

## Get start with RAK7201



**Notice:** at present, RAK7201 only supports the EU868/US915 band

### Where is the latest firmware binary?

If you want to get a pre-compiled firmware instead of compiling the source code by yourself, you can find the latest firmware on Github repository.

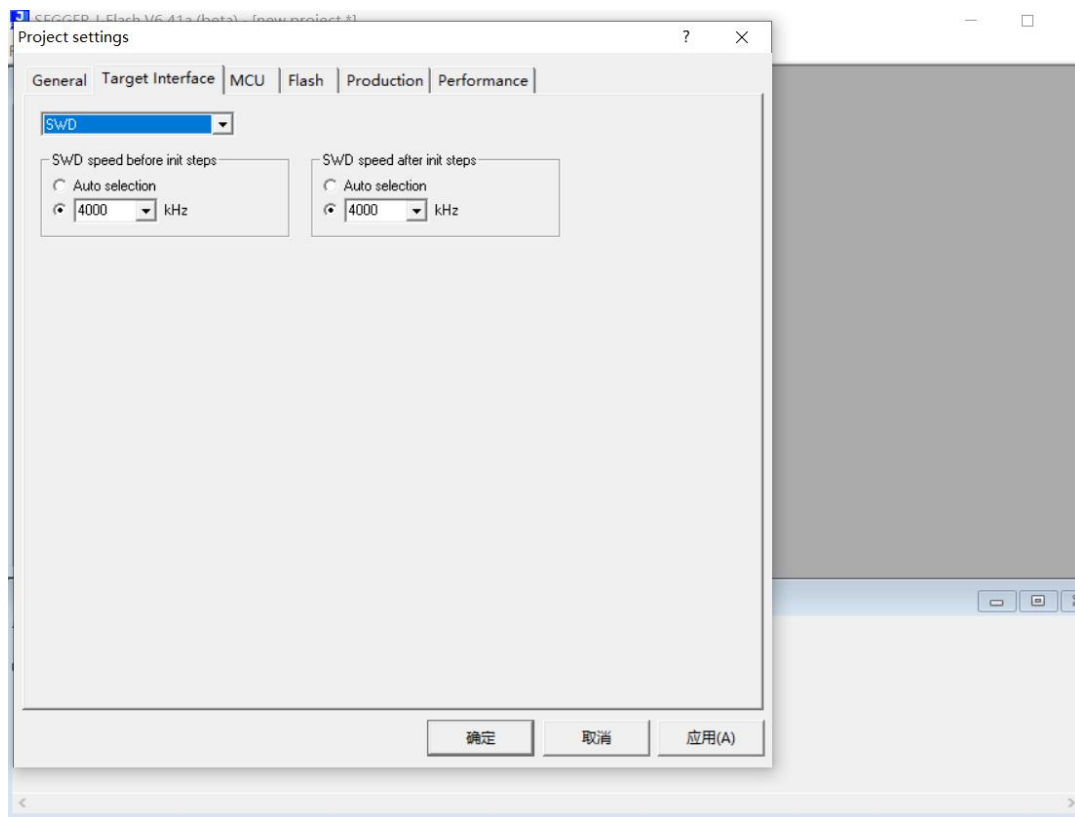
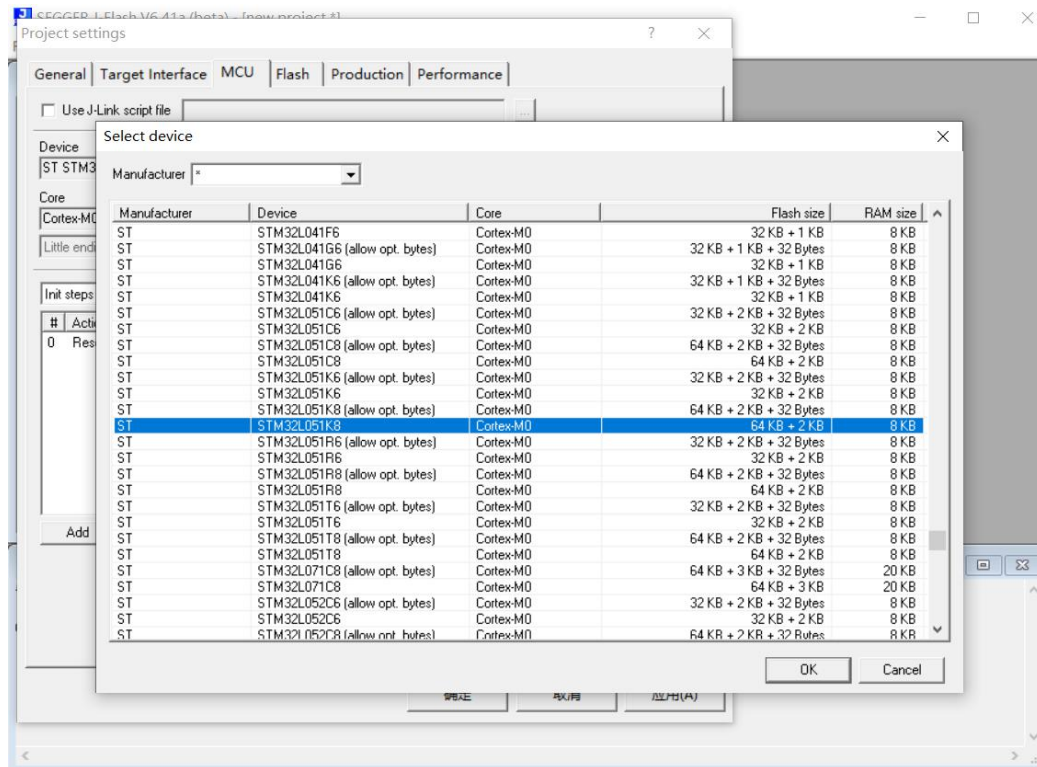
[https://github.com/RAKWireless/Get-start-with-RAK7201/blob/master/868-22dbButton\\_051.bin](https://github.com/RAKWireless/Get-start-with-RAK7201/blob/master/868-22dbButton_051.bin)

[https://github.com/RAKWireless/Get-start-with-RAK7201/blob/master/915-22dbButton\\_051.bin](https://github.com/RAKWireless/Get-start-with-RAK7201/blob/master/915-22dbButton_051.bin)

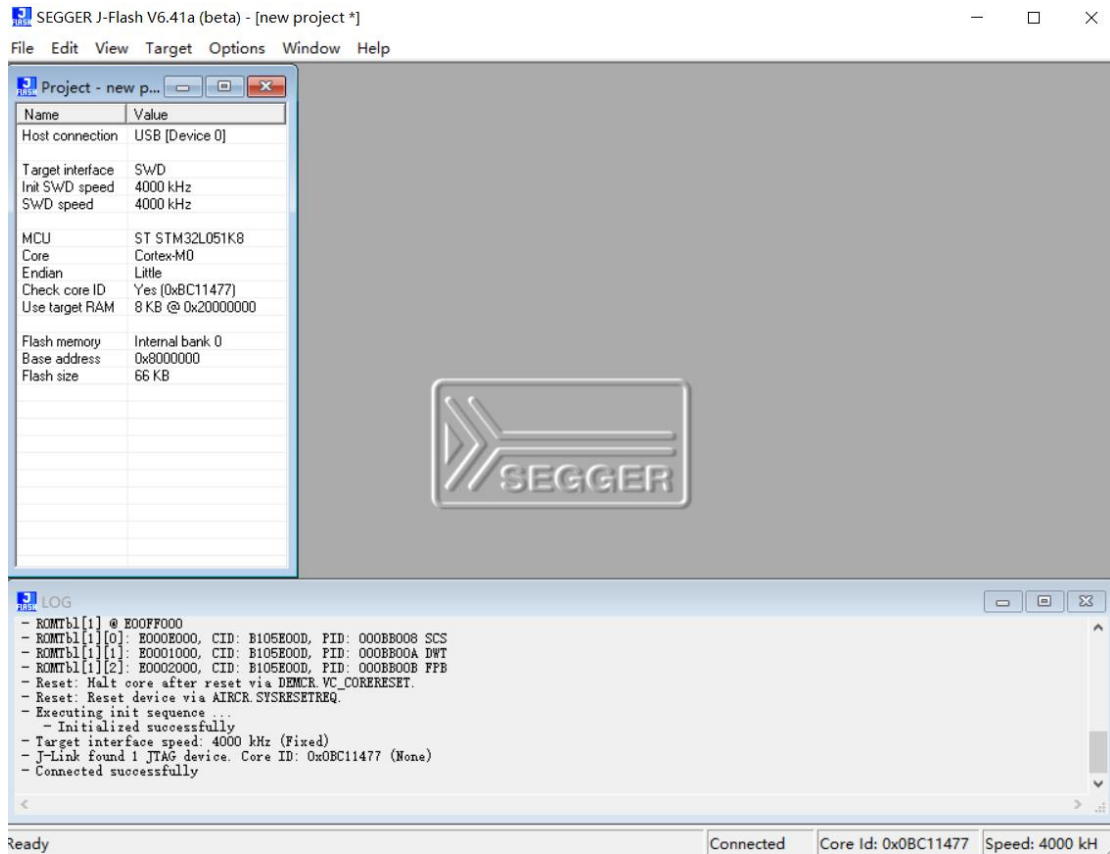
### How to flash the bootloader into RAK7201?

