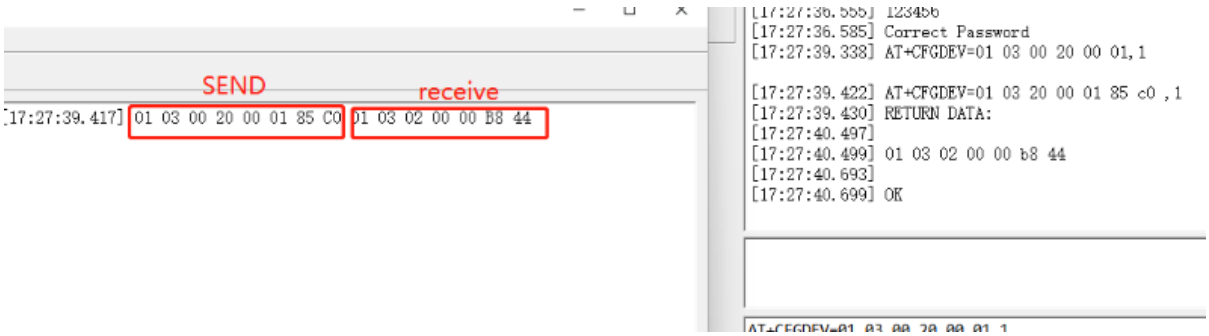
Run RS485-LN/BL command and monitor if it is correct.

6.5.4 Example – Test the CFGDEV command

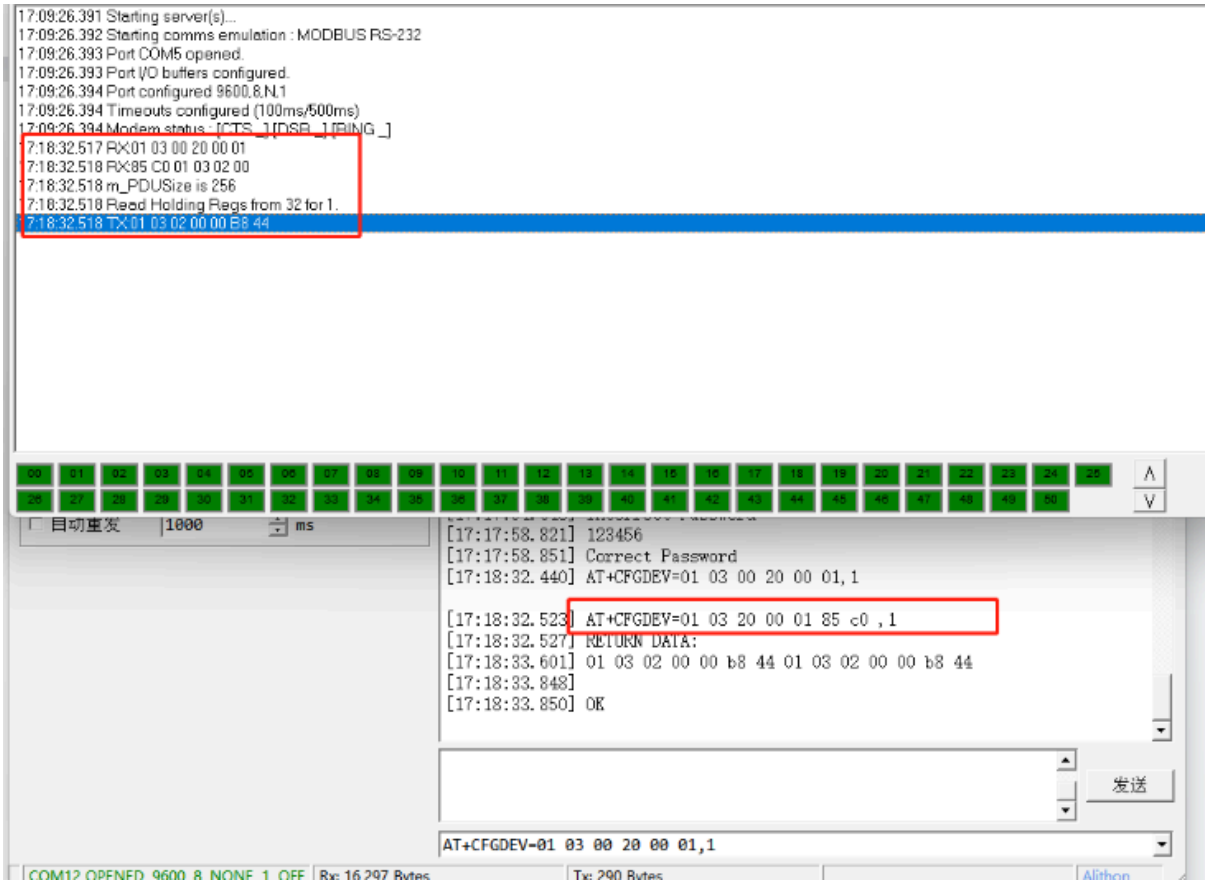
RS485-LN sent below command:

