

# Get Start with RAK7200 Tracker Device

Version V1.0 | July 2019

**[www.RAKwireless.com](http://www.RAKwireless.com)**

*Visit our website for more document.*

36 PAGES

## Table of Contents

1. Where is the latest firmware file?.....	3
2. How to burn the update firmware into RAK7200?.....	3
3. How to configure RAK7200?.....	9
4. How to connect with TTN?.....	12
4.1 Join in OTAA mode.....	17
4.2 Join in ABP mode.....	20
5. How to connect with LoRaServer?.....	21
5.1 Join in OTAA mode.....	24
5.2 Join in ABP mode.....	30
6. Revision History.....	36
7. Document Summary.....	36

## 1. Where is the latest firmware file?

If you want to get a pre-compiled firmware instead of compiling the source code by yourself, you can find the latest firmware on RAK website after it is released.

## 2. How to burn the update firmware into RAK7200?

Firstly, please install the “STM32CubeProgrammer” tool on your Windows PC. You can download it from here:

[https://www.st.com/content/st\\_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-programmers/stm32cubeprog.html#overview](https://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-programmers/stm32cubeprog.html#overview)

Secondly, connect RAK7200 with your PC's USB interface as follow:

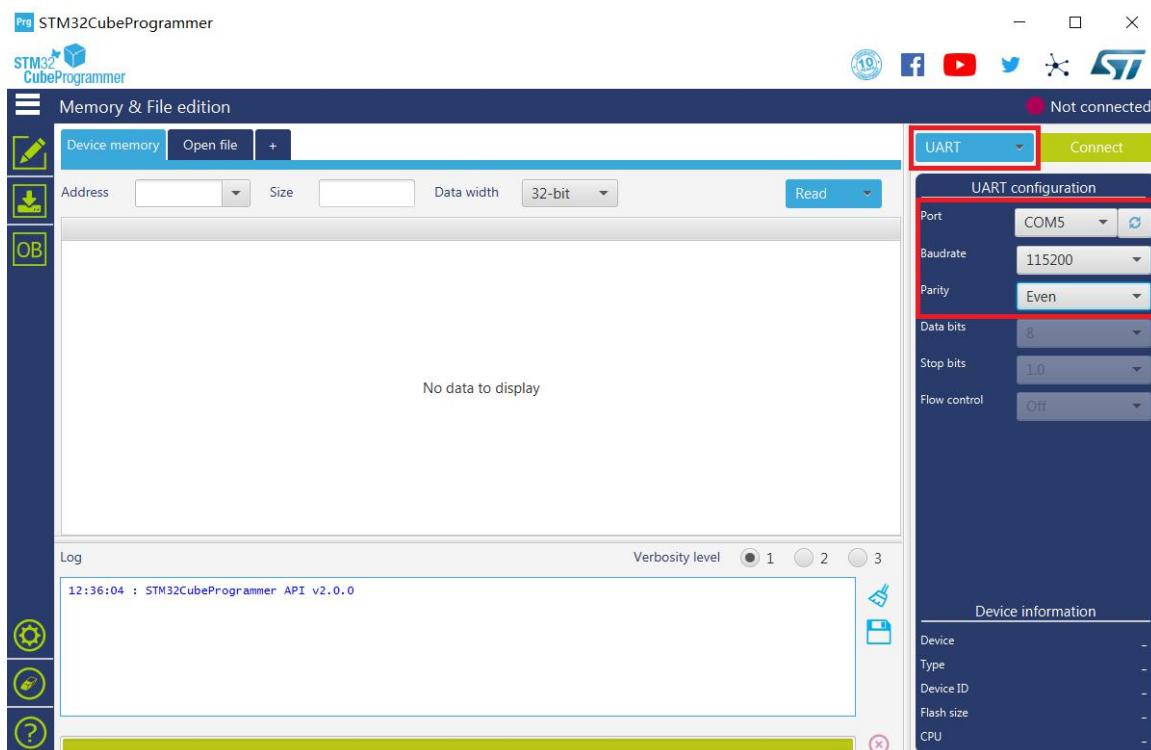


Now, you need to let RAK7200 work in boot mode. You can do it as follow:

1. Hold down the BOOT0 button, then press the Reset button;
2. Loosen the Reset button, then Loosen the BOOT0 button.

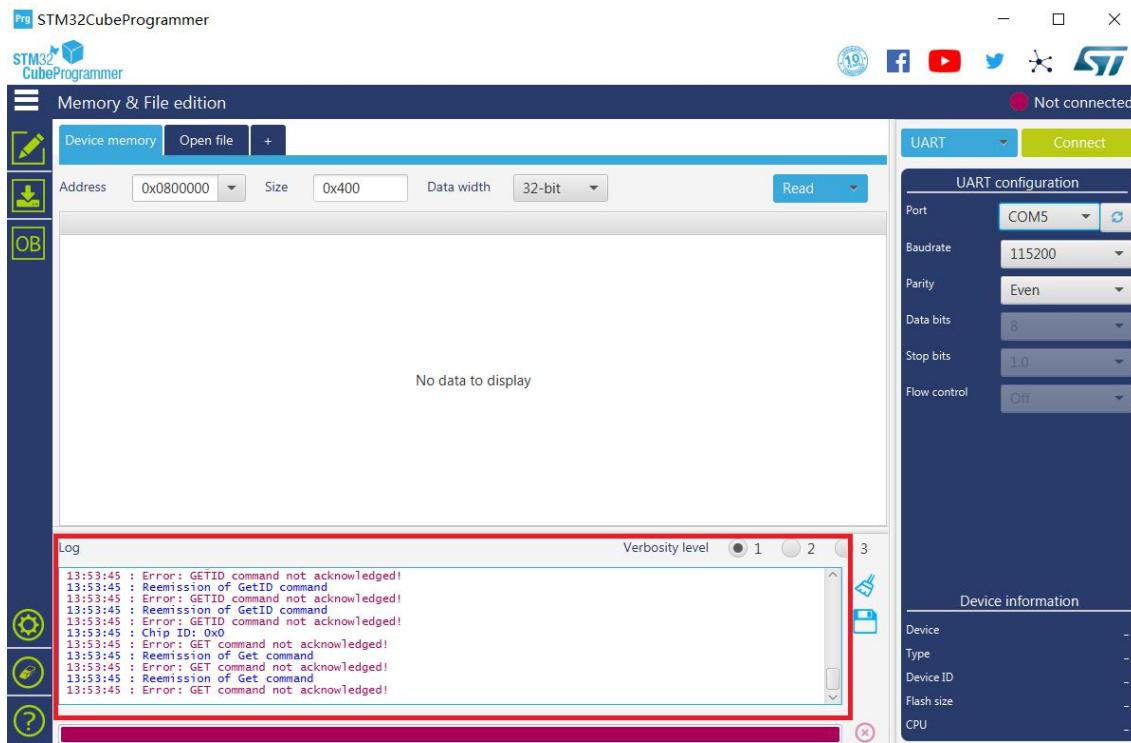


Open the “STM32CubeProgrammer” tool, and select UART type, then configure the Port, Baudrate, and Parity as the following picture shows:



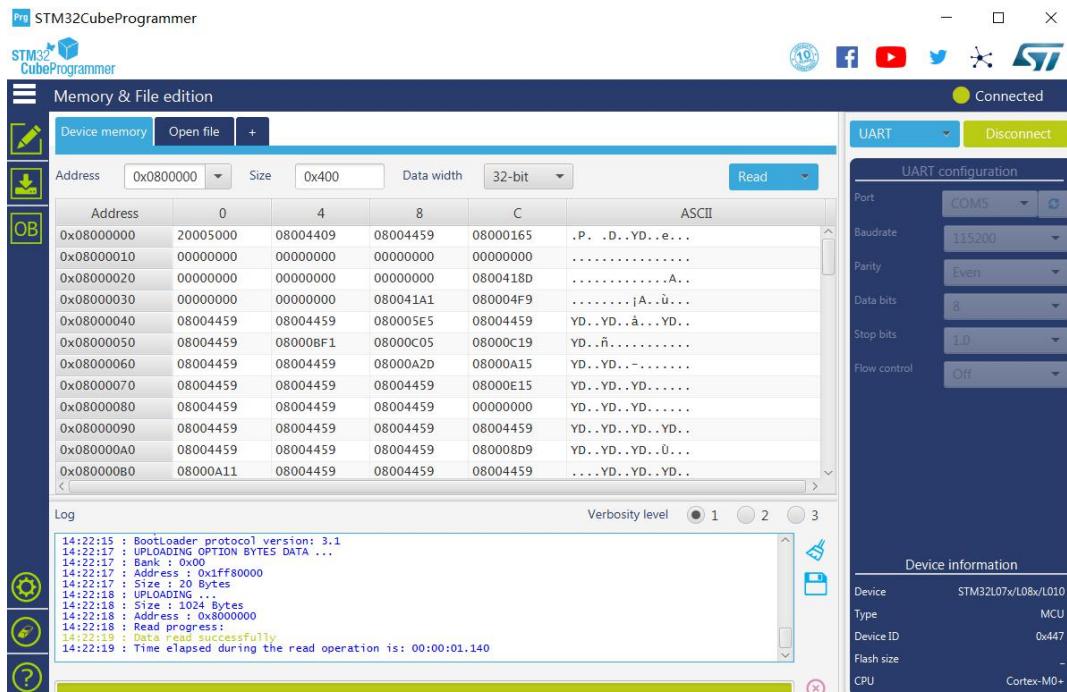
Note: It is COM5 on my computer, and it may be another COM port on your computer.