You can flash the bootloader for RAK7201 using J-Flash of J-Link as follow:

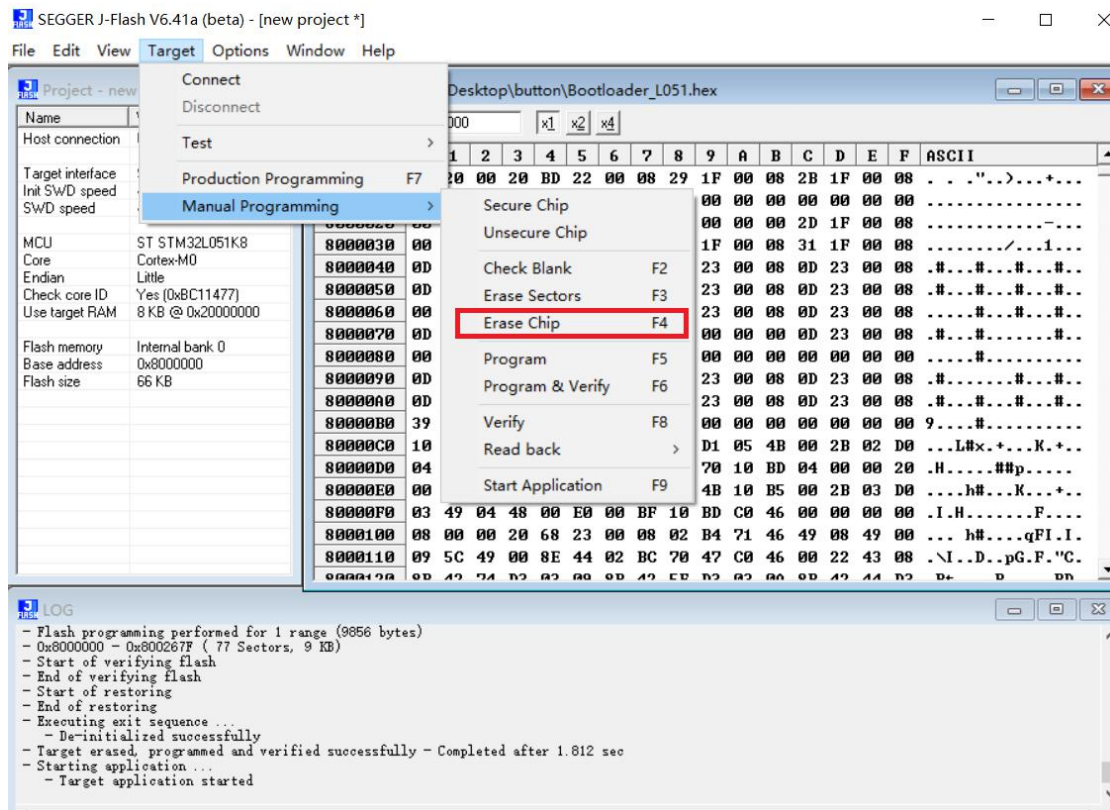
Open J-Flash and choose the correct MCU and SWD as the following pictures show:



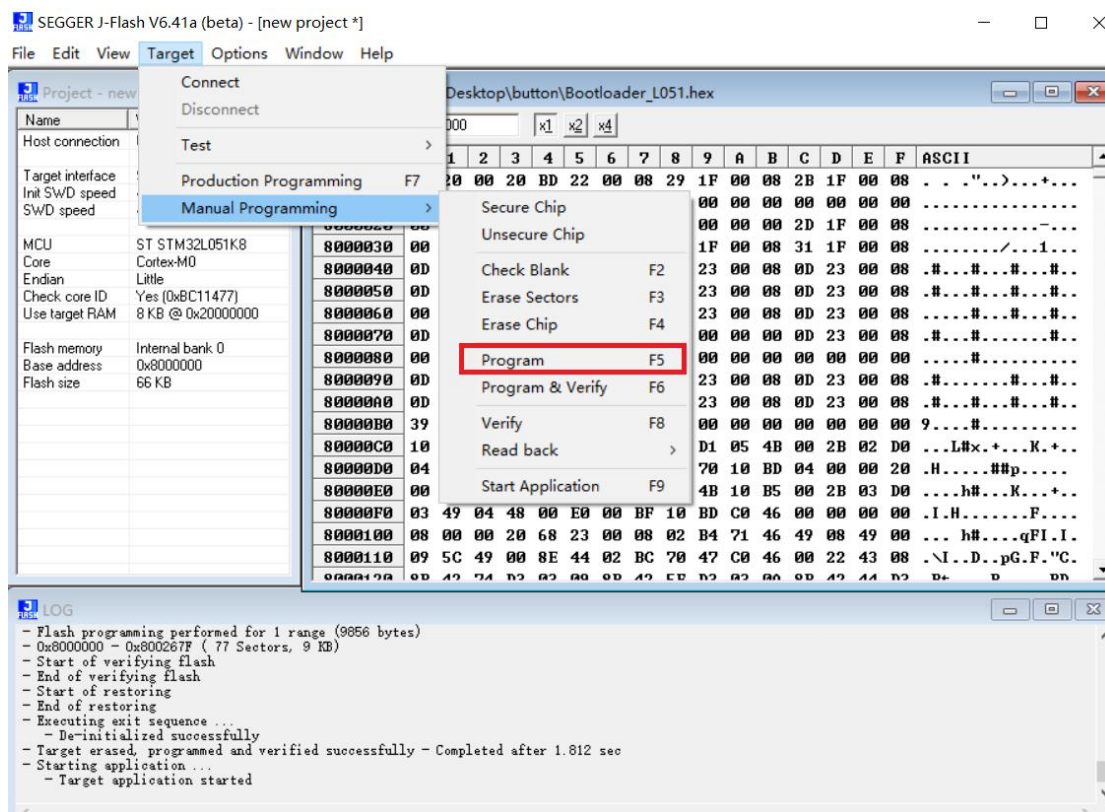
After "Connect" successfully, drag the bootloader file to the space of the following page:



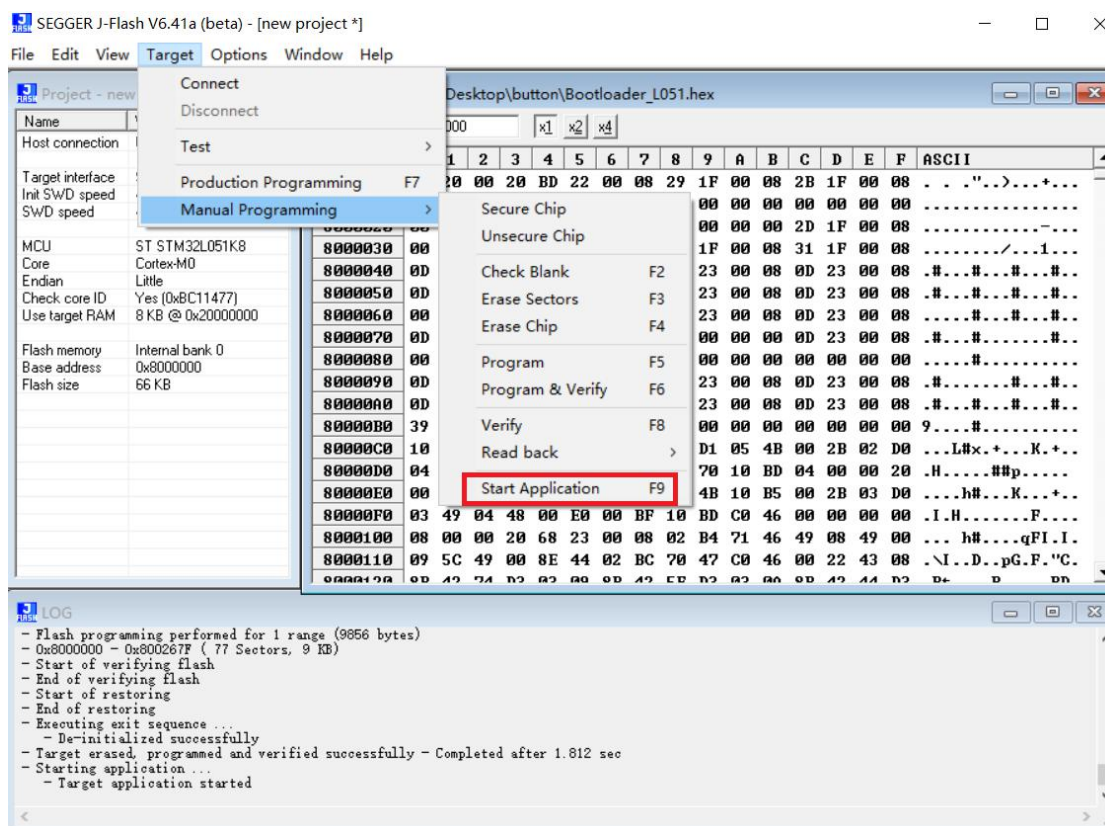
Now, "Erase Chip" firstly:



Secondly, "Program":



Then “Start Application”:



OK! We've flashed the bootloader for RAK7201 successfully!

## How to burn the update firmware into RAK7201?

If you have just burned the bootloader into RAK7201 according to the Burning Bootloader into the Device section, proceed to Step 2.

