RAK4200 Module AT Command Manual

Introduction

The RAK4200 module is designed to simplify LoRa P2P peer-to-peer and LoRaWAN communication. This module saves you in dealing with complicated SPI protocol with the LoRa transceivers. Instead, a well-known serial communication interface is provided for sending commands and requesting the internal status of the module. This approach allows a straightforward way to integrate LoRa technology into your projects.

On top of this serial interface, a set of AT commands is defined. An external microcontroller will be able to control the RAK4200 module as a classic AT modem. Through the AT commands, you can set parameters of the LoRaWAN communication, controlling GPIO pins, analog inputs, etc.

In the RAK4200 module, the serial communication is exposed on the **UART1 port**, through **Pin 4 (UART1_TX)** and **Pin 5 (UART1_RX)**. The parameters of the UART1 communication are **115200** / **8-N-1**. The firmware upgrade is also possible through this port. To get familiar with the pin distribution of this module and find a schematic circuit of a reference application, refer to RAK4200 Datasheet. A summary is also provided in the Appendix IV.

In addition, the RAK4200 module also exposes another serial port through **Pin 2 (UART2_TX)** and **Pin 1 (UART2_RX)**. This port is named UART2. You can use it to connect another MCU or an additional UART peripheral, such as a GPS module.

UART2 is **Pin 2 (TX2)** and **Pin 1 (RX2)** on modules. In the case that the target application only requires one single UART port, then it is recommended to make use of the UART2 to connect to your MCU and reserve the UART1 for future firmware upgrades.

Links to Quick Start Guide

For AT commands example usage, you can check these sections of the quick start guide:

- TTN OTAA/ABP
- ChirpStack OTAA/ABP
- LoRa P2P

Software Tool

If you don't have a serial port tool yet, it is recommended to download and install the RAK Serial Port Tool. There are some ready-made AT commands in this tool that will be very useful for you.

- RAK Serial Port Tool

For more detailed information on how to use this tool, refer to the following guide:

- RAK Serial Port Tool Guide

Content

AT Command Syntax

The AT command is based on ASCII characters. A command begins with the prefix `at` and ends with `<CR><LF>` (i.e. `\r\n`). The maximum length is **255 characters** which includes the `<CR><LF>` characters at the end of the command. For the rest of the document, the `\r\n` part is omitted for the sake of clarity.

The AT commands can be classified into the following groups:

- **Read Command**: Reads the current configuration or status of the module. The command name and the list of parameters are separated by the `=` character. The `<m>` parameter is separated with its associated value `<n>` by the `:` character.

```
`at+get_config=<m>:<n>`
```

- **Write Command**: Writes/modifies the current configuration of the module. The command name and the list of parameters are separated by the `=` character. The `<m>` parameter is separated with its associated value `<n>` by the `:` character.

```
`at+set_config=<m>:<n>`
```

- **Operational Command**: Some commands are neither read nor write commands but are used to execute an action.

```
at+send=lora:<m>:<n> // Sends data through the LoRa transceiver.
```

- **Special Command**: The RAK4200 UART port has two operational modes: **Configuration Mode** (default mode) and **Data Transmission Mode**. Data transmission mode allows you to send ASCII payloads directly to the network server via UART without using any AT Command interface like `at+send=lora:X:YYY`. Data transmission mode is explained further on Interface Type AT Command section of this document.

> 📝 **NOTE:**
>
> To enable data transmission mode, you need to input `at+set_config=device:uart_mode:<index>:<mode>` command. To switch back from data transmission mode to configuration mode (AT command default mode), the command to be entered is `+++` and does not contain terminators such as `\r` and `\n`.

After the command is executed by the module, a reply is sent back to the external MCU. In the case the command is successful, the usual reply has the following format:

```
OK [information]\r\n
```

> 📝 **NOTE:**
>
> Only Read commands have information in the replied message, while Write commands do not have an informative description.

The firmware you developed, running in the external MCU, will expect at a minimum a string of `Ok\r\n` after sending a successful command to the module. On the other hand, when the command is not successfully executed by the module, a reply will be received in the following format:

```
ERROR: [ErrCode]\r\n
```