AT+CFGDEV=01 03 20 00 01 85 c0,1 to RS485 network, and PC is able to get this command and return commands from MTU to show in the serial tool.

We can see the output from the Serial port tool to analyze. And check if they are expected result.



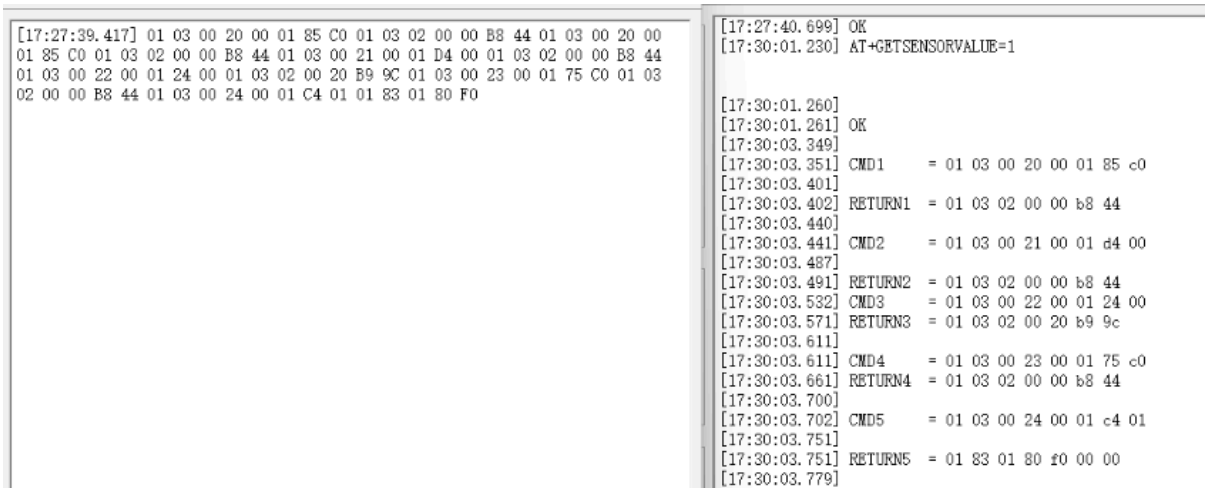
We can also use ModRSsim2 to see the output.



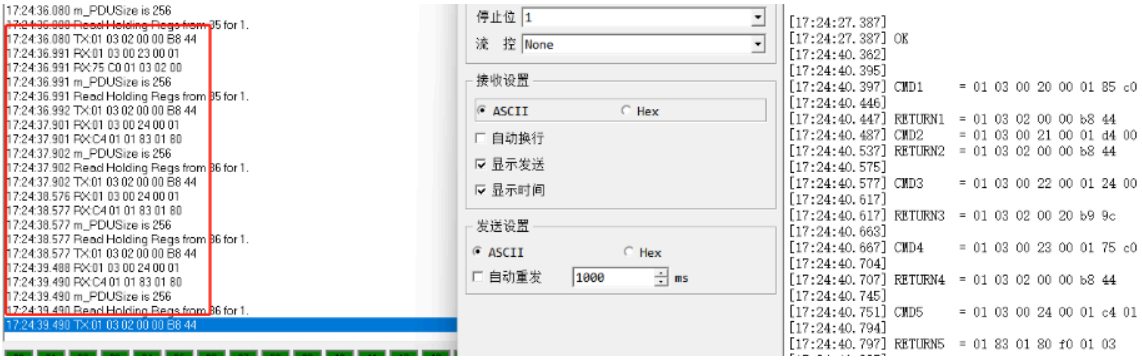
6.5.5 Example – Test CMD command sets.