Then press “Connect” button at the top right corner. If there are some errors in the Log box as follow:



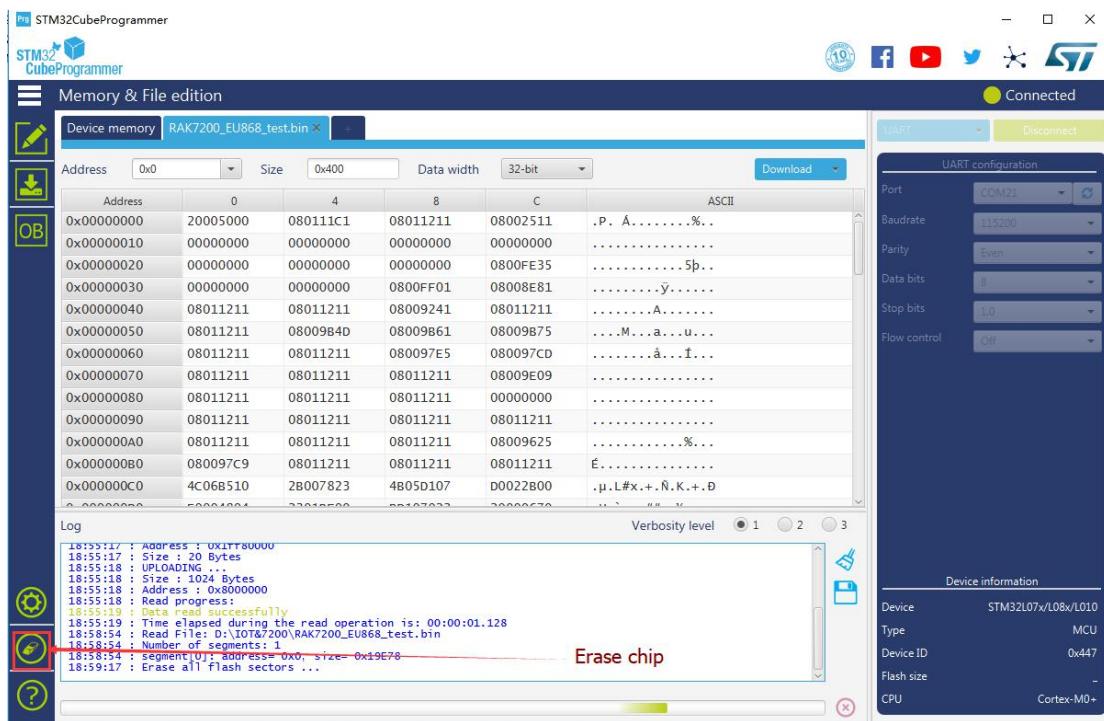
It means that RAK7200 don't work in BOOT mode. You should let it work in boot mode firstly.

The correct Log you should see is the information like the following picture shows:

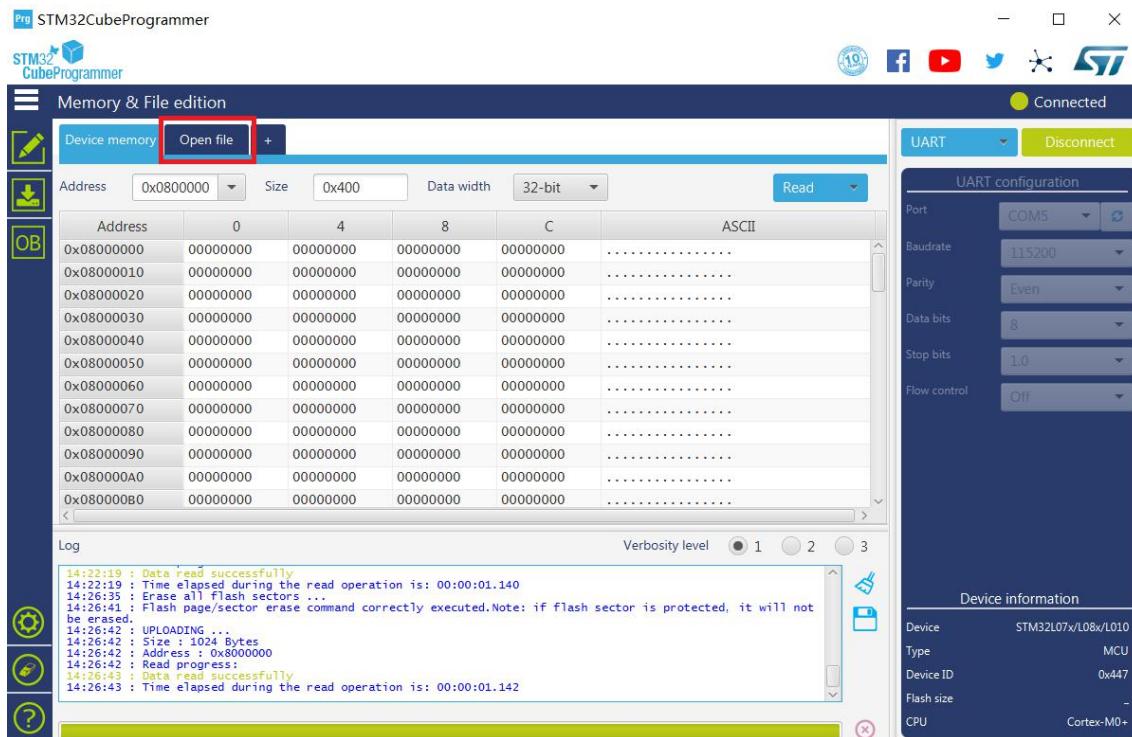


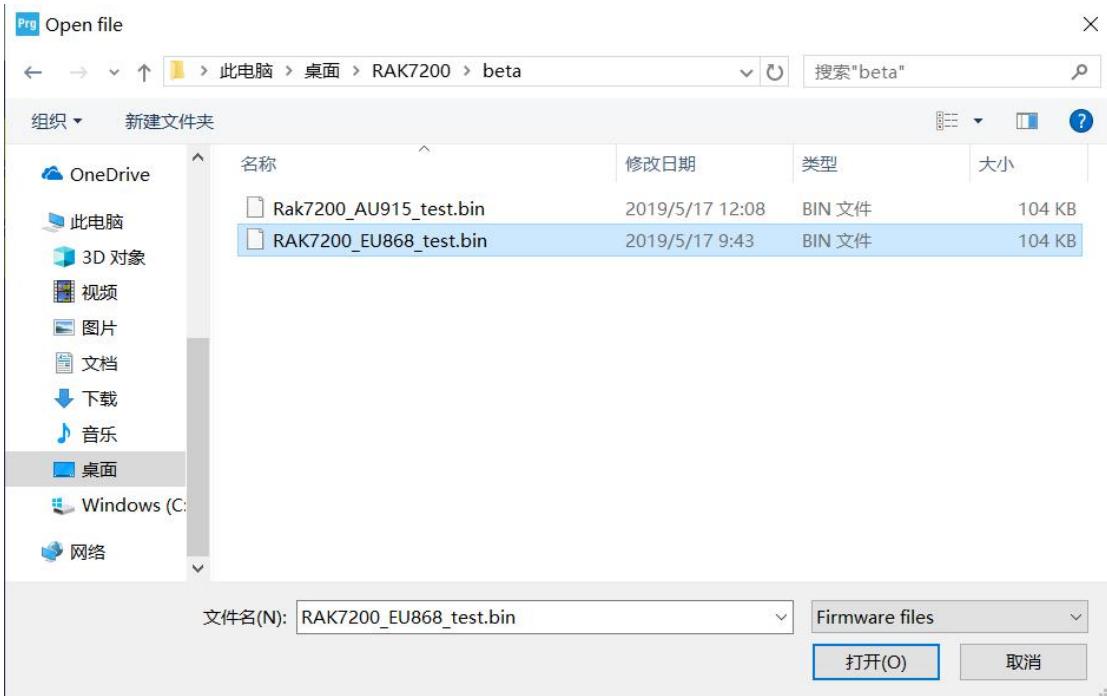
Now, let's start to burn a firmware into RAK7200.

At first, erase all data on RAK7200 according the following picture shows, it may take several seconds:

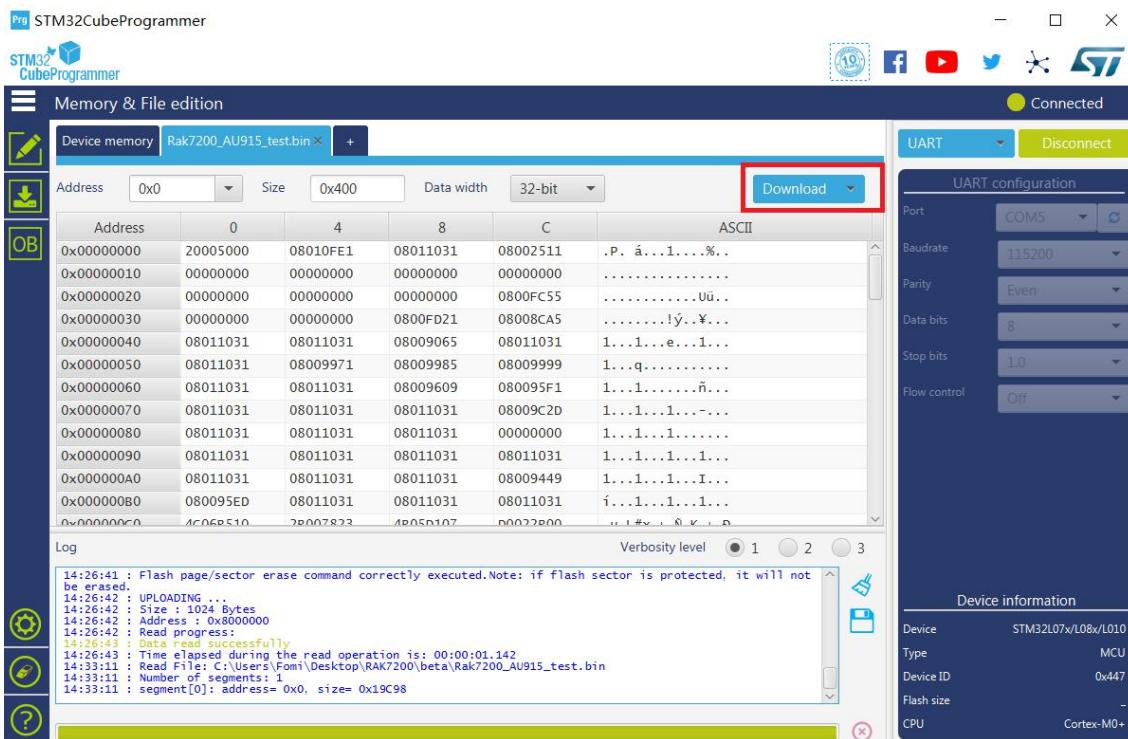


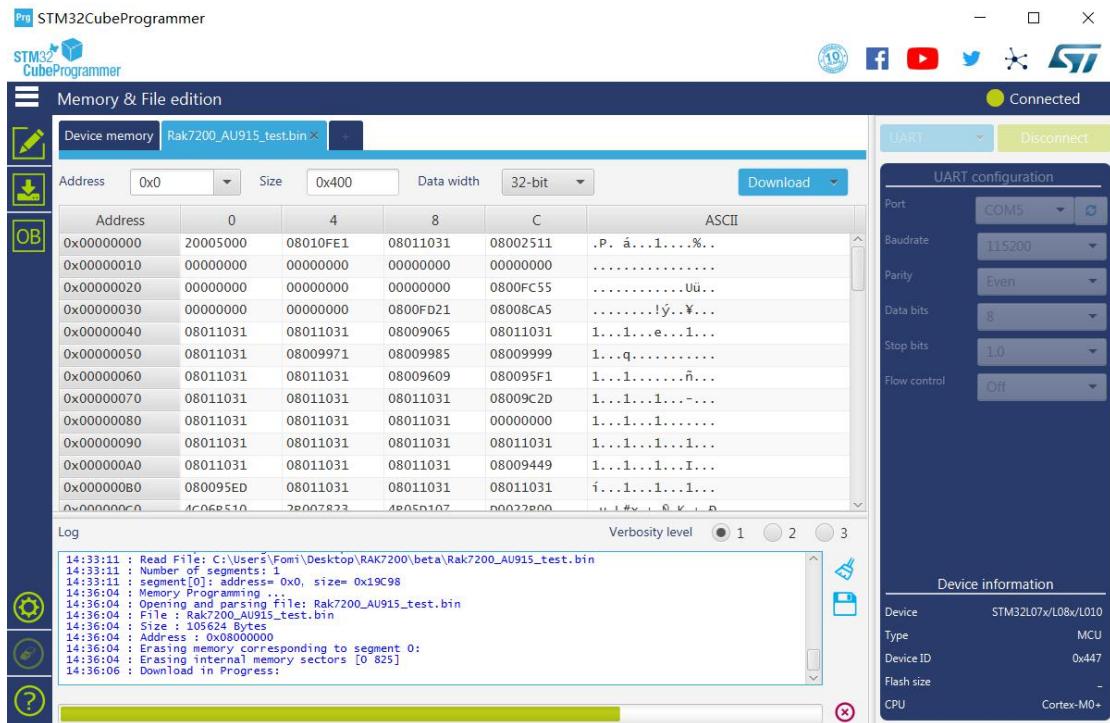
Press “Open file” and select the correct firmware file in the pop-up window as follow:



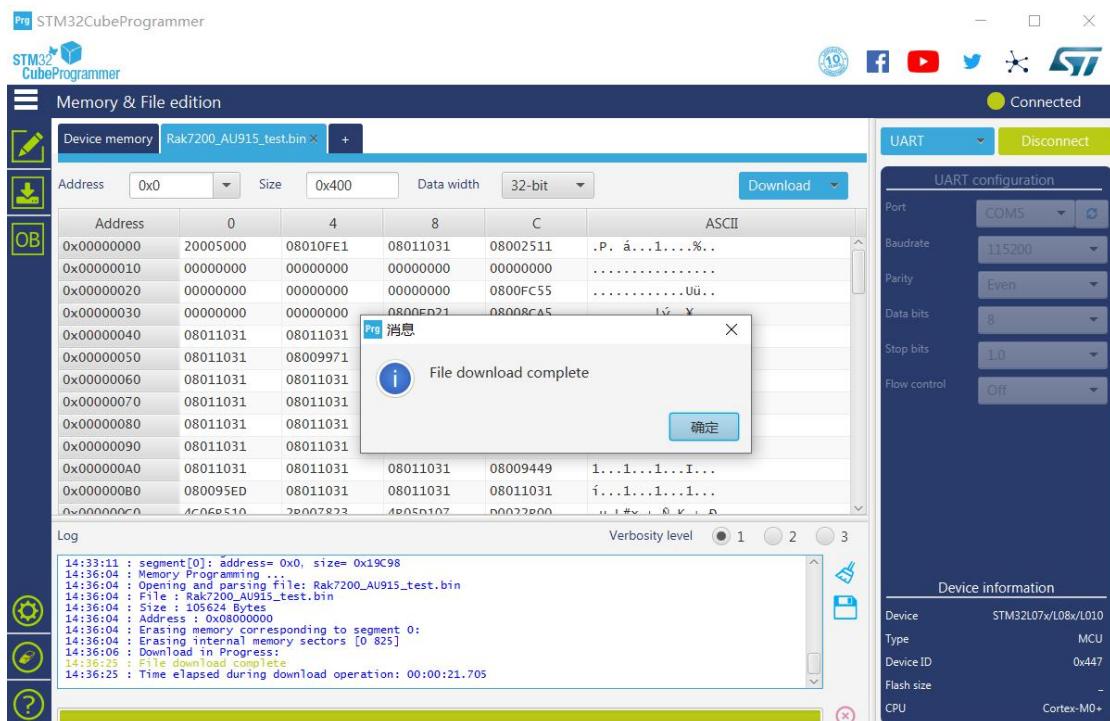


Press the “Download” button to start the burning process:





OK, you have upgraded the firmware for RAK7200 successfully!