Error Code Table

Error Code	Description
1	The last command received is an unsupported AT command.
2	Invalid parameter in the AT command.
3	There is an error when reading or writing the flash memory.
5	There is an error when sending data through the UART port. Check if you exceed 256 bytes UART buffer.
80	The LoRa transceiver is busy, could not process a new command.
81	LoRa service is unknown. Unknown MAC command received by the node. Execute commands that are not supported in the current state, such as sending the `at+join` command in P2P mode.
82	The LoRa parameters are invalid.
83	The LoRa frequency is invalid.
84	The LoRa data rate (DR) is invalid.
85	The LoRa frequency and data rate are invalid.
86	The device hasn’t joined into a LoRa network.
87	The length of the packet exceeded that maximum allowed by the LoRa protocol.
88	Service is closed by the server. Due to the limitation of the duty cycle, the server will send the "SRV_MAC_DUTY_CYCLE_REQ" MAC command to close the service.
89	This is an unsupported region code.
90	Duty cycle is restricted. Due to the duty cycle, data cannot be sent at this time until the time limit is removed.
91	No valid LoRa channel could be found.
92	No available LoRa channel could be found.
93	Status is error. Generally, the internal state of the protocol stack is wrong.
94	Time out reached while sending the packet through the LoRa transceiver.
95	Time out reached while waiting for a packet in the LoRa RX1 window.
96	Time out reached while waiting for a packet in the LoRa RX2 window.
97	There is an error while receiving a packet during the LoRa RX1 window.
98	There is an error while receiving a packet during the LoRa RX2 window.

Error Code	Description
99	Failed to join into a LoRa network.
100	Duplicate downlink message is detected. A message with an invalid downlink count is received.
101	Payload size is not valid for the current data rate (DR).
102	Many downlink packets are lost.
103	Address fail. The address of the received packet does not match the address of the current node.
104	Invalid MIC is detected in the LoRa message.

General AT Command

1. **at+version**

This command is used for reading the version number of the current firmware.

Operation	Command	Response
Read	`at+version`	`OK <version number>`

Parameter: None

Example:

```
at+version\r\n
OK V3.2.0.14
```

2. **at+help**

This command is used to obtain all the AT commands supported by the current firmware.

Operation	Command	Response
Read	`at+help`	`OK <all AT commands>`

Parameter: None

Example:

```
at+help\r\n
OK Device AT commands:
at+version
at+help
at+set_config=device:restart
at+set_config=device:sleep:X
at+get_config=device:status
at+set_config=device:uart:X:Y
at+set_config=device:uart_mode:X:Y
at+send=uart:X:YYY
at+set_config=device:gpio:X:Y
at+get_config=device:gpio:X
at+get_config=device:adc:X

LoRaWAN AT commands:
at+set_config=lora:default_parameters
at+join
at+send=lora:X:YYY
at+set_config=lora:region:XXX
at+get_config=lora:channel
at+set_config=lora:dev_eui:XXXX
at+set_config=lora:app_eui:XXXX
at+set_config=lora:app_key:XXXX
at+set_config=lora:dev_addr:XXXX
at+set_config=lora:apps_key:XXXX
at+set_config=lora:nwks_key:XXXX
at+set_config=lora:multicastenable:X
at+set_config=lora:multicast_dev_addr:XXXX
at+set_config=lora:multicast_apps_key:XXXX
at+set_config=lora:multicast_nwks_key:XXXX
at+set_config=lora:join_mode:X
at+set_config=lora:work_mode:X
at+set_config=lora:ch_mask:X:Y
at+set_config=lora:class:X
at+set_config=lora:confirm:X
at+set_config=lora:dr:X
at+set_config=lora:tx_power:X
at+set_config=lora:adr:X
at+get_config=lora:status
at+set_config=lora:dutycycle_enable:X
at+set_config=lora:send_repeat_cnt:X

LoRaP2P AT commands:
at+set_config=lorap2p:XXX:Y:Z:A:B:C
at+set_config=lorap2p:transfer_mode:X
at+send=lorap2p:XXX
```

3. **at+set_config=device:restart**

This command is used to restart the device.

Operation	Command	Response
Write	`at+set_config=device:restart`	

Parameter: None

Example:

```
at+set_config=device:restart\r\n
UART1 work mode: RUI_UART_NORMAL
Current work_mode:LoRaWAN, join_mode:ABP, Class: A
Initialization OK
```

4. **at+set_config=device:sleep:** `<status>`

This command is used to change the current state of the device between the sleep and the wake-up mode.

Operation	Command	Response
Write	`at+set_config=device:sleep:<status>`	`OK <STATUS>`

Parameter	Description
status	0: wake up
	1: sleep

Example

```
at+set_config=device:sleep:1\r\n
OK Sleep

at+set_config=device:sleep:0\r\n
OK Wake Up
```

> 📝 **NOTE:**
>
> During sleep, Pin 5 (RX1) and Pin 1 (RX2) are automatically configured as wake up pins and in external interrupt mode with an internal pull-down resistor. Wake-up will be triggered by a rising edge on these RX pins.

5. **at+get_config=device:status**

This command is used to obtain the current status of the device.