Run **AT+SENSORVALUE=1** to test the CMD commands set in RS485-LN.

Serial port tool:



ModRSsim2:

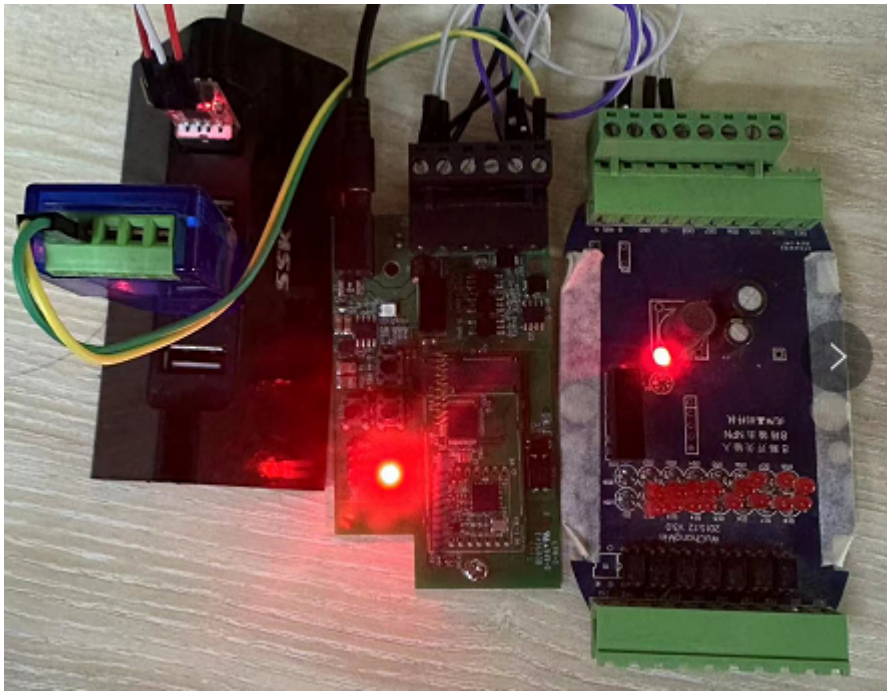


6.5.6 Test with PC

If there is still have problem to set up correctly the commands between RS485-LN and MTU. User can test the correct RS485 command set in PC and compare with the RS485 command sent out via RS485-LN. as long as both commands are the same, the MTU should return correct result.

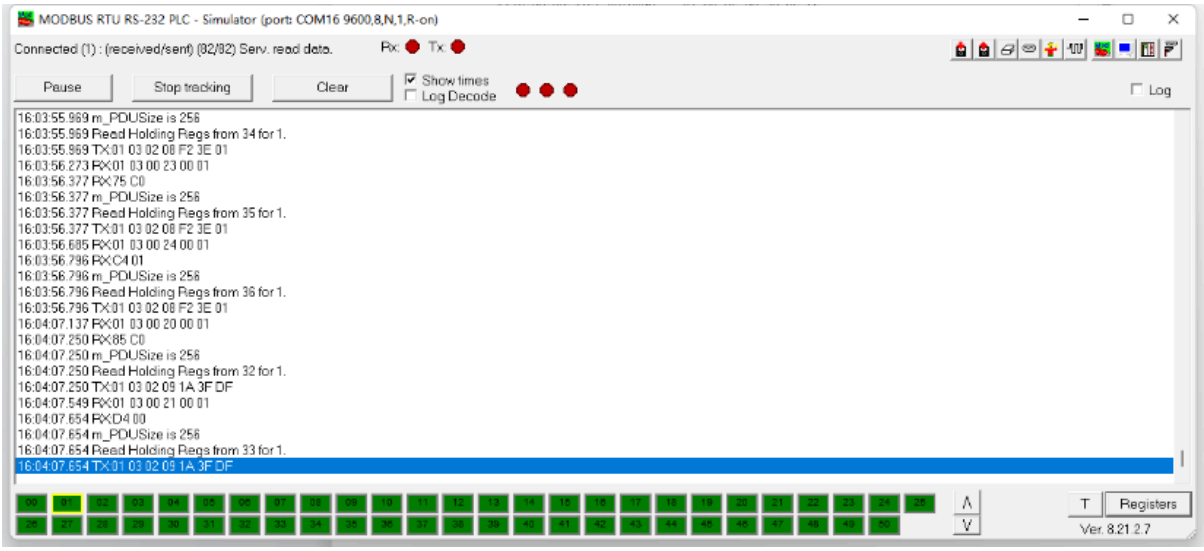
Or User can send the working commands set in PC serial tool to Dragino Support to check what should be configured in RS485-LN.