Now, “Disconnect” and close the “STM32CubeProgrammer” tool, then open a serial port tool on your PC. Choose the correct COM port and baud rate is 115200. Then press the Reset button on RAK7200, you’ll see the following log on serial port tool:

[14:17:22.929] 收<◆

---



```
*****
S76G_B version:2.0.0.0.0
*****
```

---

[14:17:23.215] 收<◆ DMP is disabled

[14:17:24.603] 收<◆ GPS Init OK

Selected LoRaWAN 1.0.2 Region: AU915

[14:17:24.655] 收<◆ Parameter not found.

Note: If you haven't a serial port tool, we recommend you to use RAK serial port tool. You can get it from RAK website and use it freely.

[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)

### 3. How to configure RAK7200?

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

The following list shows the AT commands:

AT Command	Description
at+version	Get the current firmware version number.
at+get_config=device:status	Get all information about the device's hardware components and their current status.
at+set_config=device:restart	After set, the device will restart.

<b>at+set_config=device:XXX:YYY</b>	<p>Set a certain sensor's status.</p> <p>XXX definition:the sensor's flag, gps means GPS, acc means Accelerate, magn means Magnetic, gyro means Gyroscope, pressure means Pressure, temperature means Temperature, humidity means Humidity, light_strength means Light_strength, voltage means Voltage.</p> <p>YYY definition: 0: close, 1: open</p>
<b>at+join</b>	<p>Start to join LoRa network.</p>
<b>at+send=X:YYY</b>	<p>Send a customized data.</p> <p>X definition: LoRa port</p> <p>YYY definition: the data which you want to send. The limited length is 50 Bytes, and the data must be in HEX format.</p>
<b>at+set_config=lora:work_mode:X</b>	<p>Set the work mode for LoRa.</p> <p>X definition: 0: LoRaWAN, 1: LoRaP2P, 2: Test Mode.</p>
<b>at+set_config=lora:join_mode:X</b>	<p>Set the join mode for LoRaWAN.</p> <p>X definition: 0: OTAA, 1: ABP</p>
<b>at+set_config=lora:class:X</b>	<p>Set the class for LoRa.</p> <p>X definition: 0: Class A, 1: Class B, 2: Class C</p>
<b>at+set_config=lora:region:XXX</b>	<p>Set the region for LoRa.</p> <p>XXX define: one of the following items: EU868 EU433, CN470, IN865, EU868, AU915, US915, KR920, AS923.</p>

<code>at+set_config=lora:confirm:X</code>	Set the type of messages which will be sent out through LoRa.  X definition: 0: unconfirm, 1: confirm
<code>at+set_config=lora:ch_mask:X:Y</code>	Set a certain channel on or off.  X definition: the channel number, and you can check which channel can be set before you set it.  Y definition: 0: off, 1: on
<code>at+set_config=lora:dev_eui:XXXX</code>	Set the device EUI for OTAA.  XXXX definition: the device EUI, for example, 3534353165375300
<code>at+set_config=lora:app_eui:XXXX</code>	Set the application EUI for OTAA.  XXXX definition: the application EUI, for example, 70B3D57ED001A1E2
<code>at+set_config=lora:app_key:XXXX</code>	Set the application key for OTAA.  XXXX definition: the application key, for example, D9988A5F02D80FAB8BA5F453C4A2C D2B
<code>at+set_config=lora:dev_addr:XXXX</code>	Set the device address for ABP.  XXXX definition: the device address, for example, 2601116D
<code>at+set_config=lora:apps_key:XXXX</code>	Set the application session key for ABP.  XXXX definition: the application session key, for example, 573BD4DEC56BA4A9C462DF29E54B9 BCE
<code>at+set_config=lora:nwks_key:XXXX</code>	Set the network session key for ABP.  XXXX definition: the network session key, for example, C2AA51E61BA45F57045BF48249BC3 6F6
<code>at+set_config=lora:send_interval:X</code>	Set the interval time of sending data.

	X definition: the interval time, units are seconds.
at+get_config=lora:status	It will return all of the current information of LoRa, except LoRa channel.
at+get_config=lora:channel	It will return the state of all LoRa channels, then you can see which channel is closed and which channel is open very clearly.

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

## 4. How to connect with TTN?

In this section, we'll do some practice to show how to connect RAK7200 with TTN.