Operation	Command	Response
Read	`at+get_config=device:status`	`OK <information>`

Parameter: None

Example:

```
at+get_config=device:status\r\n
OK Board Core:RAK4200
MCU:STM32L071KB
LoRa chip:SX1276
```

Interface Type AT Command

1. **at+set_config=device:uart:** `<index>:<baud_rate>`

This command is used to configure the baud rate for a UART port.

Operation	Command	Response
Write	`at+set_config=device:uart:<index>:<baud_rate>`	`OK`

Parameter:

index	UART Number (1 or 2)
baud_rate	UART Baud rate：1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200

Example:

```
at+set_config=device:uart:1:115200\r\n
OK
```

2. **at+set_config=device:uart_mode:** `<index>:<mode>`

This command is used to set the UART operation from AT **configuration mode** to **data transmission mode**.

During **data transmission mode**, all standard AT Commands will not work and the data that you sent to UART will go directly to the network server as ASCII payload with `\r\n`. If you input `AZ`, the network server will receive an uplink hex value of `415A0D0A`. This means **A=** `0x41`, **Z=** `0x5A`, **\r=** `0x0D` and **\n=** `0x0A`.

> 📝 **NOTE:**
>
> To switch back from data transmission mode to configuration mode, use `+++` (`+++` without `\ r\ n`).

Operation	Command	Response
Write	`at+set_config=device:uart_mode:<index>:<mode>`	`OK`

Parameter:

index	UART Number (1 or 2)
mode	UART Mode： Only 1 can be selected, which means the UART is set to data transmission mode.

Example:

```
at+set_config=device:uart_mode:1:1\r\n
OK

+++
OK
```

3. **at+send=uart:** `<index>:<data>`

This command is used to send data over a UART port.

Operation	Command	Response
Write	`at+send=uart:<index>:<data>`	`OK`

Parameter:

index	UART Port Number. Currently, the RAK4200 supports UART1 and UART2.
data	The data you want to send. The maximum length of data is **250 characters**, equivalent to 255 — the length of `at+...` — the length of `\ r\ n.`

Example:

```
at+send=uart:1:12345\r\n
OK
```

4. **at+get_config=device:gpio:** `<pin_num>`

This command is used to obtain the voltage level status of a pin on a module.

Operation	Command	Response
Read	`at+get_config=device:gpio:<pin_num>`	`OK <status>`

Parameter:

pin_num	Pin index of the module (GPIO pins available are Pin 3, Pin 6, Pin 9, and Pin 10)
status（Return Value）	0: Low Voltage Level 1: High Voltage Level

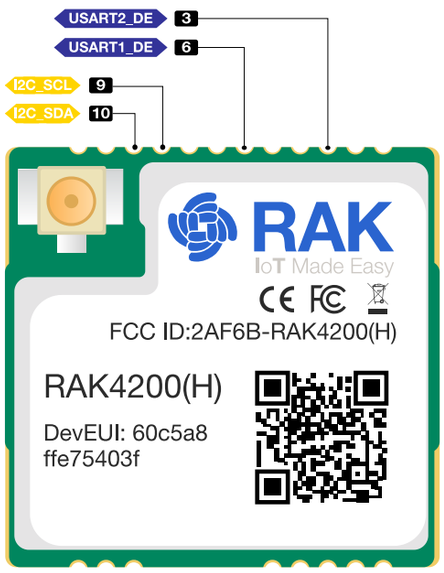

Figure 1: GPIO Pinout for RAK4200

Example:

```
at+get_config=device:gpio:3\r\n
OK 1
```

5. **at+set_config=device:gpio:** `<pin_num>:<status>`

This command is used to set the voltage level state (high or low) of a pin on a module.

Operation	Command	Response
Write	`at+set_config=device:gpio:<pin_num>:<status>`	`OK`

Parameter:

pin_num	Pin index of the module (GPIO pins available are Pin 3, Pin 6, Pin 9, and Pin 10) **Refer to Figure 1.**
status	0: Low Voltage Level 1: High Voltage Level

Example:

```
at+set_config=device:gpio:3:0\r\n
OK
```

6. **at+get_config=device:adc:** `<pin_num>`

This command is used to obtain the voltage level of an ADC pin of the module.

Operation	Command	Response
Read	`at+get_config=device:adc:<pin_num>`	`OK <voltage>`

Parameter:

pin_num	ADC pin index of the module (ADC pin is assigned to Pin 3)
Voltage（Return Value）	Voltage，Unit: **mV**

Figure 2: ADC Pinout for RAK4200

Example:

```
at+get_config=device:adc:3\r\n
OK 1663mV
```

LoRaWAN Type AT Command

1. **at+join**

This command is used to join a LoRaWAN network.