Connection method:



Link situation:

```
[16:03:35.274] CMD4     = 01 03 00 23 00 01 75 c0
[16:03:35.320] RETURN4  = 01 03 02 08 98 be 2e
[16:03:35.367] CMD5     = 01 03 00 24 00 01 c4 01
[16:03:35.412] RETURN5  = 01 03 02 08 98 be 2e
[16:03:35.444] Payload  = 01
[16:03:35.951]
[16:03:35.961] [134251]***** UpLinkCounter= 11 *****
[16:03:35.992] [134252]TX on freq 910700000 Hz at DR 0
[16:03:36.245] [134545]RX on freq 923300000 Hz at DR 8
[16:03:36.287] [134547]txDone
[16:03:41.239] [139539]RX on freq 924500000 Hz at DR 10
[16:03:41.281] [139571]RX on freq 923300000 Hz at DR 8
[16:03:41.311] [139573]rxTimeOut
[16:03:46.016]
[16:03:46.019] CMD1     = 01 03 00 20 00 01 85 c0
[16:03:46.065] RETURN1  = 01 03 02 08 c0 bf d4 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00
[16:03:46.204] CMD2     = 01 03 00 21 00 01 d4 00
[16:03:46.252] RETURN2  = 01 03 02 08 c0 bf d4
[16:03:46.297] CMD3     = 01 03 00 22 00 01 24 00
[16:03:46.344] RETURN3  = 01 03 02 08 c0 bf d4
[16:03:46.374] CMD4     = 01 03 00 23 00 01 75 c0
[16:03:46.421] RETURN4  = 01 03 02 08 c0 bf d4
[16:03:46.467] CMD5     = 01 03 00 24 00 01 c4 01
[16:03:46.515] RETURN5  = 01 03 02 08 c0 bf d4
[16:03:46.545] Payload  = 01
[16:03:47.052]
[16:03:47.057] [145352]***** UpLinkCounter= 12 *****
[16:03:47.088] [145354]TX on freq 910500000 Hz at DR 0
[16:03:47.347] [145646]RX on freq 923300000 Hz at DR 8
[16:03:47.383] [145648]txDone
[16:03:52.340] [150639]RX on freq 923900000 Hz at DR 10
[16:03:52.376] [150671]RX on freq 923300000 Hz at DR 8
[16:03:52.407] [150673]rxTimeOut
```