1. In case you have not just burned the bootloader, as instructed in the previous section you need to manually go into boot mode.

```

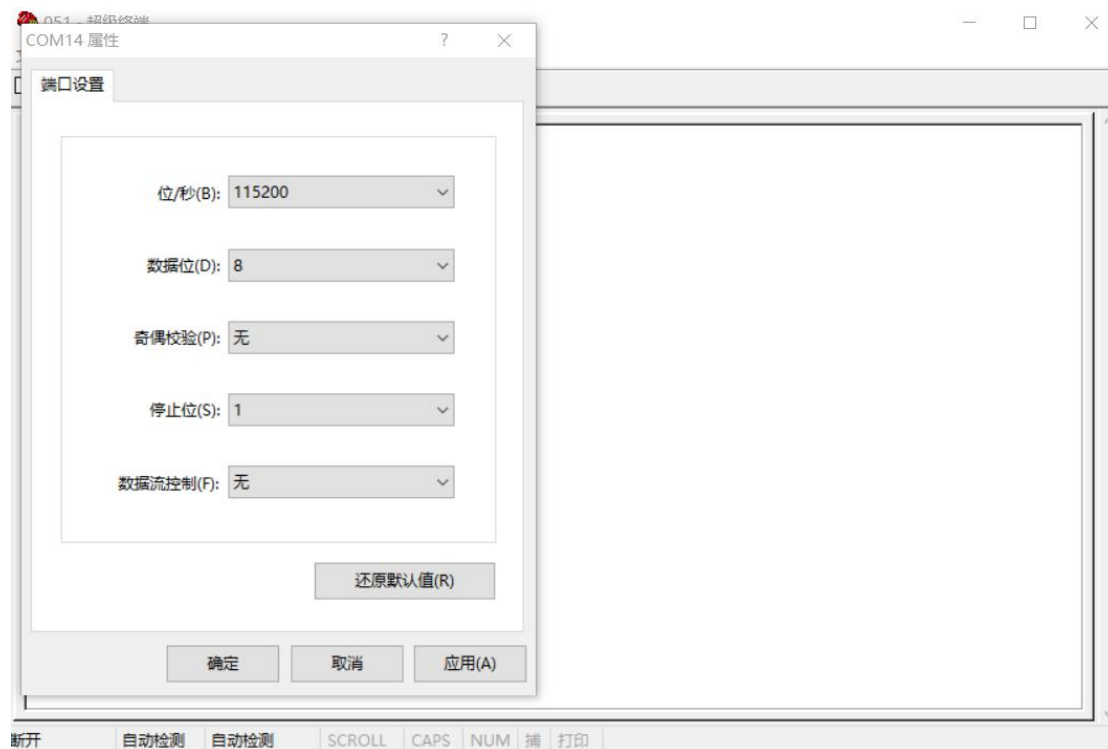
[15:08:32.926]发→◇at+boot
[15:08:32.938]收←◆Enter bootloader .....

===== Main Menu =====
Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====
    
```

2. Open the tool "Hyper Terminal" and configure as follow:

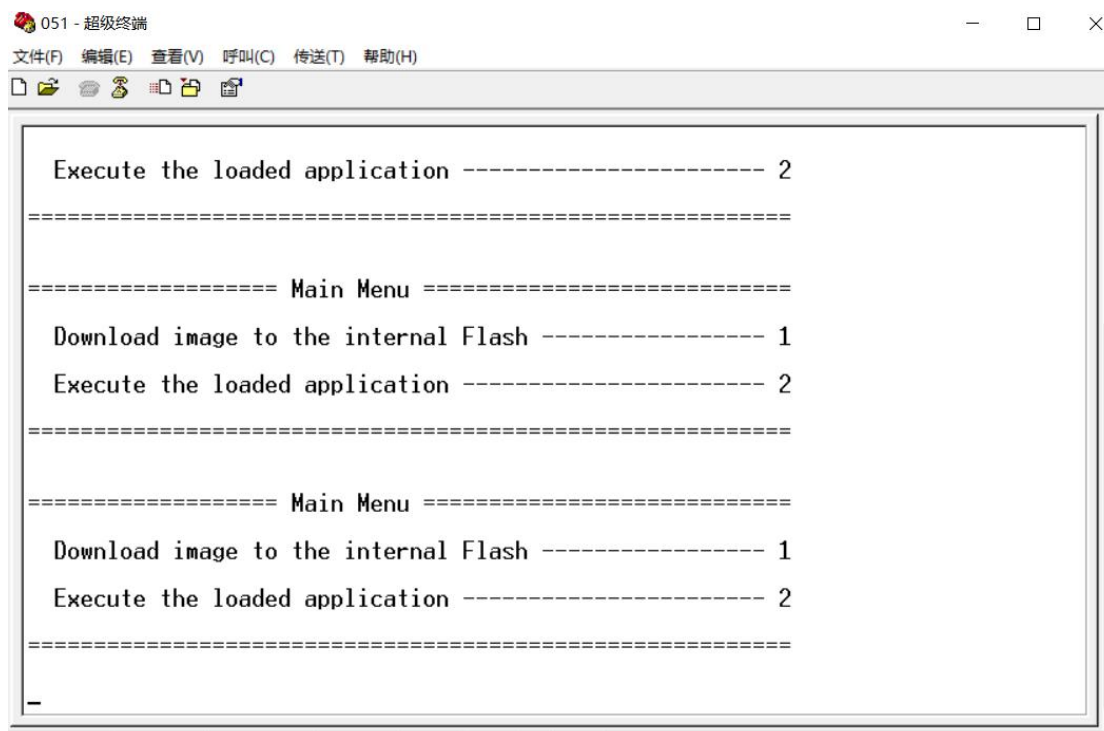
Tool:

[https://github.com/RAKWireless/Get-start-with-RAK7201/blob/master/super\\_terminal\\_v1.01.zip](https://github.com/RAKWireless/Get-start-with-RAK7201/blob/master/super_terminal_v1.01.zip)





3.If it connects successfully, you can see the following information in the Hyper Terminal page:



```

051 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

Execute the loaded application ----- 2
=====

===== Main Menu =====

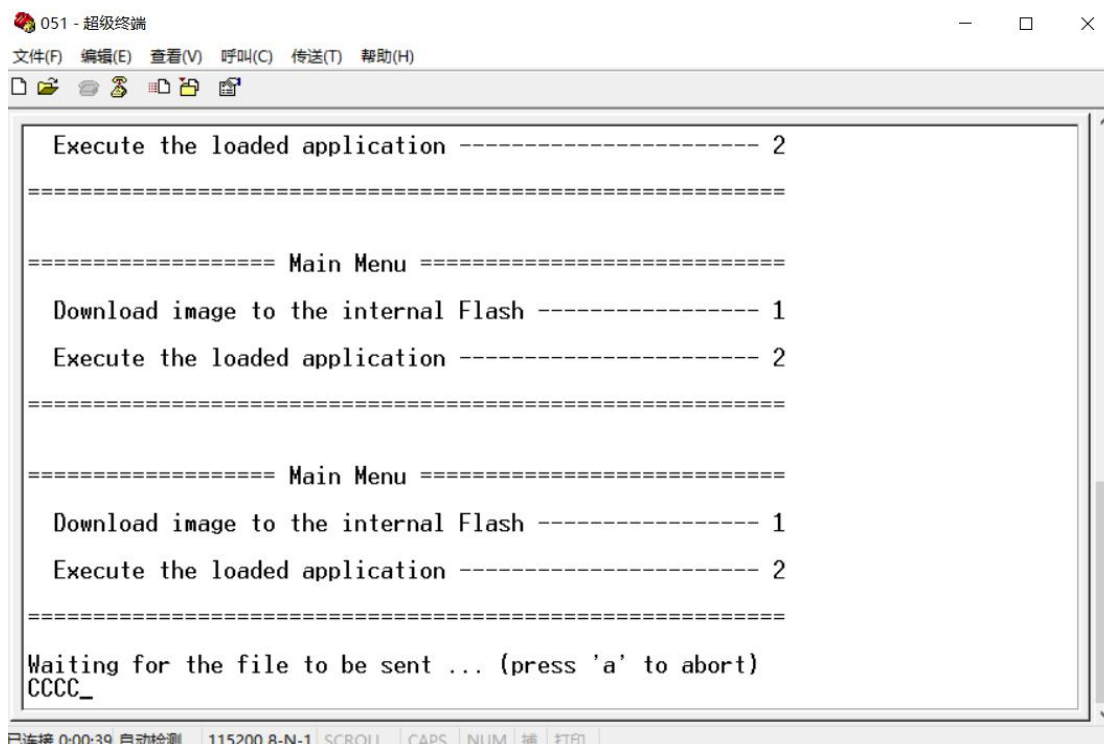
Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====

===== Main Menu =====

Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====
    
```

4.As you see, “1” means upgrading a firmware for RAK7201, and “2” means executing the firmware.