Operation	Command	Response
	`at+join`	`OK Join Success`

Parameter: None

Example:

```
at+join\r\n
OK Join Success
```

2. **at+send=lora:** `<port>:<data>`

This command is used to send data via LoRaWAN.

Operation	Command	Response
	`at+send=lora:<port>:<data>`	`OK`

Parameter:

port	Sending port of LoRa. The value range is 1-223.
data	The sending data format is in hexadecimal format. The possible values are between **00-FF**. The module will internally cast every two characters into a byte before sending it to the LoRa transceiver. The maximum length varies depending on the band frequency and DR (LoRaWAN standard). Refer to Appendix III.

Example :

When sending data as unconfirmed uplink:

```
at+send=lora:1:5A00\r\n
OK
```

When sending data as confirmed uplink:

```
at+send=lora:1:5A00\r\n
OK
at+recv=0,-105,-12,0
```

> 📝 **NOTE**
>
> - When sending a confirmed message, you will receive an ACK response, i.e. `at+recv=...`. The `0, -105, -12,0` stands for:
>   - `0`: For the LoRa port;
>   - `-105`: For the RSSI;
>   - `-12`: For the SNR;
>   - `0`: For the length of the data (no valid data in ACK).
> - When sending an unconfirmed message, sometimes the gateway will send MAC commands to nodes, and the node will also receive `at+recv=...`.

3. **at+set_config=lora:region:** `<region>`

This command is used to set the appropriate working frequency band.

Operation	Command	Response
Write	`at+set_config=lora:region:<region` >	`OK`

Parameter:

region EU433, CN470, IN865, EU868, US915, AU915, KR920, AS923. The default is EU868.

Example：

```
at+set_config=lora:region:EU868\r\n
OK
```

> 📝 **NOTE**
>
> In the AS923 frequency band, the supported frequency plan is "as2" and the dwell time is set to 1.

4. **at+get_config=lora:channel**

This command is used to read all the LoRa channel information given the current region configured on the board.

Operation	Command	Response
Read	`at+get_config=lora:channel`	`OK <channel information>`

Parameter: None

Example: EU868 Region

```
at+get_config=lora:channel\r\n
OK *0,on,868100000,0,5; *1,on,868300000,0,5; *2,on,868500000,0,5; 3,off,0,0,0; 4,off,0,0,0; 5,off
```

> 📝 **NOTE**
>
> With ***0,on,868100000,0,5** as an example，the following is the channel parameter analysis:
>
> - `*` at the beginning, if the channel is open;
> - `0` is the channel ID;
> - `on` indicates the current status of the channel;
> - `868100000` is the actual frequency of the channel，unit is Hz;
> - `0,5` indicates the DR of the channel, DR0~DR5.

5. **at+set_config=lora:ch_mask:** `<channel_number>:<status>`

This command is used to enable (on) or disable (off) a channel in the current region.

Operation	Command	Response
Write	`at+set_config=lora:ch_mask:<channel_number>:<status>`	`OK`

Parameter:

channel_number	Channel number
status	0: off 1: on

Example:

```
at+set_config=lora:ch_mask:0:0\r\n
OK
```

6. **at+set_config=lora:dev_eui:** `<dev_eui>`

This command is used to set the Device EUI parameter for the LoRaWAN OTAA mode.

Operation	Command	Response
Write	`at+set_config=lora:dev_eui:<dev_eui>`	`OK`

Parameter:

dev_eui	Device EUI

Example :

```
at+set_config=lora:dev_eui:3530353064377716\r\n
OK
```

7. **at+set_config=lora:app_eui:** `<app_eui>`

This command is used to set the Application EUI parameter for the LoRaWAN OTAA mode.

Operation	Command	Response
Write	`at+set_config=lora:app_eui:<app_eui>`	`OK`

Parameter:

app_eui	Application EUI

> 📝 **NOTE:**
>
> All zero value Application EUI `at+set_config=lora:app_eui:0000000000000000` is **not supported** and will return error.

Example :

```
at+set_config=lora:app_eui:0000000000000001\r\n
OK
```

8. **at+set_config=lora:app_key:** `<app_key>`

This command is used to set the Application Key parameter for the LoRaWAN OTAA mode.

Operation	Command	Response
Write	`at+set_config=lora:app_key:<app_key>`	`OK`

Parameter:

app_key	Application Key

Example:

```
at+set_config=lora:app_key:841986913ACD00BBC2BE2479D70F3228\r\n
OK
```

9. **at+set_config=lora:dev_addr:** `<dev_addr>`

This command is used to set the Device Address parameter for the LoRaWAN ABP mode.