6.6 Where to get the decoder for RS485-LB/LS?

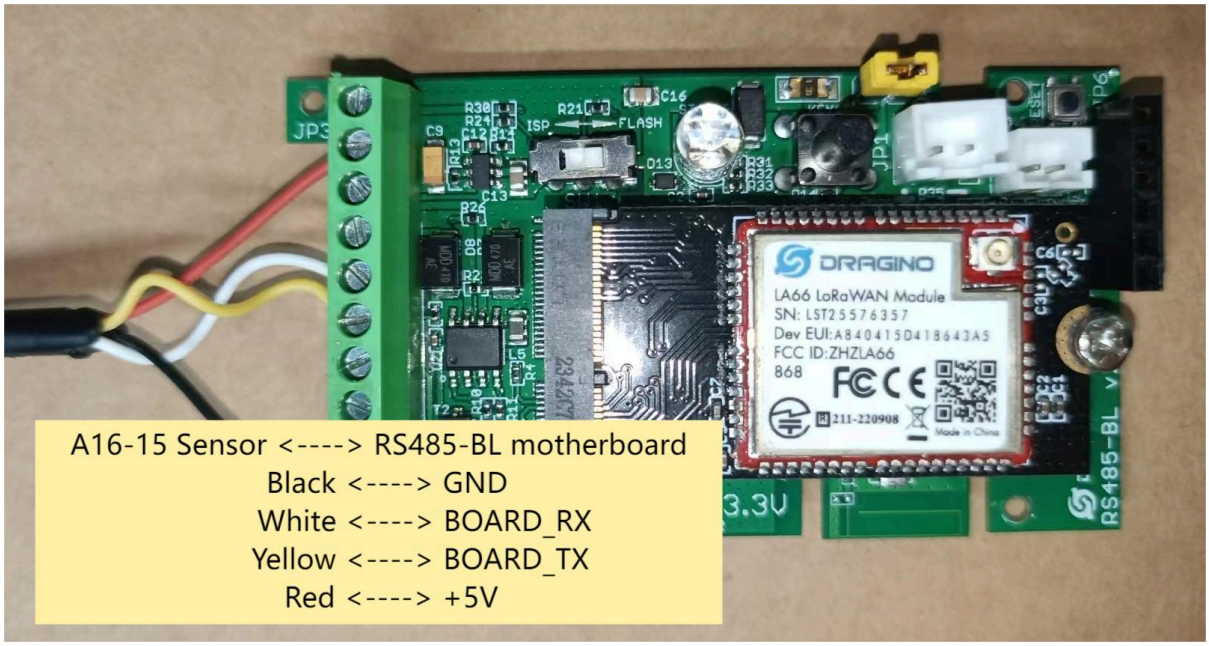
The decoder for RS485-LB/LS needs to be written by yourself. Because the sensor to which the user is connected is custom, the read device data bytes also need custom parsing, so there is no universal decoder. We can only provide templates for decoders (no intermediate data parsing part involved)

6.7 How to connect RS485-LB node to UART distance sensor?

Take the A16-15 as an example, but the same setup can support different distance sensors, compare specifications see this link to DDS04-LB.

A16-15		Detect Distance: 50cm ~ 1500cm Bling Spot Distance: 0 ~ 50cm Accuracy: ±(1cm+S*0.3%) (S: Distance) Measure Angle: ~ 40° Cable Length: 1.5 meter Temperature Compensation Suitable for Long Distance Detect IP67 Water Proof

