RS485-NB -- RS485/UART to NB-IoT Converter User Manual

last modified by Mengting Qiu

on 2024/08/26 16:31

Table of Contents

1. Introduction	. 5
1.1 What is RS485 / UART to NB-IoT Converter	. 5
1.2 Features	. 6
1.3 Specification	. 7
1.4 Applications	. 7
1.5 Sleep mode and working mode	. 8
1.6 Button & LEDs	. 8
1.7 BLE connection	. 8
1.8 Pin Definitions . Switch & SIM Direction	. 9
1.81 Jumper JP2	. 9
1.8.2 BOOT MODE / SW1	. 9
183 Beset Button	. 9
1.84 SIM Card Direction	10
1.8.5 SW2 Jumper (Define UABT level to external Sensor)	10
2 Use BS485-NB to communicate with IoT Server	10
2.1 Send data to lot server via NB-IoT network	10
2.2 Configure Device to Bead BS485 Sensors	11
2.2.1 Method 1 via BS485 Configure Tool	11
2.2.2 Method 2 via AT Commands	12
2.2.3 Configure senses	15
2.2.4 Configure read commands for each sampling	15
2.2.4 Unlink when the concert is not reaching	10
2.2.5 Unlink when the sensor is not responding	10
2.2.0 Opinik on interlupt	20
2.3 Fayload Types	20
2.3.1 General JSOIT FORMal (Type=5)	20
2.3.2 MEX IOIIIIal Payload (Type=0)	21
2.5 Fost command to bondle MODDLy device	22
	23
2.6 R5465 command timeout	23
2.7 Clear RS485 Command	23
2.8 Set Payload Version	23
2.9 Control output power duration	24
2.10 Get sensor value	24
2.11 +3V3 Output	24
2.12 +5V Output	24
2.13 Switch Jumper	25
2.14 Clock logging (Since firmware version v1.0.7)	25
2.15 Example Query saved historical records	26
2.16 Uplink log query	27
2.17 Scheduled domain name resolution	28
2.18 Set the QoS level	28
3. Configure RS485-NB	29
3.1 Configure Methods	29
3.2 Serial Access Password	29
3.3 AT Commands Set	31
4. Battery & Power Consumption	32
5. Firmware update	33
6. FAQ	33
6.1 How can I access t BC660K-GL AT Commands?	33
6.2 How many RS485-Slave can RS485-NB connects?	33
6.3 How to configure the device through the MQTT subscription function?	33
6.4 How to Use RS485-NB to connect to RS232 devices?	34
6.5 How to judge whether there is a problem with the set COMMAND	34

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual

7.	6.5.1 Introduce: 6.5.2 Set up PC to monitor RS485 network With Serial tool 6.5.3 With ModRSsim2: 6.5.4 Example – Test the CFGDEV command 6.5.5 Example – Test CMD command sets. 6.5.6 Test with PC Order Info Packing Info	34 35 36 39 40 41 43 43
0. 9.	Support	43 44



Table of Contents:

• <u>1. Introduction</u>

- 1.1 What is RS485 / UART to NB-IoT Converter
- <u>1.2 Features</u>
- 1.3 Specification
- <u>1.4 Applications</u>
- <u>1.5 Sleep mode and working mode</u>
- 1.6 Button & LEDs
- 1.7 BLE connection
- <u>1.8 Pin Definitions</u>, Switch & SIM Direction
 - <u>1.8.1 Jumper JP2</u>
 - <u>1.8.2 BOOT MODE / SW1</u>
 - 1.8.3 Reset Button
 - 1.8.4 SIM Card Direction
 - 1.8.5 SW2 Jumper (Define UART level to external Sensor)

- 2. Use RS485-NB to communicate with IoT Server
 - 2.1 Send data to IoT server via NB-IoT network
 - 2.2 Configure Device to Read RS485 Sensors
 - 2.2.1 Method 1 -- via RS485 Configure Tool
 - 2.2.2 Method 2 -- via AT Commands
 - 2.2.2.1 Configure UART settings for RS485 or TTL communication
 - <u>2.2.3 Configure sensors</u>
 - 2.2.4 Configure read commands for each sampling
 - 2.2.5 Uplink when the sensor is not responding
 - 2.2.6 Uplink on Interrupt
 - <u>2.3 Payload Types</u>
 - <u>2.3.1 General Json Format(Type=5)</u>
 - 2.3.2 HEX format Payload(Type=0)
 - 2.4 Test Uplink and Change Update Interval
 - 2.5 Fast command to handle MODBUS device
 - 2.6 RS485 command timeout
 - <u>2.7 Clear RS485 Command</u>
 - <u>2.8 Set Payload version</u>
 - <u>2.9 Control output power duration</u>
 - 2.10 Get sensor value
 - <u>2.11 +3V3 Output</u>
 - <u>2.12 +5V Output</u>
 - 2.13 Switch Jumper
 - 2.14 Clock logging (Since firmware version v1.0.7)
 - <u>2.15 Example Query saved historical records</u>
 - <u>2.16 Uplink log query</u>
 - <u>2.17 Scheduled domain name resolution</u>
 - 2.18 Set the QoS level
- <u>3. Configure RS485-NB</u>
 - <u>3.1 Configure Methods</u>
 - <u>3.2 Serial Access Password</u>
 - 3.3 AT Commands Set
- <u>4. Battery & Power Consumption</u>
- <u>5. Firmware update</u>
- 6. FAQ
 - 6.1 How can Laccess t BC660K-GL AT Commands?
 - 6.2 How many RS485-Slave can RS485-NB connects?
 - 6.3 How to configure the device through the MQTT subscription function?
 - 6.4 How to Use RS485-NB to connect to RS232 devices?
 - 6.5 How to judge whether there is a problem with the set COMMAND
 - 6.5.1 Introduce:
 - 6.5.2 Set up PC to monitor RS485 network With Serial tool
 - 6.5.3 With ModRSsim2:
 - 6.5.4 Example Test the CFGDEV command
 - 6.5.5 Example Test CMD command sets.
 - 6.5.6 Test with PC
- 7. Order Info
- 8. Packing Info
- 9. Support

1. Introduction

1.1 What is RS485 / UART to NB-IoT Converter

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

RS485-NB can interface to RS485 sensor, 3.3v/5v UART sensor or interrupt sensor. RS485-NB 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-NB supports different uplink methods including **MQTT**, **MQTTs**, **UDP** & **TCP** for different application requirement, and support uplinks to various IoT Servers.