Operation	Command	Response
Write	`at+set_config=lora:dev_addr:<dev_addr>`	`OK`

Parameter:

dev_addr	Device Address

Example:

```
at+set_config=lora:dev_addr:260125D7\r\n
OK
```

10. **at+set_config=lora:apps_key:** `<apps_key>`

This command is used to set the Application Session Key parameter for the LoRaWAN ABP mode.

Operation	Command	Response
Write	`at+set_config=lora:apps_key:<apps_key>`	`OK`

Parameter:

apps_key	Application Session Key

Example:

```
at+set_config=lora:apps_key:841986913ACD00BBC2BE2479D70F3228\r\n
OK
```

11. **at+set_config=lora:nwks_key:** `<nwks_key>`

This command is used to set the Network Session Key parameter for the LoRaWAN ABP mode.

Operation	Command	Response
Read	`at+set_config=lora:nwks_key:<nwks_key>`	`OK`

Parameter:

nwks_key	Network Session Key

Example:

```
at+set_config=lora:nwks_key:69AF20AEA26C01B243945A28C9172B42\r\n
OK
```

12. **at+set_config=lora:multicastenable:** `<IsEnable>`

This command is used to enable or disable the multicast feature.

Operation	Command	Response
Write	`at+set_config=lora:multicastenable:<IsEnable>`	`OK`

Parameter:

IsEnable	0: disable 1: enable The default is disable.

Example:

```
at+set_config=lora:multicastenable:1\r\n
OK
```

13. **at+set_config=lora:multicast_dev_addr:** `<multicast_dev_addr>`

This command is used to set the Device Address for the multicast feature.

Operation	Command	Response
Write	`at+set_config=lora:multicast_dev_addr:<multicast_dev_addr>`	`OK`

Parameter:

multicast_dev_addr	Multicast Device Address

Example:

```
at+set_config=lora:multicast_dev_addr:260111fd\r\n
OK
```

14. **at+set_config=lora:multicast_apps_key:** `<multicast_apps_key>`

This command is used to set the Application Session Key for the multicast feature.

Operation	Command	Response
Write	`at+set_config=lora:multicast_apps_key:<multicast_apps_key>`	`OK`

Parameter:

multicast_app_addr	Multicast Application Session Key

Example:

```
at+set_config=lora:multicast_apps_key:F13DDFA2619B10411F02F042E1C0F356\r\n
OK
```

15. **at+set_config=lora:multicast_nwks_key:** `<multicast_nwks_key>`

This command is used to set the Network Session Key for the multicast feature.

Operation	Command	Response
Write	`at+set_config=lora:multicast_nwks_key:<multicast_nwks_key>`	`OK`

Parameter:

multicast_nwks_key	Multicast Network Session Key

Example:

```
at+set_config=lora:multicast_nwks_key:1D1991F5377C675879C39B6908D437A6\r\n
OK
```

16. **at+set_config=lora:join_mode:** `<mode>`

This command is used to switch the LoRaWAN access mode between the OTAA and the ABP mode.

Operation	Command	Response
Write	`at+set_config=lora:join_mode:<mode>`	`OK`

Parameter:

mode	Activation mode 0: OTAA 1: ABP The default is OTAA.

Example：

```
at+set_config=lora:join_mode:1\r\n
OK
```

17. **at+set_config=lora:class:** `<class>`

This command is used to set LoRaWAN class to Class A, Class B, or Class C.

Operation	Command	Response
Write	`at+set_config=lora:class:<class>`	`OK`

Parameter:

class	0: Class A 1: Class B （Not supported at this time） 2: Class C The default is Class A.

Example:

```
at+set_config=lora:class:0\r\n
OK
```

18. **at+set_config=lora:confirm:** `<type>`

This command is used to set the type of data to be sent: Confirmed/Unconfirmed.

Operation	Command	Response
Write	`at+set_config=lora:confirm:<type>`	`OK`

Parameter:

type	0: Unconfirm Type 1: Confirm Type The default is unconfirm type.

Example:

```
at+set_config=lora:confirm:0\r\n
OK
```

19. **at+set_config=lora:dr:** `<dr>`

This command is used to set the data rate (DR) of LoRa.

Operation	Command	Response
Write	`at+set_config=lora:dr:<dr>`	`OK`

Parameter:

dr	The data rate of LoRa is related to the current region. In most of the LoRa areas, it is common to use 0 to 5. Detailed reference can be made to LoRaWAN 1.0.2 specification.

20. **at+set_config=lora:tx_power:** `<tx_power>`

This command is used to set the RF transmission power level of the LoRa transceiver.