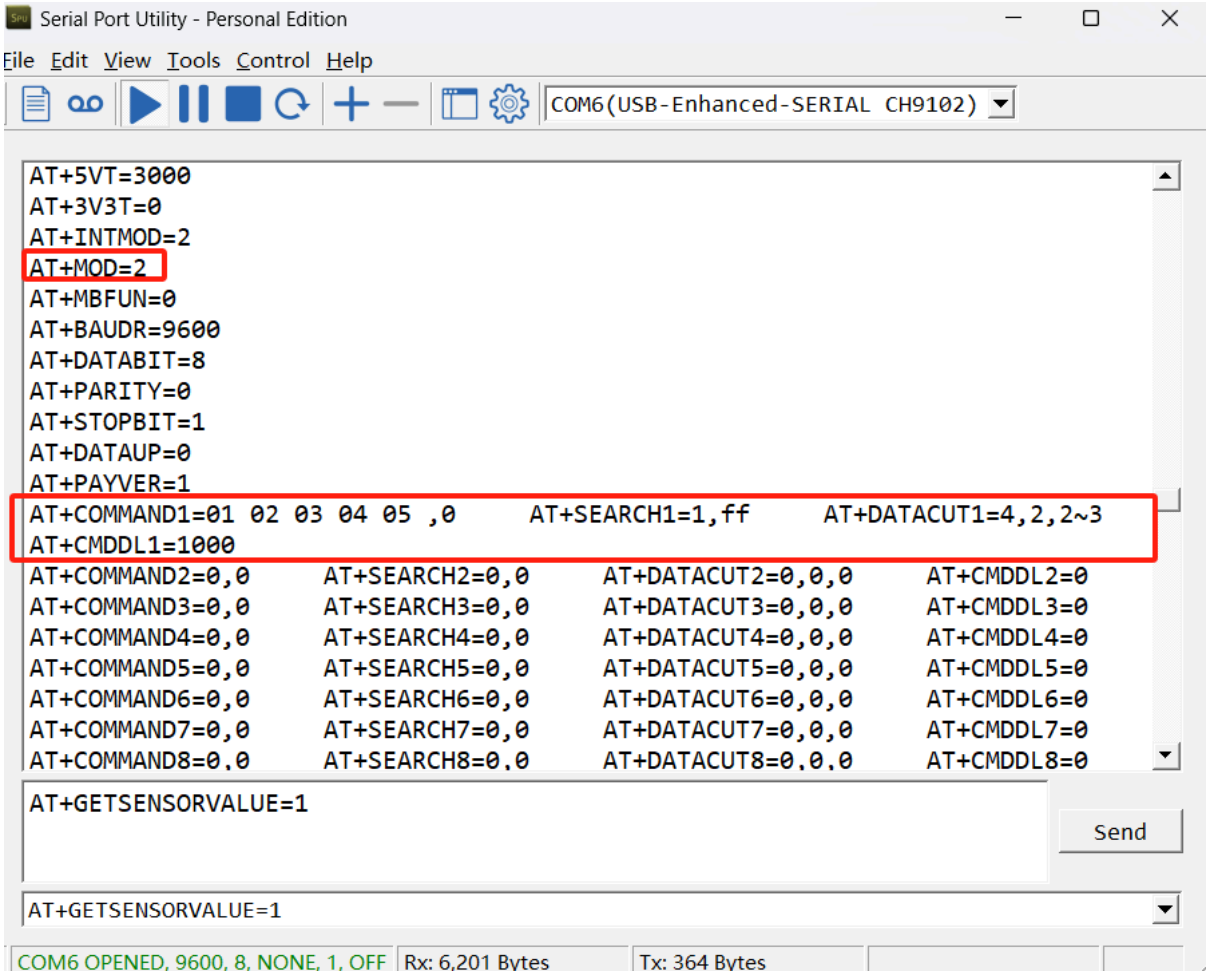
- Connection:



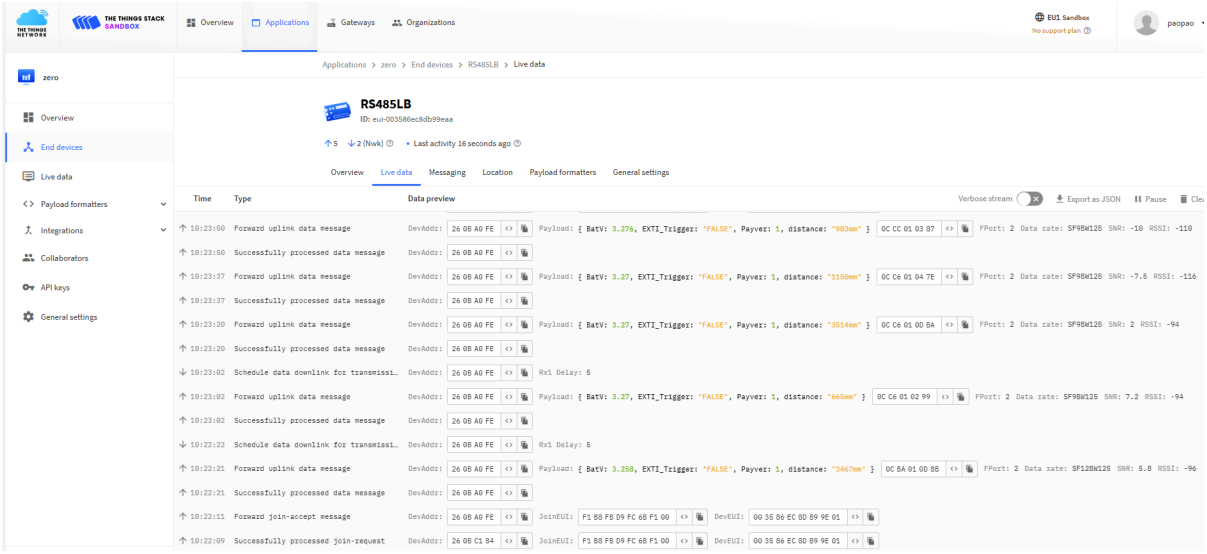
- **AT command configuration:**

AT+MOD=2

AT+COMMAND1=01 02 03 04 05 ,0 AT+SEARCH1=1,ff AT+DATACUT1=4,2,2~3 AT +CMDDL1=1000



- **Decoder: Decoding refer to this link.**
- **Example in TTN:**



7. Trouble Shooting

7.1 Downlink doesn't work, how to solve it?

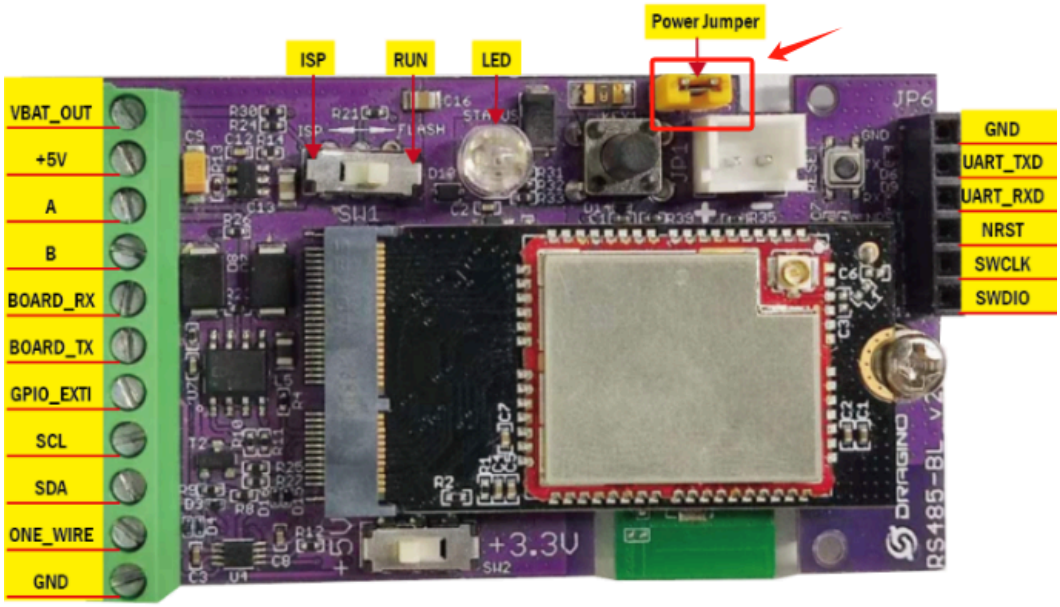
Please see this link for debug: LoRaWAN Communication Debug