5.Now, type “1” to upgrade the firmware for RAK7201:



```

051 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

Execute the loaded application ----- 2
=====

===== Main Menu =====

Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====

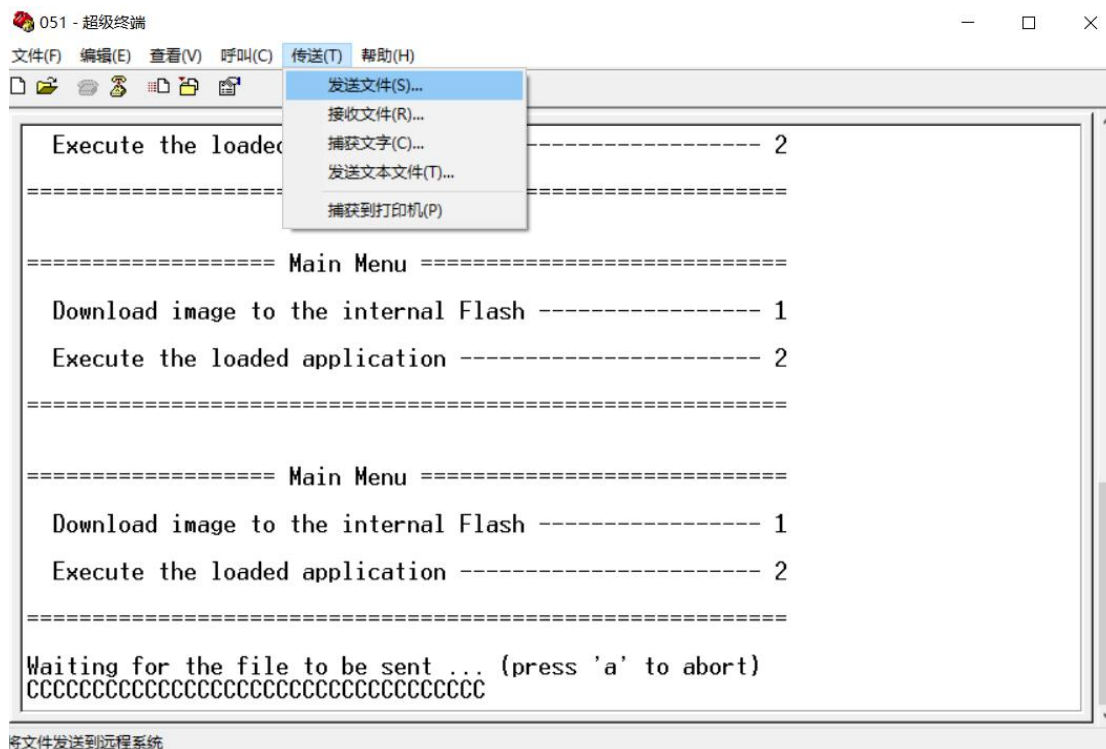
===== Main Menu =====

Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====

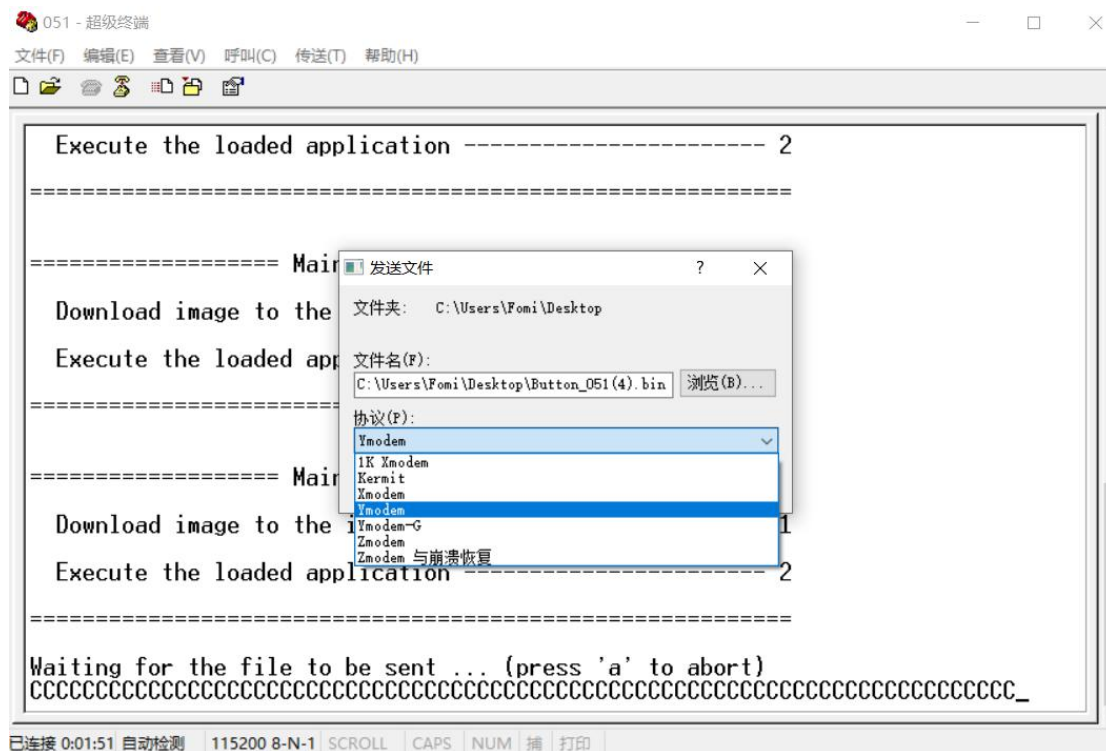
Waiting for the file to be sent ... (press 'a' to abort)
CCCC_
    
```

已连接 0:00:39 自动检测 115200 8-N-1 SCROLL CAPS NUM 捕 打印

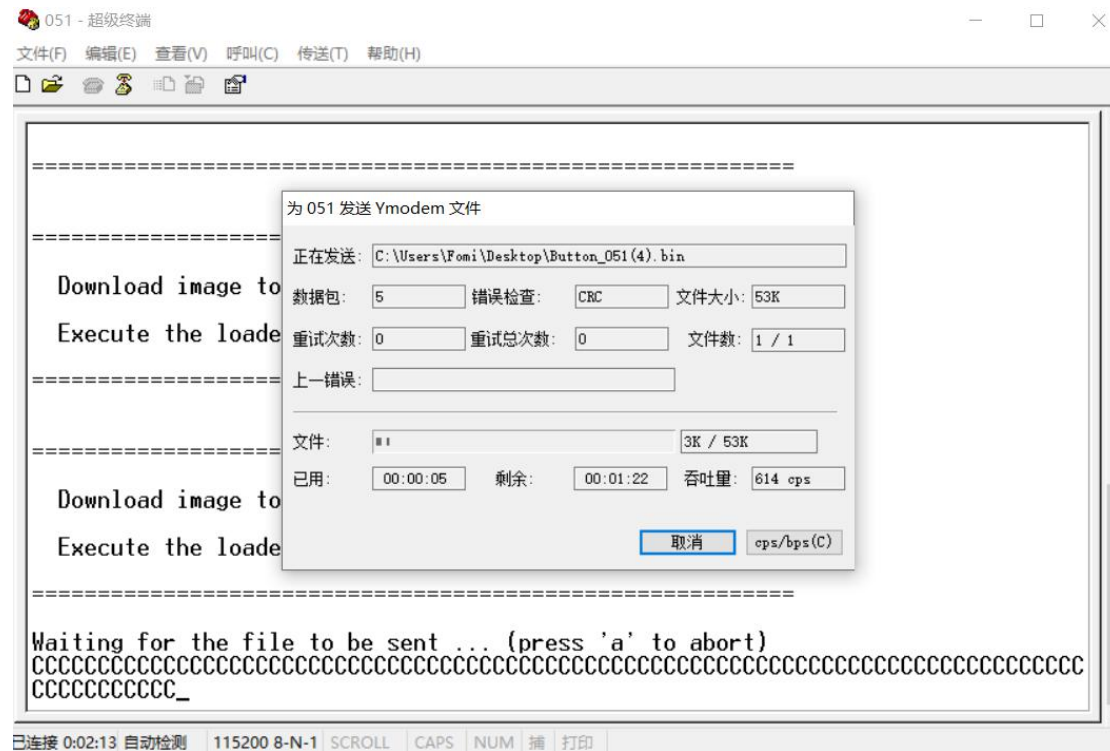
Choose the firmware file:



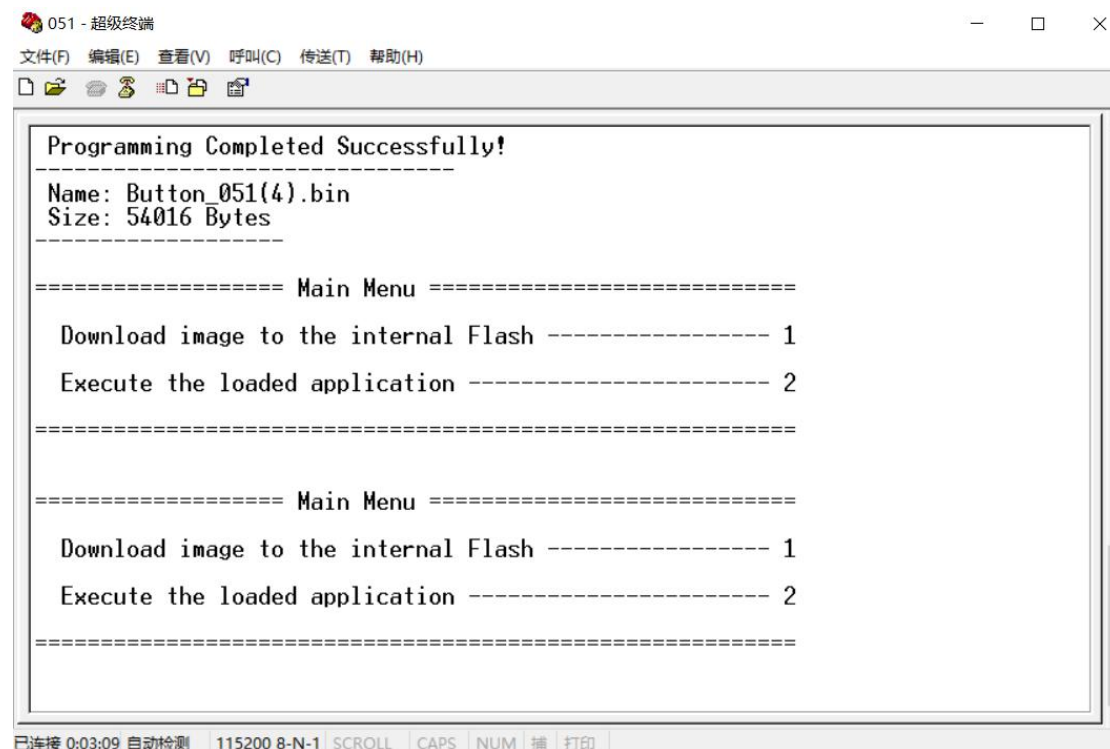
Please note that you should choose "Ymodem" as the following picture shows:



OK! Let's start to upgrade:

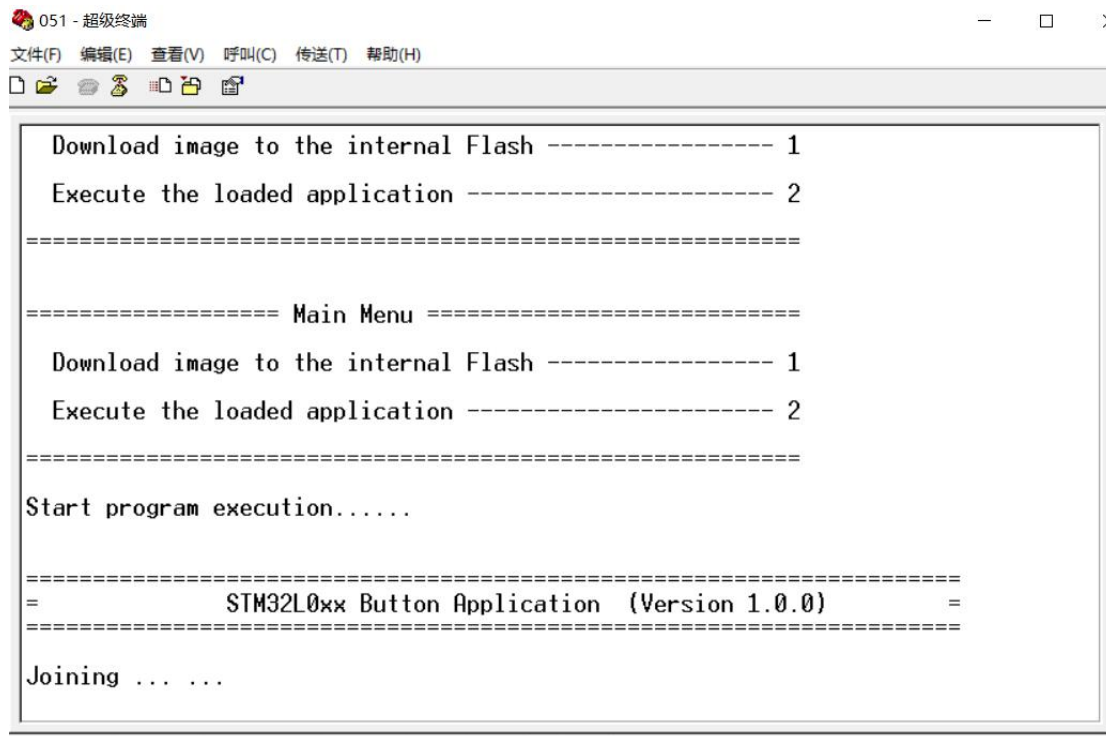


If we upgrade the firmware for RAK7201 successfully, you can see the following information:





Now, type "2" to execute the firmware we just upgraded:



```

051 - 超级终端
文件(F) 编辑(E) 查看(V) 呼叫(C) 传送(T) 帮助(H)

Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====

===== Main Menu =====

Download image to the internal Flash ----- 1
Execute the loaded application ----- 2
=====

Start program execution.....

=====
= STM32L0xx Button Application (Version 1.0.0) =
=====

Joining ... ..
    
```

Great! As you see, it works.

Now, disconnect with Hyper Terminal, then connect with a serial port tool.

## How to configure RAK7201?

You can configure RAK7201 by sending AT commands into it from a serial port tool running on your PC.

Please notice that the UART of RAK7201 is disable by default because of power consumption.

If you want to configure RAK7201 through UART, you can do as follow:

Press the key 1 for several seconds until you see the led of the key 4 lights up which means RAK7201 is working in configuration mode now, and you can configure it through UART.

After RAK7201 joins successfully, you can press any key to send message as the following picture shows:

```
[14:54:23.354]收← Lower_Power mode 0
[14:54:25.705]发→ ◇ at+set_config=lora:app_key:00000000000004740000000000000474
[14:54:25.818]收← ◆ ok
[14:54:26.448]发→ ◇ at+set_config=lora:app_eui:00000000000000474
[14:54:26.557]收← ◆ ok
[14:54:27.040]发→ ◇ at+set_config=lora:dev_eui:00000000000000474
[14:54:27.150]收← ◆ ok
```

The following list shows the AT commands:

AT Command	Description
at+boot	Let the device work in boot mode immediately.
at+reset	After set, the device will restart.
at+join	Start to join LoRa network.
at+get_config=lora:dev_eui	Get the device EUI for OTAA.
at+get_config=lora:app_eui	Get the application EUI for OTAA.
at+get_config=lora:app_key	Get the application key for OTAA.
at+get_config=lora:apps_key	Get the application session key for ABP.
at+get_config=lora:nwks_key	Get the network session key for ABP.
at+get_config=lora:dev_addr	Get the device address for ABP.
at+set_config=lora:join_mode:X	Set the join mode for LoRaWAN.  X definition: 0: OTAA, 1: ABP
at+set_config=lora:dev_eui:XXXX	Set the device EUI for OTAA.  XXXX definition: the device EUI, for example, 3534353165375300
at+set_config=lora:app_eui:XXXX	Set the application EUI for OTAA.  XXXX definition: the application EUI, for example, 70B3D57ED001A1E2
at+set_config=lora:app_key:XXXX	Set the application key for OTAA.  XXXX definition: the application key, for example, D9988A5F02D80FAB8BA5F453C4A2CD2B

at+set_config=lora:apps_key:XXXX	Set the application session key for ABP. XXXX definition: the application session key, for example, 573BD4DEC56BA4A9C462DF29E54B9BCE
at+set_config=lora:nwks_key:XXXX	Set the network session key for ABP. XXXX definition: the network session key, for example, C2AA51E61BA45F57045BF48249BC36F6
at+set_config=lora:dev_addr:XXXX	Set the device address for ABP. XXXX definition: the device address, for example, 2601116D

More information, please have a look at the next section <How to Connect with TTN?>.

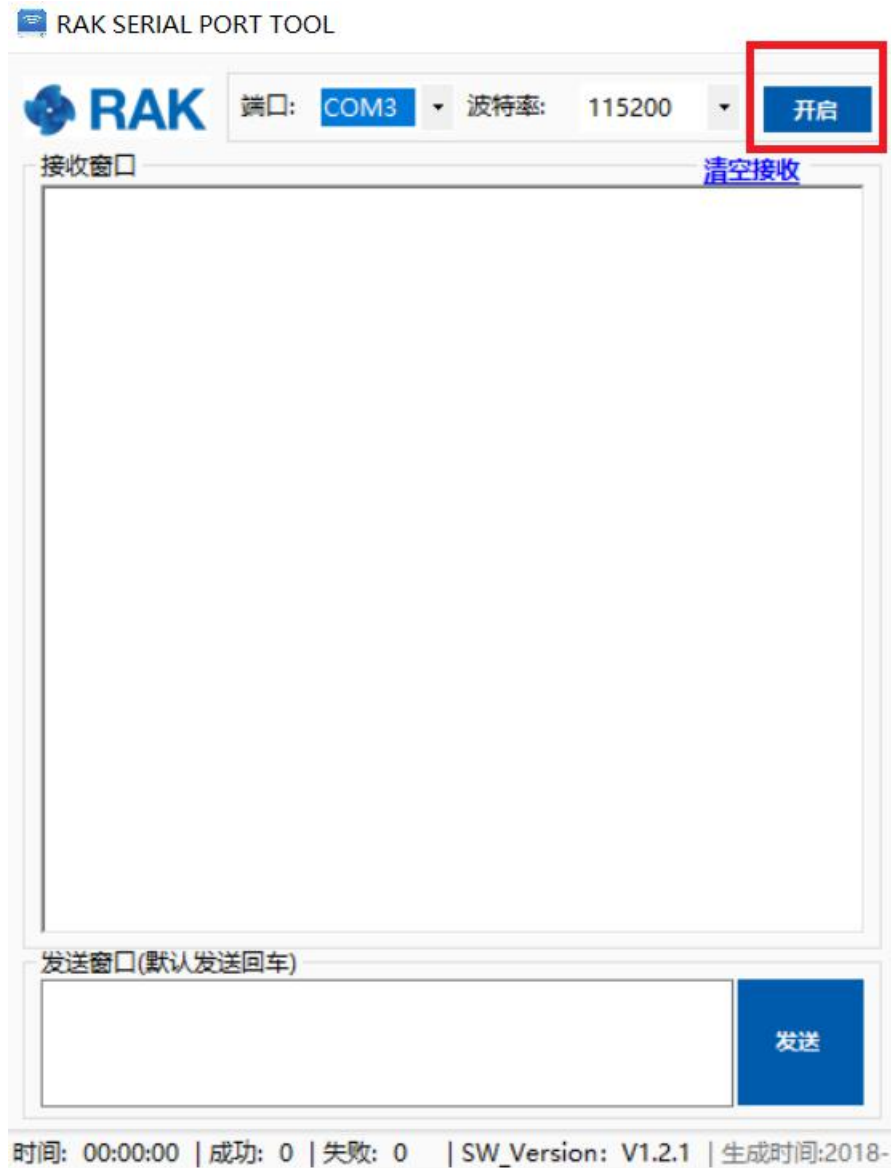
## How to connect with TTN?