Operation	Command	Response
Write	`at+set_config=lora:tx_power:<tx_power>`	`OK`

Parameter:

tx_power	Refer to Appendix II for possible values of tx_power. The table of Appendix II is based on LoRaWAN 1.0.2 specification. LoRa transmit power level varies depending on frequency band. If the resulting TX power is higher than the capability of LoRa Radio, the output power will be based on the max TX power of the LoRa Radio in the module. For RAK4200 module, the max TX power is 20 dBm. Take note of this when using regional bands with MaxEIRP higher than 20 dBm like US915, AU915 and IN865 whose MaxEIRP is 30 dBm. The default setting is 0.

Example:

```
at+set_config=lora:tx_power:0\r\n
OK
```

21. **at+set_config=lora:adr:** `<status>`

This command is used to turn on/off the ADR feature of the LoRa communication.

Operation	Command	Response
Write	`at+set_config=lora:adr:<status>`	`OK`

Parameter:

status	0: Turn off 1: Turn on The default is on.

Example:

```
at+set_config=lora:adr:0\r\n
OK
```

22. **at+get_config=lora:status**

This command is used to get all the information related to the current LoRa status, except the channel information.

Operation	Command	Response
Read	`at+get_config=lora:status`	`OK <lora status detail>`

Parameter: None

Example:

```
at+get_config=lora:status\r\n
OK Work Mode: LoRaWAN
Region: EU868
Send_interval: 600s
Auto send status: false.
MulticastEnable: true.
Multi_Dev_Addr: 260111FD
Multi_Apps_Key: F13DDFA2619B10411F02F042E1C0F356
Multi_Nwks_Key: 1D1991F5377C675879C39B6908D437A6
Join_mode: OTAA
DevEui: 0000000000000888
AppEui: 0000000000000888
AppKey: 00000000000008880000000000000888
Class: C
Joined Network:false
IsConfirm: unconfirm
AdrEnable: true
EnableRepeaterSupport: false
RX2_CHANNEL_FREQUENCY: 869525000, RX2_CHANNEL_DR:0
RX_WINDOW_DURATION: 3000ms
RECEIVE_DELAY_1: 1000ms
RECEIVE_DELAY_2: 2000ms
JOIN_ACCEPT_DELAY_1: 5000ms
JOIN_ACCEPT_DELAY_2: 6000ms
Current Datarate: 4
Primeval Datarate: 4
ChannelsTxPower: 0
UpLinkCounter: 0
DownLinkCounter: 0
```

23. **at+set_config=lora:dutycycle_enable:** `<status>`

This command is used to enable or disable the Duty Cycle feature.

Operation	Command	Response
Write	`at+set_config=lora:dutycycle_enable:<status>`	`OK`

Parameter:

status	0: disable 1: enable The default is disable.

Example:

```
at+set_config=lora:dutycycle_enable:1\r\n
OK
```

24. **at+set_config=lora:send_repeat_cnt:** `<num>`

This command is used to set the number of retransmitting attempts on an uplink message. When activated, the board will resend a message if its corresponding ACK (downlink) is not received after sending a confirmed uplink message. The default value is 0, which means that the board will not resend any message by default.

Operation	Command	Response
Write	`at+set_config=lora:send_repeat_cnt:<status>`	`OK`

Parameter:

num	Number of retries, up to 7. The default is 0.

Example:

```
at+set_config=lora:send_repeat_cnt:1\r\n
OK
```

25. **at+set_config=lora:default_parameters**

This command is used to restore OTAA, ABP, multicast related network access parameters set at the factory, including dev_eui，app_eui, etc.

Operation	Command	Response
Write	`at+set_config=lora:default_parameters`	`OK`

Parameter: none

Example:

```
at+set_config=lora:default_parameters\r\n
OK
```

LoRa P2P Type AT Command

1. **at+set_config=lora:work_mode:** `<mode>`

This command is used to switch the LoRa work mode between the LoRaWAN and the LoRa P2P mode. This command will cause the module to restart.

Operation	Command	Response
Write	`at+set_config=lora:work_mode:<mode>`	

Parameter:

mode	Work Mode of LoRa 0: LoRaWAN 1: LoRa P2P The default is LoRaWAN mode.

Example:

```
at+set_config=lora:work_mode:1\r\n
UART1 work mode: RUI_UART_NORMAL
Current work_mode:P2P
Initialization OK
```

2. **at+set_config=lorap2p:** `<frequency>:<spreadfact>:<bandwidth>:<codingrate>:<preamlen>:<power>`

This command is used to set the relevant parameters of LoRa P2P mode and is only valid when the LoRa mode is switched to LoRa P2P before.