RS485-NB supports **BLE configure and OTA update** which make user easy to use.

RS485-NB is IP67 waterproof and powered by 8500mAh Li-SOCI2 battery, it is designed for long-term use up to several years.

RS485-NB has optional built-in SIM card and default IoT server connection version. Which makes it works with simple configuration.

RS485-NB in a NB-IoT Network



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
- · Support multiply RS485 devices by flexible rules
- Support Modbus protocol
- Support Interrupt uplink
- Supports connecting a UART sensors with 3.3V or 5V
- Multiply Sampling and one uplink
- Support Bluetooth v5.1 remote configure and update firmware
- Uplink via MQTT, MQTTs, TCP, or UDP
- Uplink on periodically
- · Downlink to change configure
- 8500mAh Battery for long term use
- Nano SIM card slot for NB-IoT SIM

1.3 Specification

Common DC Characteristics:

- Supply Voltage: 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/OUT pins
- 1 x I2C Interface
- 1 x one wire interface

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
 B12 @H-FDD: 740MHz
- B13 @H-FDD: 740MHz
 B17 @H-FDD: 730MHz
- B18 @H-FDD: 730MHz
 B18 @H-FDD: 870MHz
- B19 @H-FDD: 870MHz
- B10 @H-FDD: 070MHz
 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

- 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 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



Behavior on ACT	Function	Action
Pressing ACT between 1s < time < 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.
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 attach NB-IoT 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 attach NB-IoT network or not.
Fast press ACT 5 times.	Deactivate Device	Red led will solid on for 5 seconds. Means device is in Deep Sleep Mode.

Note: When the device is executing a program, the buttons may become invalid. It is best to press the buttons after the device has completed the program execution.

1.7 BLE connection

RS485-NB 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

RS485-NB use the mother board which as below.



1.8.1 Jumper JP2

Power on Device when put 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.

1.8.5 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

2. Use RS485-NB to communicate with IoT Server

2.1 Send data to IoT server via NB-IoT network

The RS485-NB is equipped with a NB-IoT module, the pre-loaded firmware in RS485-NB 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 RS485-NB.

Below shows the network structure:



There are two version: -GE and -1D version of RS485-NB.

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 RS485-NB 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

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual



1D Version: This version has 1NCE SIM card pre-installed and configure to send value to DataCake. User Just need to select the sensor type in DataCake and Activate RS485-NB and user will be able to see data in DataCake. See here for <u>DataCake Config Instruction</u>.

2.2 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-NB supports flexible command set. User can use <u>Dragino RS485 Tool</u>, <u>AT Commands or LoRaWAN Downlink</u> Command to configure how RS485-NB should read the sensor and how to handle the return from RS485 or TTL sensors.

2.2.1 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

😻 DRAGNO-RS485-Corring VILD		-	- n ×
nda constit 1 (3):"RS465 Commands IN.SERSOL	Ewenik Ontoni	bet v (1): General sets searce: son v srotaren) fement seid pet fiormet seid pet	Confiure Sever Realists
		15405 Sease rates data	O forica Xiseba
Foll Constit	2 : Console Window		ingani firmatili-17,4-16 Mgi tar 2 georgin
			O Diserets lipto:
Roll Commanded			ispet firmett3.5.9 Mg to 8 lytes
Foll Command=1			
foll Constit	dT Carnaed		Channe the massive discrete batter drive rates
full Connel+S	Saul		Serena Consead Provine
Fall Caraci-10		(4): Process Sensor	Returns
Poll Communicati	5 : Commands Preview		
Poll Commandmid			
Foll Companies			
Fall Canaci-11			
Fall Canaci-15			
Cast 82005 Company			AC+CROBLODS405 Sensor response live Unit-re-
Clear All	Report Conversels Save Conversion States Mill		Ferencie AL instruction

2.2.2 Method 2 -- via AT Commands

2.2.2.1 Configure UART settings for RS485 or TTL communication

RS485-NB 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-NB 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-NB setting to match.

AT Commands	Description	Example
AT+BAUDR	Set the baud rate.	AT+BAUDR=9600
	Default Value is: 9600.	Options: (1200,2400,4800,14400,19200,115200)
AT+PARITY	Set UART parity.	AT+PARITY=0
	Default Value is: no parity.	Option: 0: no parity, 1: odd parity, 2: even parity
AT+STOPBIT	Set serial stopbit	AT+STOPBIT=0 for 1 bit
	Default Value is: 1bit.	AT+STOPBIT=1 for 1.5 bits
		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+PAYVER=1				
AT+MBFUN=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	
AT+COMMANDD=0,0	AT+SEARCHD=0,0	AT+DATACUTD=0,0,0	AT+CMDDLD=0	
AT+COMMANDE=0,0	AT+SEARCHE=0,0	AT+DATACUTE=0,0,0	AT+CMDDLE=0	
AT+COMMANDF=0,0	AT+SEARCHF=0,0	AT+DATACUTF=0,0,0	AT+CMDDLF=0	

ОК

upload payload:

[41270]*****Upload start:0*****

CMD1 = fe 03 00 00 00 03 11 c4 RETURN1 = fe 03 06 00 00 09 49 00 00 b6 cb Payload = 01 00 00 09 49 00 00

[41928]BAT:3.584 V [45458]Open a Socket Service successfully [52516]Datagram is sent by RF [53550]Send complete

2.2.3 Configure sensors

Some sensors might need to configure before normal operation. User can configure such sensor via PC or through RS485-NB 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
	AT+CFGDEV=xx xx	
	mm: 0: no CRC, 1: add CRC-16/MODBUS in the end of this command	

2.2.4 Configure read commands for each sampling

Note: Please upgrade firmware to v1.0.3 or above. Otherwise, if a data error occurs in AT+COMMANDx check crc mode, the device will enter an inoperable state.