In this section, we'll do some practice to show how to connect RAK7201 with TTN.

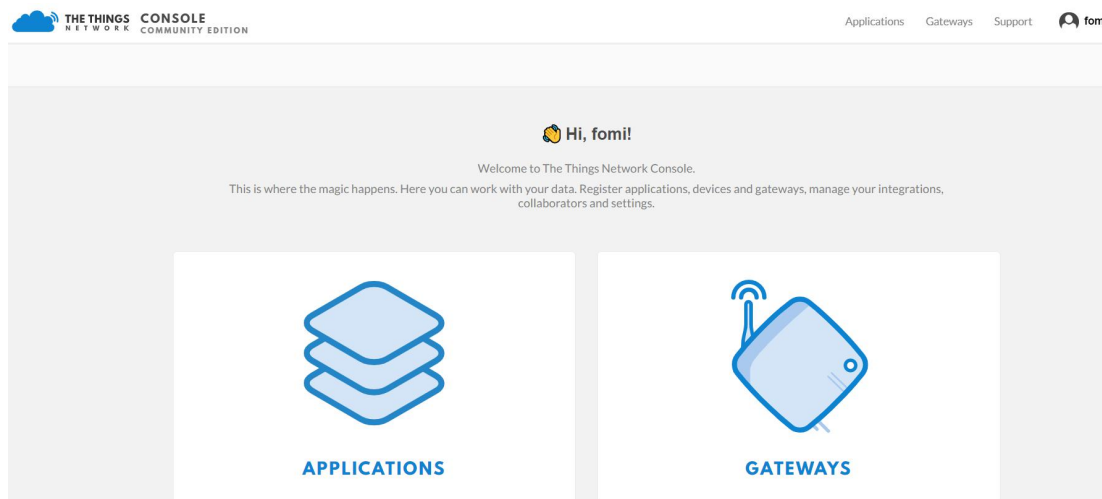
Firstly, open a serial port tool on your PC. I recommend you to use RAK serial port too:

[http://docs.rakwireless.com/en/LoRa/RAK811/Tools/RAK\\_SERIAL\\_PORT\\_TOOL\\_V1.2.1.zip](http://docs.rakwireless.com/en/LoRa/RAK811/Tools/RAK_SERIAL_PORT_TOOL_V1.2.1.zip)

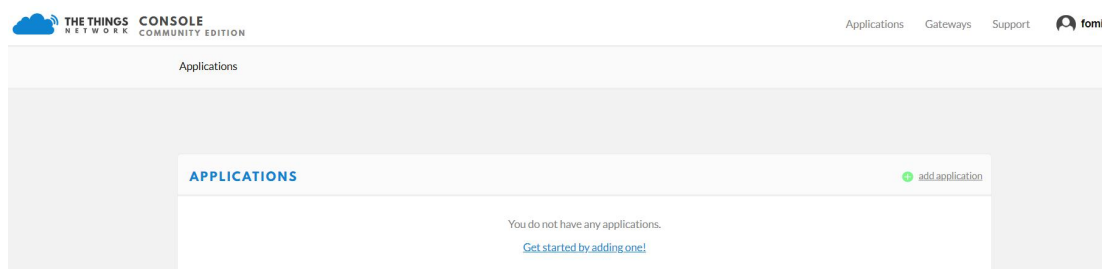
Open the serial port as the following picture shows:



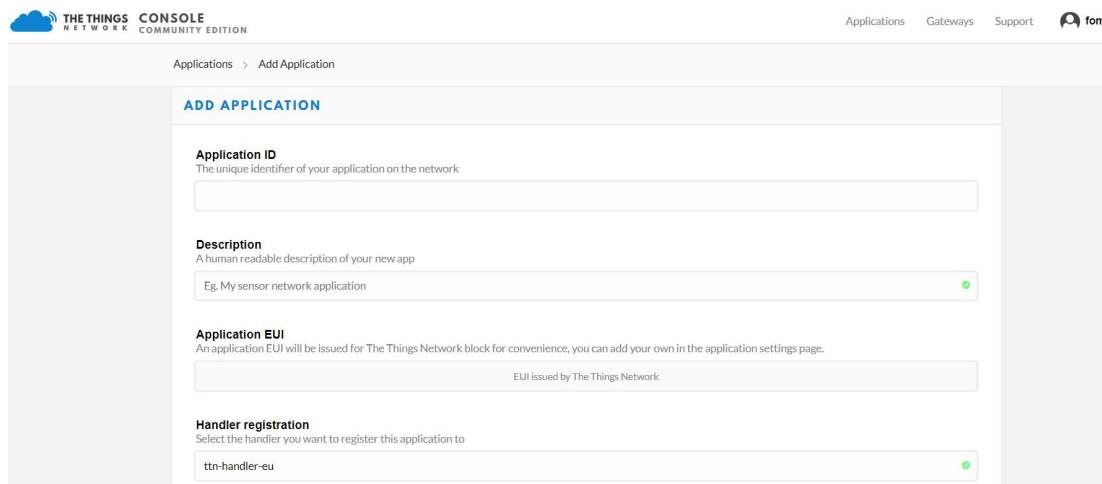
Open the link <https://www.thethingsnetwork.org/> and login, then open the “Console” page from the right corner at the top:



Press “APPLICATIONS”:



“add application”:




Fill in the correct contents.

Please note that the content you fill in “Application ID” item should be in low case, and it must be the unique ID on TTN network.



THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications Gateways Support 

Applications > Add Application

### ADD APPLICATION

**Application ID**  
The unique identifier of your application on the network

lorabuttontest123

**Description**  
A human readable description of your new app

my lora button test

**Application EUI**  
An application EUI will be issued for The Things Network block for convenience, you can add your own in the application settings page.


EUI issued by The Things Network

**Handler registration**  
Select the handler you want to register this application to

ttn-handler-eu

Then press the “Add application” button at the bottom of this page, and you can see the following page:

THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications Gateways Support 

Applications > lorabuttontest123

Overview Devices Payload Formats Integrations Data Settings

### APPLICATION OVERVIEW

[documentation](#)

**Application ID** lorabuttontest123

**Description** my lora button test

**Created** 8 seconds ago

**Handler** ttn-handler-eu (current handler)

### APPLICATION EUIS

[manage euis](#)

<> 70 B3 D5 7E D0 01 C1 CF

At the middle of this page, you can find the box named “DEVICES”:

**DEVICES** [register device](#) [manage devices](#)

Error while fetching devices  
Error: App 01a7c01e4db9b44d not registered to a handler

Just “register device”:

THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications Gateways Support fomi

Applications > 01a7c01e4db9b44d > Devices

**REGISTER DEVICE** [bulk import devices](#)

**Device ID**  
This is the unique identifier for the device in this app. The device ID will be immutable.

**Device EUI**  
The device EUI is the unique identifier for this device on the network. You can change the EUI later.

0 bytes

**App Key**  
The App Key will be used to secure the communication between you device and the network.

this field will be generated

**App EUI**

70 B3 D5 7E D0 01 C1 CB

Please note that when you fill this value in “Device ID”, it must be in low case.