Operation	Command	Response
Write	`at+set_config=lorap2p:<frequency>:<spreadfact>:<bandwidth>:<codingrate>:<preamlen>:<power>`	`OK`

Parameter:

frequency	Frequency, the unit is Hz The default is 869525000 Hz.
spreadfact	Spreading factor The default is 12.
bandwidth	0: 125 kHz 1: 250 kHz 2: 500 kHz The default is 0.
codingrate	1: 4/5 2: 4/6 3: 4/7 4: 4/8 The default is 1.
preamble	Preamble Length. 5~65535 The default is 8.
power	TX power. The unit is in dBm. 5~20 The default is 20.

Example:

```
at+set_config=lorap2p:869525000:12:0:1:8:20\r\n
OK
```

3. **at+set_config=lorap2p:transfer_mode:** `<mode>`

This command is used to switch the state of the LoRa transceiver between sending and receiving state, and it is only valid when the LoRa mode is set to LoRa P2P before.

Operation	Command	Response
Write	`at+set_config=lorap2p:transfer_mode:<mode>`	`OK`

Parameter：

mode	1: receiver mode 2: sender mode The default is sender mode.

Example:

```
at+set_config=lorap2p:transfer_mode:1\r\n
OK
```

4. **at+send=lorap2p:** `<data>`

This command is used to send data in LoRa P2P mode, and it is only valid when the LoRa mode is set to LoRa P2P before.

Operation	Command	Response
Send	`at+send=lorap2p:<data>`	`OK`

Parameter：

data	The data to be sent, and the format is hexadecimal.

Example:

```
at+send=lorap2p:1234\r\n
OK
```

In LoRa P2P mode, the receiving node receives the data and outputs the data in the following format:

```
at+recv=<RSSI>,<SNR>,<Data Length>:<Data>
```

Appendix I: Data Rate by Region

EU868/EU433/AS923

Data Rate	Configuration	Indicative Physical Bit Rate [bit/s]
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6	LoRa: SF7 / 250 kHz	11000
7	FSK: 50 kbps	50000
8 ~ 15	RFU	

CN470/KR920

Data Rate	Configuration	Indicative Physical Bit Rate [bit/s]
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6 ~ 15	RFU	

US915

Data Rate	Configuration	Indicative Physical Bit Rate [bit/s]
0	LoRa: SF10 / 125 kHz	980
1	LoRa: SF9 / 125 kHz	1760
2	LoRa: SF8 / 125 kHz	3125
3	LoRa: SF7 / 125 kHz	5470
4	LoRa: SF8 / 500 kHz	12500
5 ~ 7	RFU	
8	LoRa: SF12 / 500 kHz	980
9	LoRa: SF11 / 500 kHz	1760
10	LoRa: SF10 / 500 kHz	3900
11	LoRa: SF9 / 500 kHz	7000
12	LoRa: SF8 / 500 kHz	12500
13	LoRa: SF7 / 500 kHz	21900
14 ~ 15	RFU	

AU915

Data Rate	Configuration	Indicative Physical Bit Rate [bit/s]
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6	LoRa: SF8 / 500 kHz	12500
7	RFU	RFU
8	LoRa: SF12 / 500 kHz	980
9	LoRa: SF11 / 500 kHz	1760
10	LoRa: SF10 / 500 kHz	3900
11	LoRa: SF9 / 500 kHz	7000
12	LoRa: SF8 / 500 kHz	12500

IN865

Data Rate	Configuration	Indicative Physical Bit Rate [bit/s]
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6	RFU	RFU
7	FSK: 50 kbps	50000
8 ~ 15	RFU	RFU

Appendix II： TX Power by Region

EU868

By default, MaxEIRP is considered to be +16 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
3	MaxEIRP - 6 dB
4	MaxEIRP - 8 dB
5	MaxEIRP - 10 dB
6	MaxEIRP - 12 dB
7	MaxEIRP - 14 dB
8 ~ 15	RFU

US915

By default, MaxEIRP is considered to be +30 dBm based on LoRa Alliance specification. However, the module's max TX power is only up to 20 dBm.

TXPower	Configuration (Conducted Power)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
2	26 dBm
3 ~ 9	-
10	10 dBm
11 ~ 15	RFU

AU915

By default, MaxEIRP is considered to be +30 dBm based on LoRa Alliance specification. However, the module's max TX power is only up to 20 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1 ~ 10	MaxEIRP - 2*TXPower
11 ~ 15	RFU

KR920

By default, MaxEIRP is considered to be +14 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
3	MaxEIRP - 6 dB
4	MaxEIRP - 8 dB
5	MaxEIRP - 10 dB
6	MaxEIRP - 12 dB
7	MaxEIRP - 14 dB
8 ~ 15	RFU