RS485-NB 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-NB can support 15 commands to read sensors. And combine the return to one or several uplink payloads.

Command from RS485-NB to Sensor:

RS485-NB 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-NB:

After RS485-NB send out a string to sensor, RS485-NB 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

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:

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-NB, 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 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 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

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+COMMAND2=0,0
AT+DATACUT2=0,0,0
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
ALLCOMMANDD-0,0
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		- 1
RETURN1	-	20	20	20	20	2d	30	2e	32	20 7	5
Payload	=	0c	fc	01	20	20	30	32	20	20	

· Grab a section.

AT+PAYVER=1
AT+COMMAND1=01 03 0b b8 00 02 ,1 AT+DATACUT1=8,2,4~8
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+COMMAND6=0.0 AT+DATACUT6=0.0.0
AT+COMMAND7=0.0 AT+DATACUT7=0.0.0
AT+COMMAND8=0.0 AT+DATACUT8=0.0.0
AT+CC AT, DATA CUT4-0.2.4.0
AT+CC AT+DATAGUTT=8,2,4~8
AT+c(a=8 return total 8 bytes (20 20 20 20 2d 30 2e 00)
ATLC D=Z
$\Delta T + C$ c=4~8 (grap the 4th ~ 8th bytes from return, so command 1 valid value is 20 2d 30 2e 00
ATTCOMPAND -0,0 ATTCATACOTF-0,0,0

ΟK

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

Grab different sections.

```
AT+COMMAND1=01 03 0b b8 00 02 ,1
AT+COMMAND2=0,0 AT+DATACUT2=0,0,0 AT+DATACUT2=0,0,0
AT+COMMAND2=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+DATACUT6=0 0 0
AT+DATACUT1=13,2,1~2+4~7+10~11
 a=13, return total 13 bytes (90 02 6a 82 1a 04 20 2d 30 2e dd 9b 00)
 b=2
 c=1~2+4~7+10~11 (grap the 1 ~ 2 bytes + 4~7 bytes + 10~11 bytes
so command1 valid value is 90 02 82 1a 04 20 2e dd
AITCOMMANDE-V,V
                     AITDAIACUIE-0,0,
                     AT+DATACUTF=0,0,0
AT+COMMANDF=0,0
AT+CHS=0
OK
```

CMD1	=	01	03	0b	b8	00	02	46	0a						
RETURN1	=	90	02	6a	82	1a	04	20	2d	30	<u>2e</u>	dd	9b	00	
Payload	-	0c	fc	01	90	02	82	1a	04	20	2e	dd			

Note:

AT+SEARCHx and AT+DATACUTx can be used together, if both commands are set, RS485-NB 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

2.2.5 Uplink when the sensor is not responding

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.



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.

```
[172119]Signal Strength:23
[175053]*****Upload start:5*****
CMD1 = fe 03 00 00 00 03 11 c4
RETURN1 = NULL
Payload = 01 00 00 00 00 00 00
[176587]BAT:3.591 V
[183623]Datagram is sent by RF
[184658]Send complete
[185683]*****End of upload*****
```

2.2.6 Uplink on Interrupt

Put the interrupt sensor between 3.3v_out and GPIO ext.



AT+INTMOD=0 Disable Interrupt.(Default Value)

- **AT+INTMOD=1** Interrupt trigger by rising or falling edge.
- AT+INTMOD=2 Interrupt trigger by falling edge.
- AT+INTMOD=3 Interrupt trigger by rising edge.

2.3 Payload Types

To meet different server requirement, RS485-NB supports different payload type.

Includes:

- General JSON format payload. (Type=5)
- <u>HEX format Payload</u>. (Type=0)

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

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

2.3.1 General Json Format(Type=5)

This is the General Json Format. As below:

```
{"IMEI":"866207053462705","Model":"RS485-
NB","Payload":01111122223333,"battery":3.581,"signal":23}
```

Notice:The maximum number of bytes in the Payload is 1 payload version number plus 687 RS485 data.

AT+CLOCKLOG=1,65535,1,5:

{"IMEI":"863663062798815","Model":"RS485-NB","Payload":"010da5","battery":3.295,"signal":23,"1": ["010da9","2024/07/15 02:45:09"],"2":["010da4","2024/07/15 02:44:09"],"3":["010da6","2024/07/15 02:43:09"],"4":["010da4","2024/07/15 02:42:09"],"5":["010da3","2024/07/15 02:41:18"]}



2.3.2 HEX format Payload(Type=0)

This is the HEX Format. As below:

f863663062798815 176e 0cdd 1a 00 00 66948e7b 01 0da4

HEX Format for RS485-NB									
f863663062798815	176e	0cdd	1a	00	00	66948e7b	01	0da4	
f+IMEI	Version	BAT	Singal	GPIO_EXIT Level	GPIO_EXIT Flag	Timestamp	Payload_version	RS485_sensor_data	
8 Bytes					12+n Byte	es(The maximu	ım n value is 710)		

Version:

These bytes include the hardware and software version.