7.2 Why I can't join TTN V3 in US915 /AU915 bands?

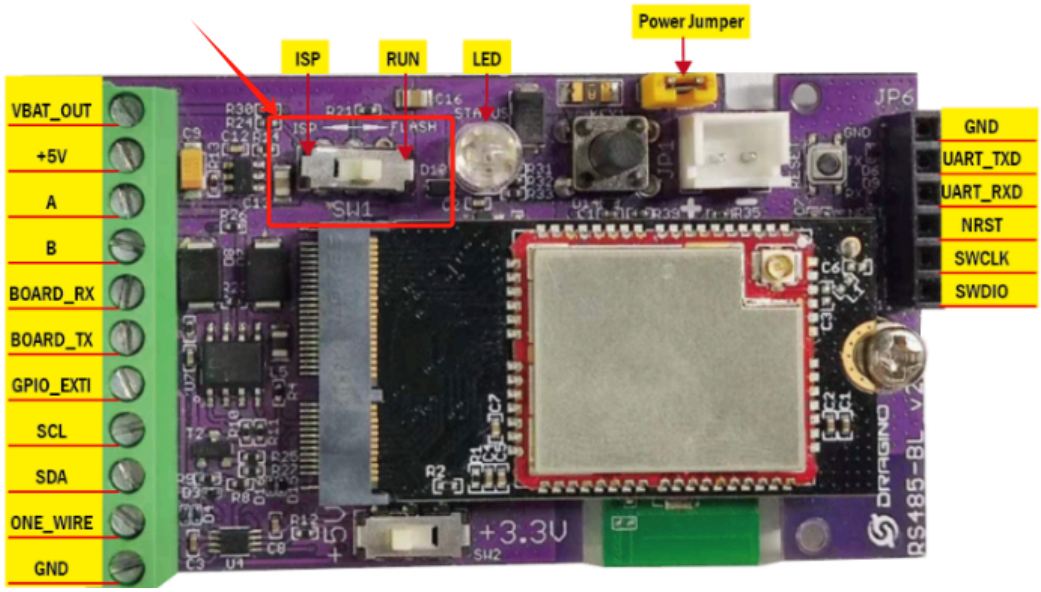
It might about the channels mapping. Please see for detail: Notice of Frequency band

7.3 Possible reasons why the device is unresponsive:

1. Check whether the battery voltage is lower than 2.8V
2. Check whether the jumper of the device is correctly connected



3. Check whether the switch here of the device is at the ISP (The switch can operate normally only when it is in RUN)



7.4 Why can't customers see the device's data in the server when the data is too long?

This is due to the limitation of the lorawan protocol, and the fixed DR needs to be adjusted to improve this problem.

Please refer to the following link for the number of bytes limited by different frequencies and different DRs in the lorawan protocol

lora-alliance.org/wp-content/uploads/2021/05/RP002-1.0.3-FINAL-1.pdf

Example:

2.4.6 EU863-870 Maximum payload size

The maximum **MACPayload** size length (*M*) is given by the following table. It is derived from limitation of the PHY layer depending on the effective modulation rate used taking into account a possible repeater encapsulation layer. The maximum application payload length in the absence of the OPTIONAL **FOpts** control field (*N*) is also given for information only. The value of N MAY be smaller if the **FOpts** field is not empty:

Data Rate	*M*	*N*
0	59	51
1	59	51
2	59	51
3	123	115
4	230	222
5	230	222
6	230	222
7	230	222
8	58	50
9	123	115
10	58	50
11	123	115
12:15	Not defined	

Table 12: EU863-870 maximum payload size (repeater compatible)

Please refer to the following command to fix DR

AT+ADR=0

AT+DR=3

Downlink command:

http://wiki.dragino.com/xwiki/bin/view/Main/End%20Device%20AT%20Commands%20and%20Downlink%20Command/#H7.4DataRate

7.5 How to solve the problem that the sensor requires a pull-up resistor on the RS485A pin?

1. Upgrade the firmware to a dedicated firmware, which will consume more power and cannot enter low power mode.

If necessary, please contact our technical support email:

support@dragino.com

2. Use AT command:

AT+RS485POWER=1

8. Order Info

Part Number: RS485-LB-XX-YY or RS485-LS-XX-YY

XX:

- **EU433**: frequency bands EU433
- **EU868**: frequency bands EU868
- **KR920**: frequency bands KR920
- **CN470**: frequency bands CN470
- **AS923**: frequency bands AS923
- **AU915**: frequency bands AU915
- **US915**: frequency bands US915
- **IN865**: frequency bands IN865
- **RU864**: frequency bands RU864
- **KZ865**: frequency bands KZ865

YY: Hole Option

12: With M12 waterproof cable hole
16: With M16 waterproof cable hole

9. Packing Info

Package Includes:

- RS485-LB or RS485-LS x 1
- Stick Antenna for LoRa RF part x 1
- Program cable x 1

Dimension and weight:

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

10. 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