AS923

By default, Max EIRP is considered to be 16 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
3	MaxEIRP - 6 dB
4	MaxEIRP - 8 dB
5	MaxEIRP - 10 dB
6	MaxEIRP - 12 dB
7	MaxEIRP - 14 dB
8 ~ 15	RFU

IN865

By default, MaxEIRP is considered to be +30 dBm based on LoRa Alliance specification. However, the module's max TX power is only up to 20 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
3	MaxEIRP - 6 dB
4	MaxEIRP - 8 dB
5	MaxEIRP - 10 dB
6	MaxEIRP - 12 dB
7	MaxEIRP - 14 dB
8	MaxEIRP - 16 dB
9	MaxEIRP - 18 dB
10	MaxEIRP - 20 dB
11 ~ 15	RFU

CN470

By default, MaxEIRP is considered to be +19.15 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
3	MaxEIRP - 6 dB
4	MaxEIRP - 8 dB
5	MaxEIRP - 10 dB
6	MaxEIRP - 12 dB
7	MaxEIRP - 14 dB
8 ~ 15	RFU

EU433

By default, MAxEIRP is considered to be +12.15 dBm.

TXPower	Configuration (EIRP)
0	MaxEIRP
1	MaxEIRP - 2 dB
2	MaxEIRP - 4 dB
3	MaxEIRP - 6 dB
4	MaxEIRP - 8 dB
5	MaxEIRP - 10 dB
6 ~ 15	RFU

Appendix III： Maximum Transmission Load by Region

> 📝 **NOTE:**
>
> The LoRaWAN stack adds 8 bytes to the user payload. In the following list, M is the maximum payload size and N is the maximum usable payload size for the user data without the MAC header.

EU868

DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	250	242
5	250	242
6	250	242
7	250	242
8 ~ 15	Not Defined	Not Defined

US915

DataRate	M	N
0	19	11
1	61	53
2	133	125
3	250	242
4	250	242
5 ~ 7	Not Defined	Not Defined
8	61	53
9	137	129
10	250	242
11	250	242
12	250	242
13	250	242
14 ~ 15	Not Defined	Not Defined

AU915

DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	250	242
5	250	242
6	250	242
7	Not Defined	Not Defined
8	61	53
9	137	129
10	250	242
11	250	242
12	250	242
13	250	242
14 ~ 15	Not Defined	Not Defined

KR920

DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	250	242
5	250	242
6 ~ 15	Not Defined	Not Defined

AS923

DataRate	Uplink MAC Payload Size (M)		Downlink MAC Payload Size (M)	
	UplinkDwellTime = 0	UplinkDwellTime = 1	DownlinkDwellTime = 0	DownlinkDwellTime = 1
0	59	N/A	59	N/A
1	59	N/A	59	N/A
2	59	19	59	19
3	123	61	123	61
4	250	133	250	133
5	250	250	250	250
6	250	250	250	250
7	250	250	250	250
8 ~ 15	RFU		RFU	

IN865

DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	250	242
5	250	242
6	250	242
7	250	242
8 ~ 15	Not Defined	Not Defined

CN470

DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	250	242
5	250	242
6 ~ 15	Not Defined	Not Defined

EU433

DataRate	M	N
0	59	51
1	59	51
2	59	51
3	123	115
4	250	242
5	250	242
6	250	242
7	250	242
8 ~ 15	Not Defined	Not Defined

Appendix IV: Pin Description of RAK4200

The pin definition of the RAK4200 module can be reviewed in the Pin Definition section of the Datasheet.

Listed are the summary of the pins of the RAK4200 module:

1. **About the UART Pin**:
   - Pin 4 (TX1) and Pin 5 (RX1) are reserved for UART1.
   - Pin 2 (TX2) and Pin 1 (RX2) are reserved for UART2.
   - During sleep, Pin 5 (RX1) and Pin 1 (RX2) are automatically configured as wake up pins and in external interrupt mode with an internal pull-down resistor. Wake-up will be triggered by a rising edge on these RX pins.
2. **About the SWD Debug Pin**: Pin 7 (SWDIO) and Pin 8 (SWCLK) are used for SWD debug port.
3. **About the Power Pin**: The power pins on the RAK4200 module include VCC on Pin 20 and GND on Pin 11, Pin 13, Pin 14, and Pin 19.
4. **About the Reset Pin**: The reset pin on the RAK4200 module is Pin 18.
5. **About the RF Antenna Pin**: The RF antenna pin on the RAK4200 module is the Pin 12.
6. **About the ADC Pin**: The ADC pin on the RAK4200 is assigned to Pin 3.
7. **About the GPIO pin**: The GPIO pins available on the RAK4200 module are Pin 3, Pin 6, Pin 9, and Pin 10.