Higher byte: Specify Sensor Model: 0x17 for RS485-NB

Lower byte: Specify the software version: 0x6e=110, means firmware version 1.1.0

BAT (Battery Info):

Ex1: 0x0cdd = 3293mV

Signal Strength:

NB-IoT Network signal Strength.

Ex1: 0x1a = 26

- 0 -113dBm or less
- 1 -111dBm
- 2...30 -109dBm... -53dBm
- 31 -51dBm or greater
- 99 Not known or not detectable

GPIO_EXIT Level :

GPIO_EXTI is used as Interrupt Pin.

Example:

01 (H): GPIO_EXTI pin is high level.

00 (L): GPIO_EXTI pin is low level.

GPIO_EXIT Flag:

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

Note: The Interrupt Pin is a separate pin in the screw terminal.

Example:

0x00: Normal uplink packet.

0x01: Interrupt Uplink Packet.

TimeStamp:

Unit TimeStamp Example: 66948E7B(H) = 1721011835(D)

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

Payload_version:

The payload version number is used to parse different decodes.

RS485_sensor_data:

The data returned by the RS485 sensor or UART sensor needs to be cut out the AT+DATACUTx and AT +MBFUN or AT+SEARCHx commands.

2.4 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+TDC=7200 // Set Update Interval to 7200s

User can also push the button for more than 1 seconds to activate an uplink.

2.5 Fast command to handle MODBUS device

The command is for fast configure to read Modbus devices. It is only valid for the devices which follow the MODBUS-RTU protocol.

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.

2.6 RS485 command timeout

Some Modbus device has slow action to send replies. This command is used to configure the RS485-NB 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

2.7 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.

2.8 Set Payload version

This is the first byte of the uplink payload. RS485-NB 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

2.9 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.

2.10 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.

2.11 +3V3 Output

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

The +3V3 output will be valid for every sampling. RS485-NB 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

2.12 +5V Output

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

The +5V output will be valid for every sampling. RS485-NB 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.

2.13 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.

2.14 Clock logging (Since firmware version v1.0.7)

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 loging 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).

🌚 MQTT.fx - 1.7.1	-	- 0	×
File Extras Help			
normal mqtt 🔹	Connect Disconnect		-
Publish Subscribe Scripts Broker Sta	tus Log		
RS485_NB_PUB	Subscribe Qo Qo Qo	Autoscroll	○ \$▼
R\$485_NB_PUB 15	RS485 NB PUB		QoS 0
Dump Messages Mute Onsubscribe			QoS 0
	RS485_NB_PUB		12 QoS 0
	RS485_NB_PUB		13 QoS 0
	RS485_NB_PUB		14 QoS 0
	RS485_NB_PUB		15 QoS 0
	RS485_NB_PUB		
Topics Collector (0) Scan Stop 😋	15-07-2024 10:46:08.38768100		QoS 0
	<pre>{"IMEI":"863663062798815","Model":"RS485-NB","Payload": ery":3.295,"signal":23,"1":["010da9","2024/07/15 02:45:09"],"2":["010da4","2024/07/15 02:44:09"],"3":["010da6","2024/07/15 02:43:09"],"4":["010da4","2024/07/15 02:42:09"],"5":["010da3","2024/07/15 02:41:18"]}</pre>	"010da5",	"batt
	Payload decoded by Plain Text Dec	oder	•

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: 0x0A

Format: Command Code (0x0A) followed by 5 bytes.

- Example 1: Downlink Payload: 0A01FFFF0F08 +CLOCKLOG=1,65535,15,8
- Example 1: Downlink Payload: 0A0104B00F08 +CLOCKLOG=1,1200,15,8

// Set SHT record time: AT

// Set SHT record time: AT

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

2.15 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.

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual

Serial Port Utility - Personal Edition -		×
<u>Eile Edit View T</u> ools <u>C</u> ontrol <u>H</u> elp		
Serial Port Setting Port COM6(USE Baudrate 9600 Baudrate 9600 Data Bits 8 Parity None Stop Bits 1 Flow Type None Receive Setting @10DA4, 2024/07/15 02:35:09] @10DA6, 2024/07/15 02:35:09] @10DA6, 2024/07/15 02:37:09] @10DA7, 2024/07/15 02:38:09] @10DA7, 2024/07/15 02:38:09] @10DA7, 2024/07/15 02:39:09] @10DA7, 2024/07/15 02:40:09] @10DA7, 2024/07/15 02:40:715 02:40:09] @10DA6, 2024/07/15 02:40:09] @10DA7, 2024/07/15 02:40:715 02:40:20] @10DA7, 2024/07/15 02:40:20] @10DA5, 2024/07/15 02:40:20] @10DA5, 2024/07/15 02:51:14] </td <td>Se</td> <td>▲ ■ nd</td>	Se	▲ ■ nd
AI+CDP		•
COM6 OPENED, 9600, 8, NONE, 1, OFF Rx: 26,640 Bytes Tx: 1,597 Bytes		