Also, you can click the following icon and “Device EUI” will be generated automatically in the next step:

THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications Gateways Support fomi

Applications > lorabuttontest123 > Devices

Overview **Devices** Payload Formats Integrations Data Settings

**REGISTER DEVICE** [bulk import devices](#)

**Device ID**  
This is the unique identifier for the device in this app. The device ID will be immutable.

01a7c01e4db9b44d

**Device EUI**  
The device EUI is the unique identifier for this device on the network. You can change the EUI later.

0 bytes

**App Key**  
The App Key will be used to secure the communication between you device and the network.

this field will be generated

**App EUI**

70 B3 D5 7E D0 01 C1 CF

The following picture shows the final page:

THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications Gateways Support fomi

Applications > lorabuttontest123 > Devices

**REGISTER DEVICE** [bulk import devices](#)

**Device ID**  
This is the unique identifier for the device in this app. The device ID will be immutable.

01a7c01e4db9b44d

**Device EUI**  
The device EUI is the unique identifier for this device on the network. You can change the EUI later.

this field will be generated

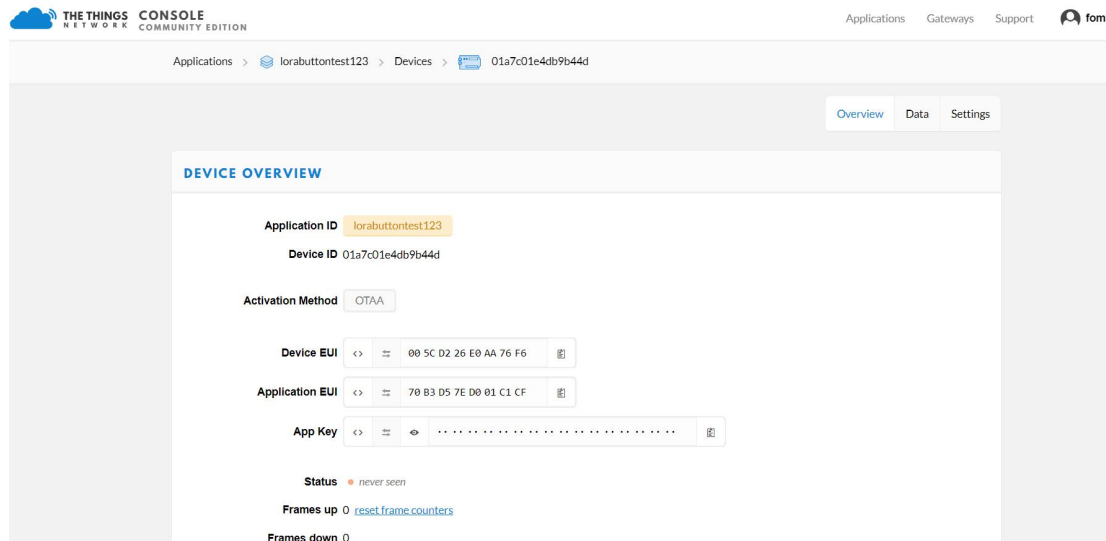
**App Key**  
The App Key will be used to secure the communication between you device and the network.

this field will be generated

**App EUI**

70 B3 D5 7E D0 01 C1 CF

Then press the “Register” button at the bottom of this page to finish.



As you see, the default activation method is OTAA.

OK! Now, let's join in OTAA mode and EU868 frequency for example!

Set the join mode to OTAA:

Set the Device EUI:

Set the Application EUI:

Set the Application Key:

Join in OTAA mode:

Join successfully!

Let's press key 1 on RAK7201:

```

[15:42:34.595]发→◇at+set_config=lorajoin_mode:0
[15:42:34.700]收←◆ok
[15:42:43.558]发→◇at+set_config=loradev_eui:
[15:42:43.664]收←◆ok
[15:42:44.347]发→◇at+set_config=lorapp_eui:
[15:42:44.457]收←◆ok
[15:42:45.027]发→◇at+set_config=lorapp_key:
[15:42:45.141]收←◆ok
[15:42:48.835]发→◇at+join
[15:43:40.108]收←◆Joining ...
[15:43:45.573]收←◆Status is OK, node has joined the network
[15:43:48.617]收←◆KEY1 Fall
[15:43:50.400]收←◆MCPS confirmed succseeful
[15:43:50.782]收←◆mcpsIndication->Rssi      -54

```

Now, you can see the status on TTN is active.

## DEVICE OVERVIEW

Application ID 868-cs

Device ID 47aac86800430028

Description ....

Activation Method OTAA

Device EUI <> ⇄ 47 AA C8 68 00 43 00 28

Application EUI <> ⇄ 70 B3 D5 7E D0 01 A1 E2

App Key <> ⇄ 🔍 .....

Device Address <> ⇄ 26 01 29 BD


Network Session Key <> ⇄ 🔍 .....


App Session Key <> ⇄ 🔍 .....

Status ● 11 minutes ago

Frames up 0 [reset frame counters](#)

Frames down 1


**THE THINGS NETWORK** CONSOLE  
COMMUNITY EDITION

Applications Gateways Support 

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d

Device ID 01a7c01e4db9b44d

Activation Method OTAA

Device EUI <> ⇄ 00 5C D2 26 E0 AA 76 F6

Application EUI <> ⇄ 70 B3 D5 7E D0 01 C1 CF

App Key <> ⇄ 🔍 .....

Device Address <> ⇄ 26 01 29 D7

Network Session Key <> ⇄ 🔍 .....

App Session Key <> ⇄ 🔍 .....

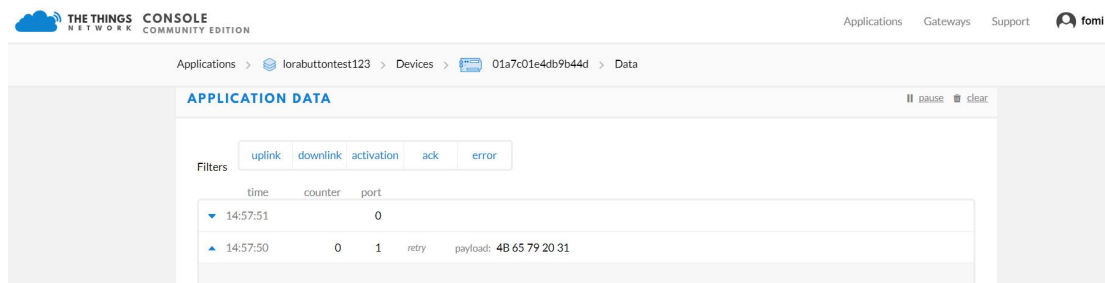
Status ● 1 minute ago

Frames up 0 [reset frame counters](#)

Frames down 0

It will send some data to TTN automatically:





THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d > Data

**APPLICATION DATA**

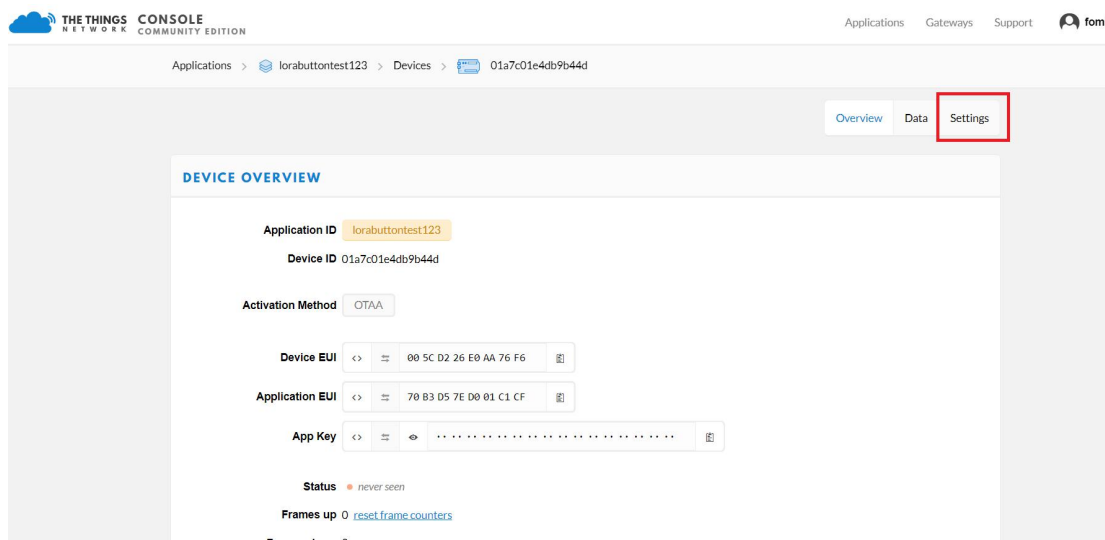
Filters: uplink downlink activation ack error

time	counter	port	
14:57:51	0		
14:57:50	0	1	retry payload: 4B 65 79 20 31

That's all about OTAA.

Now, let's try to join with TTN in ABP mode:

Firstly, click the "Settings" item as follow:



THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d

Overview Data **Settings**

**DEVICE OVERVIEW**

Application ID lorabuttontest123

Device ID 01a7c01e4db9b44d

Activation Method OTAA

Device EUI <> 00 5C D2 26 E0 AA 76 F6

Application EUI <> 70 B3 D5 7E D0 01 C1 CF

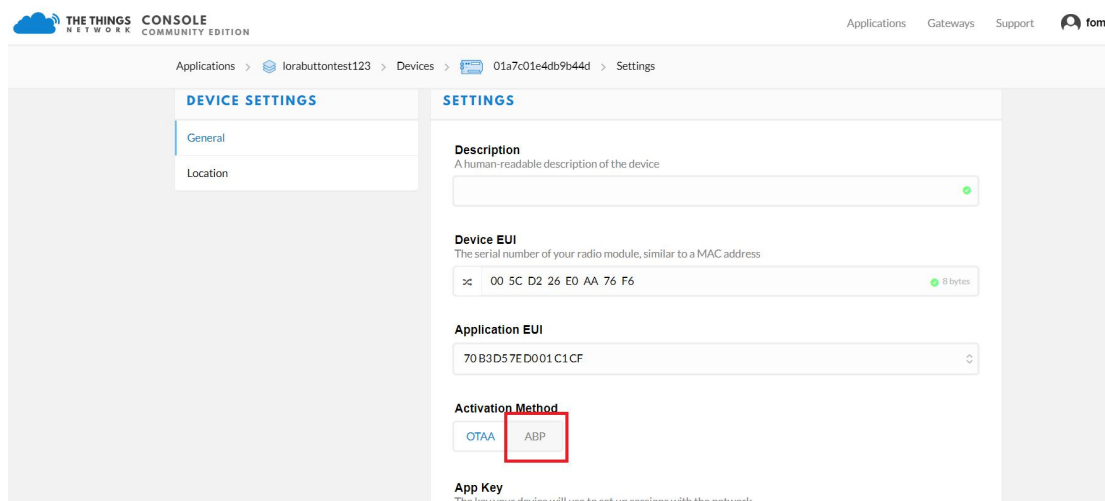
App Key <> .....