2.16 Uplink log query

• AT command: AT+GETLOG

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

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual

Serial Port Utility - Personal Edition	
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>T</u> ools <u>C</u> ontrol <u>H</u> elp	
	〇〇 (学)
Serial Port Setting	[Tx][19:18:51.683] AT+GETLOG
Port COM2(Silicon Laby	[Rx][19:18:51.704]
	[Rx][19:18:51.704] Exit sleep mode
Baudrate 9600 💌	[Rx][19:18:51.724] Signal Strength:99 *1
Data Bits 8	[Rx][19:18:51.745] Signal Strength:18
	[Rx][19:18:51.764] *****Upload start:8*****
Parity None 💌	[Rx][19:18:51.794] Open a Socket Service successfully
Stop Bits 1	[Rx][19:18:51.835] Upload data successfully
Flow Type None 💌	[[KX][19:18:51.865] Close the port successfully
	[[KX][19:18:51.895] Send complete
Receive Setting	[[XX][19:18:51.905] *****End of upload*****
© Text C Hex	[[K]][19:10:51.955]
- Text	[AX][19:10:51.955] EXIL STEEP MODE
Auto Feed Line	[[A,][17,10,31,734] 31gHd 31(Helod start)0*****
✓ Display Send	[RX][17:10:31.75] Open a Sockat Service successfully
	[Av][19.10.52.005] Open a Socket Service Sockessfully
I✓ Display Time	[Av][19.10.52.465] Option the next successfully
and addition	[Av][19:18:52 A95] Close the port successfully
Send Setting	[NJ][19.18.52.135] Send Complete
□ Loop 1000 → ms	[Ry][19:18:52 144] Evit sleen mode
	[Rx][19:18:52 164] Signal Strength:16
Line by Line	
	[Rx][19:18:52.184] *****Upload start:10*****
	[Rx][19:18:52.204] Open a Socket Service successfully
	[Rx][19:18:52.245] Upload data successfully
	[Rx][19:18:52.275] Close the port successfully
	[Rx][19:18:52.305] Send complete
	[Rx][19:18:52.315] *****End of upload*****
	[Rx][19:18:52.345]
	linuitanan nani rutu
	AT+GETLOG

2.17 Scheduled domain name resolution

This command is used to set up scheduled domain name resolution.

AT command:

• AT+DNSTIMER // Unit: hour

After setting this command, domain name resolution will be performed regularly.

2.18 Set the QoS level

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

AT command:

- AT+MQOS // 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. Configure RS485-NB

3.1 Configure Methods

RS485-NB supports below configure method:

- AT Command via Bluetooth Connection (Recommended): <u>BLE Configure Instruction</u>.
- 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.

Serial Port Utility - Personal Edition	on	-		×
<u>File Edit View T</u> ools <u>C</u> ontrol	<u>H</u> elp			
📄 🚥 🕨 🚺 🔳 🗘	+- 🗔 🐵			
Serial Port Setting Port COM3(Sil	ок			•
Data Bits 8 Parity None	***** UpLinkCounter= 10 ***** TX on freq 867.100 MHz at DR 5 txDone			
Stop Bits 1 Flow Type None	RX on freq 867.100 MHz at DR 5			
Receive Setting	rxTimeout			
<pre> Text ○ Hex Auto Feed Line </pre>	RX on freq 869.525 MHz at DR 0 rxTimeout			
☑ Display Send ☑ Display Time	123456 Correct Password			
Send Setting				-
<pre> Text ○ Hex </pre>	123456		Sen	d
□ Loop 1000 ÷ ms				
□ Line by Line	123456			•
COM3 OPENED, 9600, 8, NONE,	1, OFF Rx: 1,604 Bytes Tx: 42 Bytes		_	

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

500 Serial Port Utility - Personal Editi	on		-		×
<u>File Edit View Tools Control</u>	Help				
	+- 🖻 🕸				
Serial Port Setting Port COM3(Sil • Baudrate 9600 •	***** UpLinkCounter= 1 TX on freq 867.900 MHz txDone	***** at DR 5			•
Data Bits 8	RX on freq 867.900 MHz	at DR 5			
Stop Bits 1 Flow Type None	Received: ADR Message				
Receive Setting	rxDone				
• Text C Hex	AT+PWRM2				
☐ Auto Feed Line ☞ Display Send □ Display Time	AT+PWORD=123ab/				
_ Send Setting					•
	AT+PWORD=123ab/			Sen	d
□ Line by Line	AT+PWORD=123ab/				•
COM3 OPENED, 9600, 8, NONE,	1. OFF Rx: 2.472 Bytes	Tx: 133 Bytes			

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.

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual

🔤 Serial Port Util	lity - Personal Edition —		\times
<u>F</u> ile <u>E</u> dit <u>V</u> iew	<u>I</u> ools <u>C</u> ontrol <u>H</u> elp		
📄 🛯 🕨			
-Serial Por	rt Setting		•
Port	Option ? >	<	
Baudrate	Port Receive Send Log Display		
Data Bits			
Parity			
Stop Bits	Auto add line feed when send (Only valid in text mode)		
Flow Type	Comment Support		
-Receive S	□ Format Send		
• Text	Line Ending Mode		
🗆 Auto Fe	\odot CR & LF (\r\n) $\ \bigcirc$ Line Feed (\n) $\ \bigcirc$ Carriage Return (\r) $\ \bigcirc$ LF & CR (\n\r)		
🔽 Display			
🗆 Display	OF Cancel Imply	-	
- Send Setta			-
• Text	C Hex AT+PWORD=123ab/		
□ Loop 100	00 <u>+</u> ms		send
□ Line by	Line AT+PWORD=123ab/		-

3.3 AT Commands Set

AT+ <cmd>?</cmd>	: Help on <cmd></cmd>

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

AT+<CMD>=? : Get the value

General Commands

AT	: Attention
AT?	: Short Help
ATZ	: MCU Reset
AT+TDC	: Application Data Transmission Interval
AT+CFG	: Print all configurations
AT+CFGMOD	: Working mode selection
AT+DEUI	: Get or set the Device ID
AT+PRO	: Choose agreement
AT+RXDL	: Extend the sending and receiving time
AT+DNSCFG	: Get or Set DNS Server

AT+GETSENSORVALUE : Returns the current sensor measurement
AT+SERVADDR : Server Address
AT+MOD: Get or Set work mode
AT+5VT : Get or Set extend the time of 5V power
AT+3V3T: Get or Set extend the time of 3V3 power
AT+INTMOD: Get or Set the trigger interrupt mode (0:input,1:falling or rising,2:falling,3:rising)
AT+BAUDR: Get or Set baudr of uart
AT+DATABIT: Get or Set databit(7:7 bit,8:8 bit) of uart
AT+PARITY: Get or Set parity(0:none,1:odd,2:even) of uart
AT+STOPBIT: Get or Set stopbit(0:1 bit,1:1.5 bit,2:2 bit) of uart
AT+CMDEAR: Erase command(number of begin to number of ending)
AT+PAYVER: Get or Set payload version
AT+MBFUN: Get or Set Modbus Funtion code(0:none,1:01 or 02,2:03 or 04) for automatic cut
AT+COMMAND: Send data of command1-15 to RS485 deceive for payload
AT+SEARCH: Retrieve the hexadecimal character that appears in command 1-15
AT+DATACUT: Cut receive data after use command1-15
AT+CMDDL: Set delay timeout after send rs485 command1-15
AT+CFGDEV: Send data to RS485 deceive
AT+DNSTIMER: Regularly resolve domain names

MQTT Management

AT+CLIENT	: Get or Set MQTT client
AT+UNAME	: Get or Set MQTT Username
AT+PWD	: Get or Set MQTT password
AT+PUBTOPIC	: Get or Set MQTT publish topic
AT+SUBTOPIC	: Get or Set MQTT subscription topic

Information

AT+FDR	: Factory Data Reset
AT+PWORD	: Serial Access Password
AT+LDATA	: Get the last upload data

4. Battery & Power Consumption

RS485-NB use ER26500 + SPC1520 battery pack. 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 many RS485-Slave can RS485-NB connects?

The RS485-NB can support max 32 RS485 devices. Each uplink command of RS485-NB can support max 16 different RS485 command. So RS485-NB 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.

6.3 How to configure the device through the MQTT subscription function?

Subscription content: {AT COMMAND}

Example:

Setting AT+5VT=500 through Node-RED requires MQTT to send the content {AT+5VT=500}.



The serial port displays:

CMD1 = fe 03 00 00 00 03 11 c4 RETURN1 = fe 03 06 11 11 22 22 33 33 65 54 Payload = 01 11 11 22 22 33 33

[42203]BAT:3.584 V [47629]Opened the MQTT client network successfully [51187]Successfully connected to the server AT+PWRM2 [54905]Upload data successfully [58443]Received downlink data:<u>AT+5VT=500</u> [59330]Subscribe to topic successfully [62877]Close the port successfully [63916]Send complete [64940]*****End of upload*****

6.4 How to Use RS485-NB to connect to RS232 devices?

Use RS485-NB 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-NB. The principle is to put the PC in the RS485 network and sniff the packet between Modbus MTU and RS485-NB. We can use this way to:

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



Example Connection:



6.5.2 Set up PC to monitor RS485 network With Serial tool

Note: Receive and send set to hex mode

■ 友善丰口调试助手 - 试用版	_		\times
文件(F) 编辑(E) 视問(V) 工具(T) 控制(C) 帮助(H)			
- 串口设置			
端 □ COM5(USB-SERIAL CH340) ▼			
波特率 9600 🔹			
数据位 8			
校验位 None 🔹			
停止位 1			
流 控 None -			
- 接收设置			
ASCII			
「自动换行			
▼ 显示发送			
☞ 显示时间			
- 发送设置			
C ASCII			
□ 自动重发 1000 ÷ ms			
	_		
		发ì	ž
AT7	-		-
COM5 OPENED, 9600, 8, NONE, 1, OFF Rx: 0 Bytes Tx: 0 Bytes	E	lithon_	_

6.5.3 With ModRSsim2:

(1) Select serial port MODBUS RS-232

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual

😹 MODBUS RTU	RS-232	PLC - Simulator (port: 1	9600,8,N,1,R-on)						_	D X
Connected (0) : (re	ceived/	sent) (0/0) Serv. read da	ata. R	x 🌒 Tx 🔴					🛔 🔒 🗗	🗢 者 M 👪	. 🖪 🖻
Address: C H 🕫	D VO	Holding Regs (40000	0) 💌 🕈 Fm	decimal +/- 💌	Prot. MODBUS	RS-232				•	Clone
Address	+0	+1	+2	+3	MODBUS MODBUS	RS-232 TCP/IP					
400001-400010	0	0	0	0	Allen Bra	lley DF1					
400011-400020	0	0	0	0	JOY SCC	DÉ1					
400021-400030	0	0	0	0	0	D	0	0	0	0	
400031-400040	0	0	0	0	0	0	0	0	0	0	
400041-400050	0	0	0	0	0	0	0	0	0	0	
400051-400060	0	0	0	0	0	0	0	0	0	0	
400061-400070	0	0	0	0	0	0	0	0	0	0	
400071-400080	0	0	0	0	0	0	0	0	0	0	
400081-400090	0	0	0	0	0	0	0	0	0	0	
400091-400100	0	0	0	0	0	0	0	0	0	0	
400101-400110	0	0	0	0	0	D	0	0	0	0	
400111-400120	0	0	0	0	0	D	0	0	0	0	
400121-400130	0	D	0	0	0	D	0	0	0	0	
400131-400140	0	0	0	0	0	D	0	0	0	0	
400141-400150	0	0	0	0	0	D	0	0	0	0	
400151-400160	0	0	0	0	0	D	0	0	0	0	
400161-400170	0	0	0	0	0	0	0	0	0	0	
400171-400180	0	0	0	0	0	0	0	0	0	0	
400181-400190	U	U	U	U	0	0	0	U	U	0	
400191-400200	U	U	U	U	U	U	0	U	U	0	
1400201-400210	U	U	U	U	U	U	0	U	U	U	
00 01 02	03	04 05 00 07	08 09 10	11 12 1	14 15	16 17 18	19 20 21	22 23 24 2		т	Comme
									17		00.1115
		32 33								Ver. 8	.21.2.7 .
							116.50.	00 1011 AuD-m-			