Status ● never seen

Frames up 0 [reset frame counters](#)

Frames down 0

Click "ABP":



THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d > Settings

**DEVICE SETTINGS**

General Location

**SETTINGS**

**Description**  
A human-readable description of the device

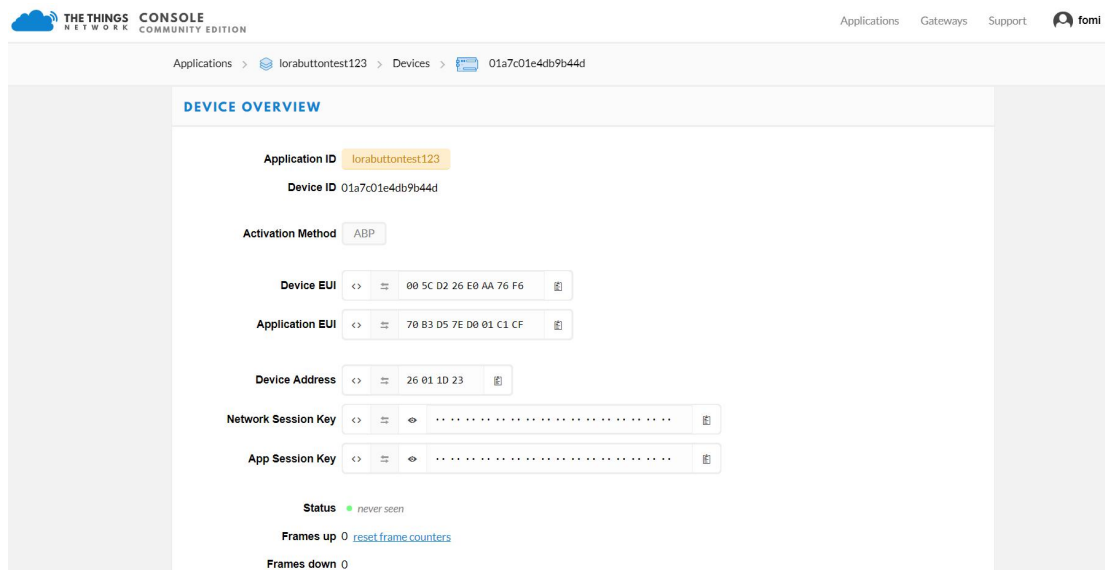
**Device EUI**  
The serial number of your radio module, similar to a MAC address  
00 5C D2 26 E0 AA 76 F6

**Application EUI**  
70 B3 D5 7E D0 01 C1 CF

**Activation Method**  
OTAA **ABP**

**App Key**  
The key your device will use to set up sessions with the network

Then press the “Save” button at the bottom of this page, then you can see the following page:



THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d

### DEVICE OVERVIEW

Application ID **lorabuttontest123**

Device ID 01a7c01e4db9b44d

Activation Method **ABP**

Device EUI <> 00 5C D2 26 E0 AA 76 F6

Application EUI <> 70 B3 D5 7E D0 01 C1 CF

Device Address <> 26 01 1D 23

Network Session Key <> .....

App Session Key <> .....

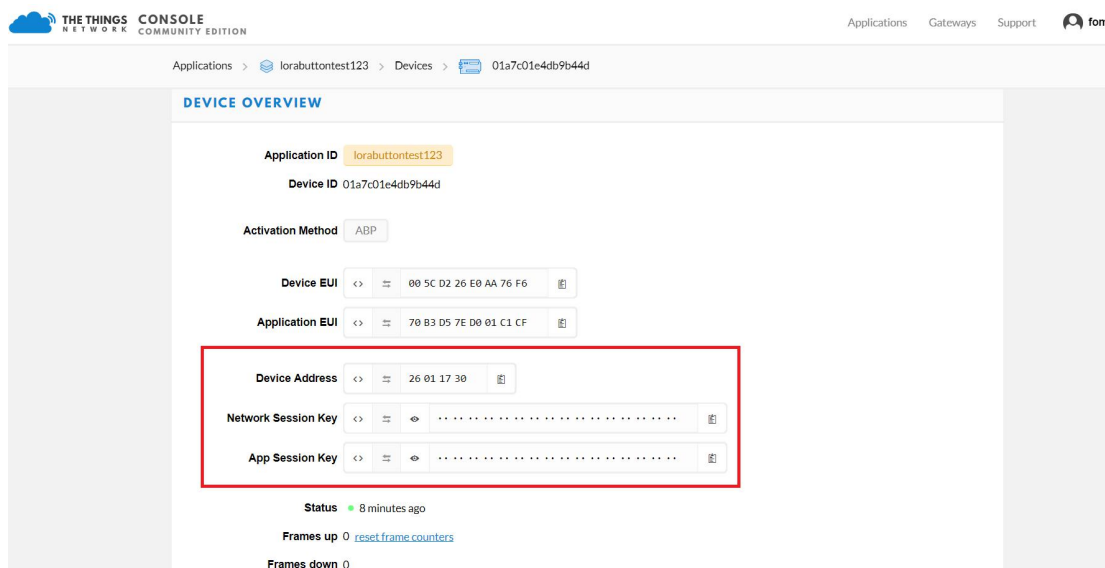
Status ● never seen

Frames up 0 [reset frame counters](#)

Frames down 0

Now, the activation method is ABP.

Then, let's configure RAK7201 using the following contents:



THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d

### DEVICE OVERVIEW

Application ID **lorabuttontest123**

Device ID 01a7c01e4db9b44d

Activation Method **ABP**

Device EUI <> 00 5C D2 26 E0 AA 76 F6

Application EUI <> 70 B3 D5 7E D0 01 C1 CF

Device Address <> 26 01 17 30

Network Session Key <> .....

App Session Key <> .....

Status ● 8 minutes ago

Frames up 0 [reset frame counters](#)

Frames down 0

At first, set the join mode to ABP:

Set the device address:

Set the APP session key:

Set the network session key:

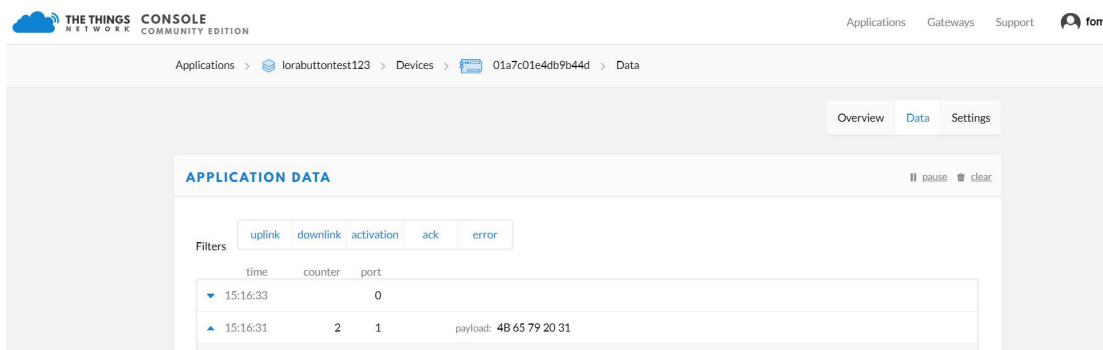
Join in ABP mode:

That's OK. Now, let's press key 3 on RAK7201:

```

[15:40:03.444]发→◇at+set_config=lorawan:join_mode:1
[15:40:03.552]收←◆ok
[15:40:10.155]发→◇at+set_config=lorawan:dev_addr:
[15:40:10.265]收←◆ok
[15:40:10.732]发→◇at+set_config=lorawan:apps_key:
[15:40:10.843]收←◆ok
[15:40:11.364]发→◇at+set_config=lorawan:nwks_key:
[15:40:11.475]收←◆ok
[15:40:14.780]发→◇at+join
[15:40:14.782]收←◆ok
[15:40:17.704]收←◆KEY3 Fall
[15:40:19.224]收←◆MCPS confirmed succseeful
[15:40:19.684]收←◆mcpsIndication->Rssi      -52
    
```

You can see the data on TTN:



The screenshot shows the 'Data' tab of the TTN console for application 'lorabuttontest123' and device '01a7c01e4db9b44d'. The 'APPLICATION DATA' table is displayed with filters for 'uplink', 'downlink', 'activation', 'ack', and 'error'. The table has columns for 'time', 'counter', 'port', and 'payload'. Two data points are visible:

time	counter	port	payload
15:16:33	0		
15:16:31	2	1	payload: 4B 65 79 20 31

That's all about ABP mode.

## Button additional function description

1. Long press key 1: the red light is on at 4 keys, and the configuration mode is on.
2. Long press key 2: 4 blue lights flash twice and the board is restarted.