(2) Click the serial port icon

MODBUS RTU	J RS-232 PL	.C - Simulator (port:	9600,8,N,1,R-on)							- 0	×
Connected (0) : (re	aceived/se	nt) (0/0) Serv. read (ilata. Po	c 🔴 Tx: 🔴					🛔 🔒 🔗	◎ 🛃 🗤 📓 🗖	10 7
Address: C H @	D 1/0	Holding Regs (4000	000) 💌 🌒 Fmt	decimal +/- 💌	Prot: MODBUS	RS-232				•	🗆 Clone 🗆 Log
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	
400001-400010	0	0	0	0	0	0	0	0	0	0	
400011-400020	0	0	0	0	0	0	0	0	0	0	
400021-400030	0	0	0	0	D	0	0	0	0	0	
400031-400040	0	0	0	0	D	0	0	D	0	0	
400041-400050	0	0	0	0	D	0	0	0	0	0	
400051-400060	0	0	0	0	0	0	0	0	0	0	
400061-400070	0	0	0	0	0	0	0	0	0	0	
400071-400080	0	0	0	0	0	0	0	0	0	0	
400081-400090	0	0	0	0	0	0	0	0	0	0	
400091-400100	0	0	0	0	D	0	0	0	0	0	
400101-400110	0	0	0	0	D	0	0	0	0	0	
400111-400120	0	0	0	0	0	0	0	0	0	0	
400121-400130	0	0	0	0	0	0	0	0	0	0	
400131-400140	0	0	0	0	0	0	0	0	0	0	
400141-400150	0	0	0	0	0	0	0	0	0	0	
400151-400160	0	0	0	0	0	0	0	0	0	0	
400161-400170	0	0	0	0	D	0	0	0	0	0	
400171-400180	0	0	0	0	D	0	0	0	0	0	
400181-400190	U	U	U	U	U	U	U	U	U	U	
400191-400200	U	U	U	U	U	U	U	U	U	u	
400201-400210	U	U	U	U	U	U	U	U	U	U	
00 01 02 28 27 28	03 04 29 30	05 06 07 31 32 33	08 09 10 34 35 36	11 12 13 37 38 38	14 15 1 40 41 4	0 17 18 1 2 43 44 4	9 20 21 5 45 47	22 23 24 2 48 49 50	25 A V	т Ver. 8.2	Comms

S-232 MODBUS PLO	C Sim-server Settings	· · · · · · · · · · · · · · · · · · ·				
-RS-232 Port		ОК				
Port	COM5* 🔽	Cancel				
Baud rate	9600 💌	lf ∨ou see a little *				
Parity	None 💌	next to the COM port.				
Data bits	8 🔹	or being used by				
Stop bits	1 🔹	another program.				
RTS control	Enable 💌					
– Server settings –]				
Responsivenes	s (ms) 0	(0 to 10 000 ms)				
Load register	values at startup.					
	n ar stanup.	Use Long Timeouts.				
-Advanced setting	1s					
Perform MOD	BUS - MOSCAD (row/to	able) checks				
Allen-Bradley	/ master mode PLC					
Checksum calculation is using CRC (2-bytes -						

(4) Click the comms.

User Manual for LoRaWAN /NB -IoT End Nodes - RS485-NB -- RS485/UART to NB-IoT Converter User Manual

				-						
idress: O H 🖲 🛛) VO Holdin	g Regs (400000) _	- Fmt deci	nal +/- 💌 Prot	MODBUS RS-232					I ⊂ Clon
Address 4	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9
00001-400010 0	1	0	0	0	0	0	0	0	0	0
00011-400020 0	1	0	0	0	0	0	0	0	0	0
00021-400030 0	1	0	0	0	0	0	0	0	0	0
00031-400040 0	1	0	0	0	0	0	0	0	0	0
00041-400050 0	1	0	0	0	0	0	0	0	0	0
00051-400060 0	1	0	0	0	0	0	0	0	0	0
00061-400070 0	1	0	0	0	0	0	0	0	0	0
00071-400080 0)	0	0	0	0	0	0	0	0	0
00081-400090 0	1	0	0	0	0	0	0	0	0	0
00091-400100 0)	0	0	0	0	0	0	0	0	0
00101-400110 0)	0	0	0	0	0	0	0	0	0
00111-400120 0	1	0	0	0	0	0	0	0	0	0
00121-400130 0)	0	0	0	0	0	0	0	0	0
00131-400140 0)	0	0	0	0	0	0	0	0	0
00141-400150 0	1	0	0	0	0	0	0	0	0	0
00151-400160 0	1	0	0	0	0	0	0	0	0	0
00161-400170 0	1	0	0	0	0	0	0	0	0	0
00171-400180 0)	0	0	0	0	0	0	0	0	0
00181-400190 0)	0	0	0	0	0	0	0	0	0
00191-400200 0	1	0	0	0	0	0	0	0	0	0
00201-400210 0)	0	0	0	0	0	0	0	0	0

Run RS485-NB 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.

	-	U	~		[17:27:36.555] 123456
				۲	[17:27:30.585] Correct Password [17:27:39.338] AT+CFGDEV=01 03 00 20 00 01,1
SEND receive					[17:27:39.422] AT+CFGDEV=01 03 20 00 01 85 c0 ,1 [17:27:39.430] RETURN DATA: [17:27:40.497] [17:27:40.497] 01 03 02 00 00 b8 44 [17:27:40.693] 0K [17:27:40.699] OK
					AT_CECNEV_01 03 00 20 00 01 1

We can also use ModRSsim2 to see the output.

17:09:26.391 Starting server(s) 17:09:26.392 Starting comms emulation : MODBUS RS-232 17:09:26.393 Port COM6 opened. 17:09:26.393 Port VO buffers configured. 17:09:26.394 Port configured 9600.8.N.1 17:09:26.394 Timeouts configured (100ms/500ms) 12:09:26.394 Modem status : [CTS_][DSR_][RiblG_] 17:18:32.517 RX01 03 00 20 00 01 7:18:32.518 RX:85 C0 01 03 22 00 17:18:32.518 Rx:9DUSize is 256 17:18:32.518 Read Holding Regs from 32 for 1. 17:18:32.518 TX11 03 02 00 00 R5 44		
00 01 02 03 04 05 05 07 09 09	19 11 12 13 14 15 16 17 11 19 20 21 22	23 24 25 <u>A</u>
net 10 10 10 10 10 10 10 10 10 10 10 10 10	International and the state International and the state <t< td=""><td>▲ <u>发送</u></td></t<>	▲ <u>发送</u>
COM12 OPENED 9600 8 NONE 1 OFF Rv: 16 297 Bytes	AT+CFGDEV-01 03 00 20 00 01,1	Alithon

6.5.5 Example – Test CMD command sets.

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

Serial port tool:

	[12.02.10.000] OV
[17·27·39 417] 01 03 00 20 00 01 85 C0 01 03 02 00 00 B8 44 01 03 00 20 00	[[17:27:40.699] UK
	[17:30:01.230] AT+GETSENSORVALUE=1
01 85 00 01 03 02 00 00 88 44 01 05 00 21 00 01 04 00 01 05 02 00 00 88 44	
01 03 00 22 00 01 24 00 01 03 02 00 20 B9 9C 01 03 00 23 00 01 75 C0 01 03	
02 00 00 B8 44 01 03 00 24 00 01 C4 01 01 83 01 80 F0	[17,20,01,060]
	[17:30:01.200]
	[[17:30:01.261] UK
	[17:30:03.349]
	[17:30:03.351] CMD1 = 01 03 00 20 00 01 85 c0
	[17:30:03.401]
	[17:30:03.402] RETURN1 = 01 03 02 00 00 b8 44
	[17:30:03.440]
	[17:20:02,441] CWD2 = 01.02.00.21.00.01.44.00
	[17.30.03.441] CMD2 - 01 03 00 21 00 01 04 00
	[17:30:03.487]
	[17:30:03.491] RETURN2 = 01 03 02 00 00 68 44
	[17:30:03.532] CMD3 = 01 03 00 22 00 01 24 00
	[17:30:03.571] RETURN3 = 01 03 02 00 20 b9 9c
	[17:30:03.611]
	[17:30:03.611] CWD4 = 01.03.00.23.00.01.75 c0
	[17:20:02.661] RETURNA = 01.02.02.00.00 b9.44
	[17.00.00 700]
	[[17:30:03.702] CMD5 = 01 03 00 24 00 01 c4 01
	[17:30:03.751]
	[17:30:03.751] RETURN5 = 01 83 01 80 f0 00 00
	[17:30:03.779]
	La l'our out i of

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	4
[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]	CHD1 = 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 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]	CHD5 = 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	

B MODBUS RTU RS-232 PLC - Simulator (port: COM16 9600,8,N,1,	R-on)	- 0 X
Connected (1) : (received/sent) (82/82) Serv. read data. Po: 0	• Тк •	\$ \$ 8 ♥ ₩ \$. 11 ₹
Pause Stop tracking Clear	Show times	□ Log
16:03:55.969 m_PDUSize is 256 16:03:55.969 m_PDUSize is 256		
16:03:55:565 Read Holding Regs from 34 for 1.		
16:03:55:335 PX 01 03:02 23:00 01		
16:03:56:377 RX:75 C0		
16:03:56.377 m_PDUSize is 256		
16:03:56.377 Read Holding Regs from 35 for 1.		
16:03:56.377 TX:01 03 02 08 F2 3E 01		
16:03:56.685 RX:01 03 00 24 00 01		
15:03:56.796 PXC4 01		
10:03:56,796 m_PUUSi26 is 256 10:03:56,796 Deed Heldier: Deed from 36 feet1		
16.03.56.736 Meda Holding Hegs from 36 for 1.		
16:04:07 137 BX:01 03 00 20 00 01		
16:04:07.250 EX:85 C0		
16:04:07.250 m_PDUSize is 258		
16:04:07.250 Read Holding Regs from 32 for 1.		
16:04:07.250 TX:01 03 02 09 1A 3F DF		
16:04:07.549 FX:01 03 00 21 00 01		
16:04:07:654 FX:D4:00		
16:04:07:654 m_PDUSize is 256 19:04:07:654 Deed Helding Deep from 22 for 1		
16:04:07:654 TX:01:03:02:09:14:3E DE		
		A Desister
		Registers
26 27 28 29 30 31 32 33 34 38 38	37 38 39 40 41 42 43 44 45 48 47 48 49 50	V Ver. 8.21.2.7

7. Order Info

Part Number: RS485-NB-XX

XX:

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

YY: The grand connector hole size

- M12: M12 hole
- M16: M16 hole

8. Packing Info

Package Includes:

- RS485-NB NB-IoT Sensor Node x 1
- External antenna x 1

Dimension and weight:

- Device Size: 13.0 x 5 x 4.5 cm
- Device Weight: 150g
- Package Size / pcs : 14.0 x 8x 5 cm
- Weight / pcs : 180g

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.cc.