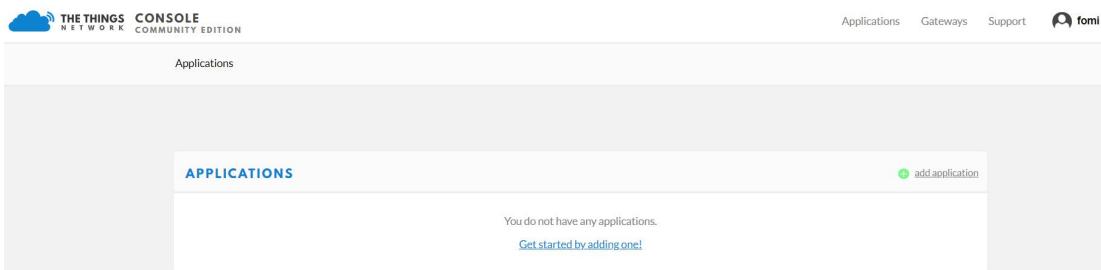
Firstly, open the serial port tool on your PC.

Open the serial port by click the following button:

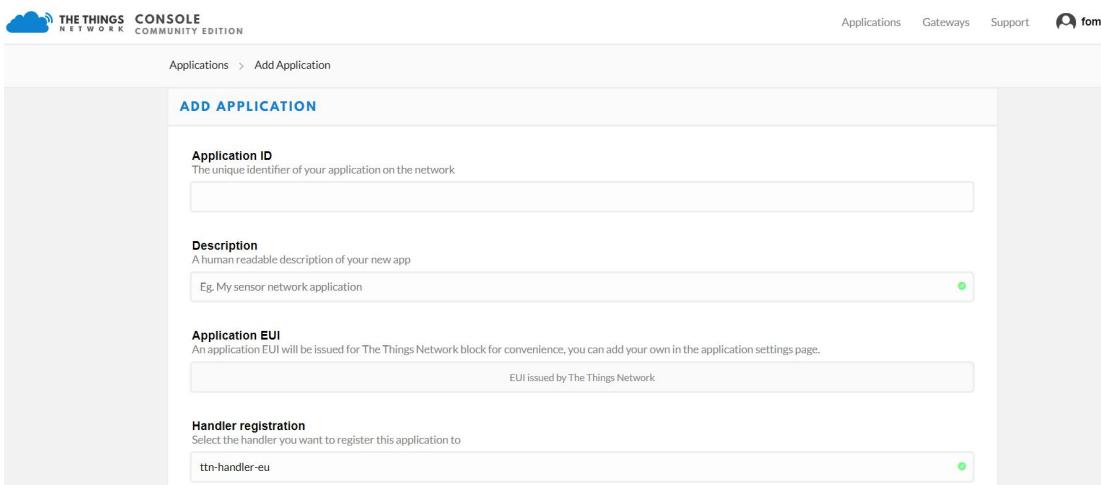


Open the link <https://www.thethingsnetwork.org/> and login, then open the “Console” page from the right corner at the top:

Press “APPLICATIONS”:

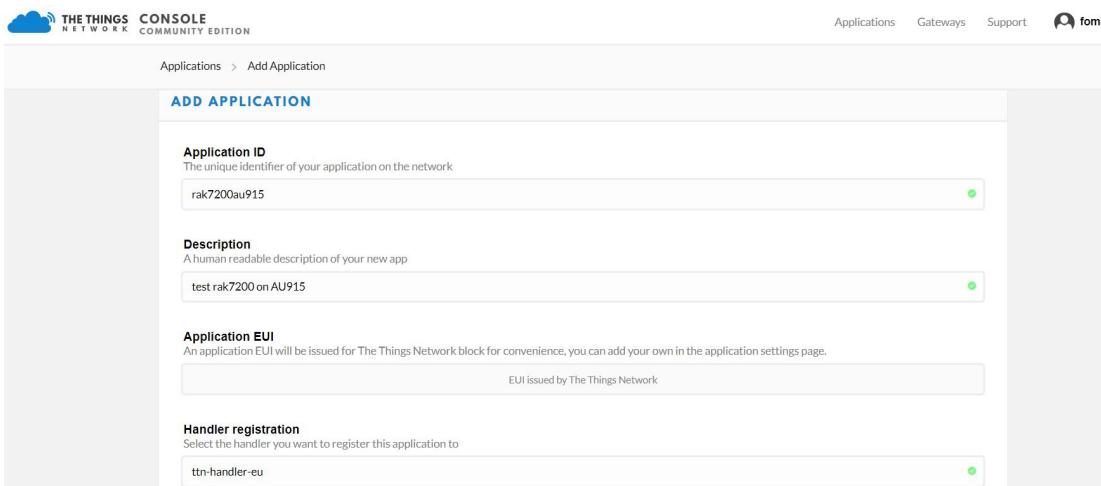


“add application”:

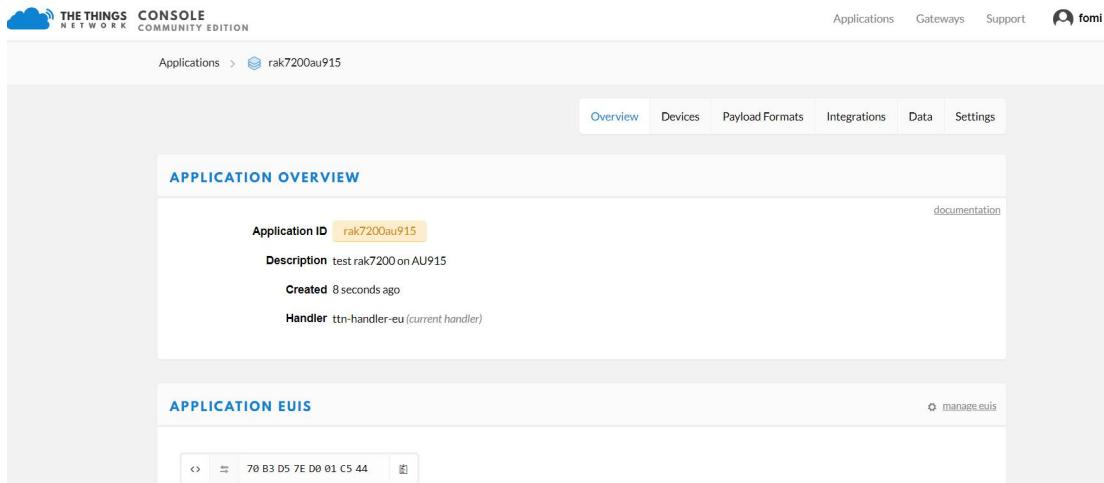


Fill in the correct contents.

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.



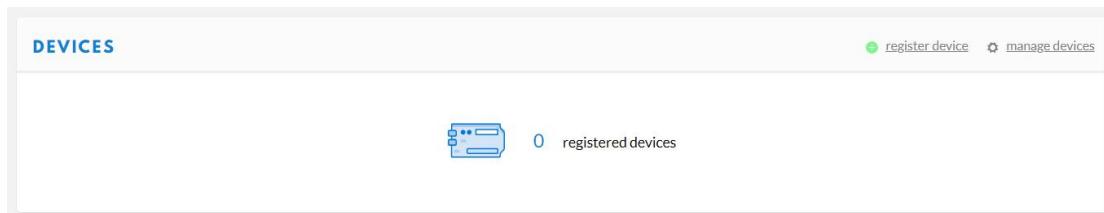
Then press the “Add application” button at the bottom of this page, and you can see the following page:



The screenshot shows the "APPLICATION OVERVIEW" section of the console. Key details include:

- Application ID:** rak7200au915
- Description:** test rak7200 on AU915
- Created:** 8 seconds ago
- Handler:** ttn-handler-eu (current handler)

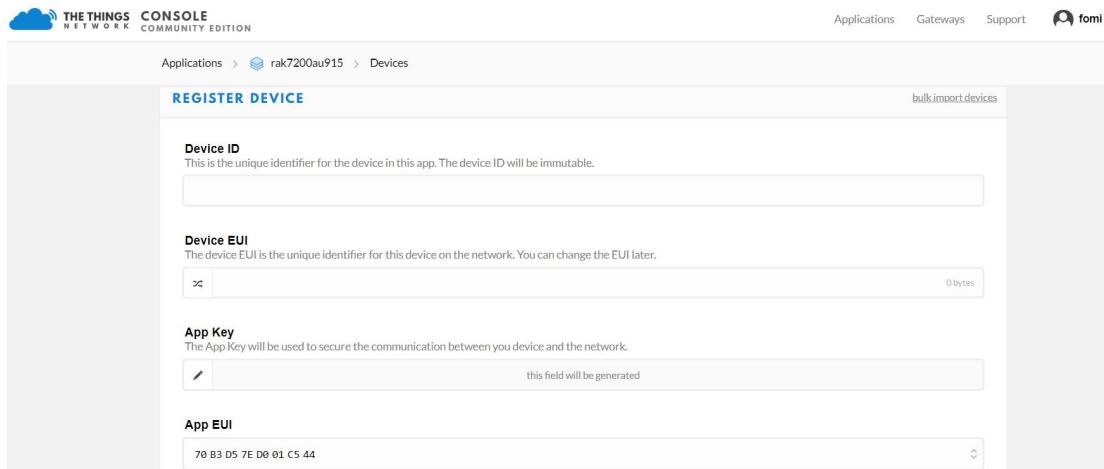
At the middle of this page, you can find the box named “DEVICES”:



The screenshot shows the "DEVICES" section with the following information:

- Icon:** A small icon representing a device.
- Count:** 0 registered devices

Just “register device”:



The screenshot shows the "REGISTER DEVICE" form with the following fields:

- Device ID:** (Input field)
- Device EUI:** (Input field with placeholder: 70 B3 D5 7E D0 01 C5 44)
- App Key:** (Input field with note: "this field will be generated")
- App EUI:** (Input field with placeholder: 70 B3 D5 7E D0 01 C5 44)

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 

Applications >  rak7200au915 > 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.

x
0 bytes

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


this field will be generated

**App EUI**


70 B3 D5 7E D0 01 C5 44

The following picture shows the final page:


**THE THINGS  
NETWORK** **CONSOLE**  
COMMUNITY EDITION
 Applications Gateways Support 

Applications >  rak7200au915 > 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.

 this field will be generated

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

 this field will be generated

**App EUI**



Cancel Register

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

**THE THINGS NETWORK** CONSOLE COMMUNITY EDITION

Applications > rak7200au915 > Devices > 353730345a377e11

**DEVICE OVERVIEW**

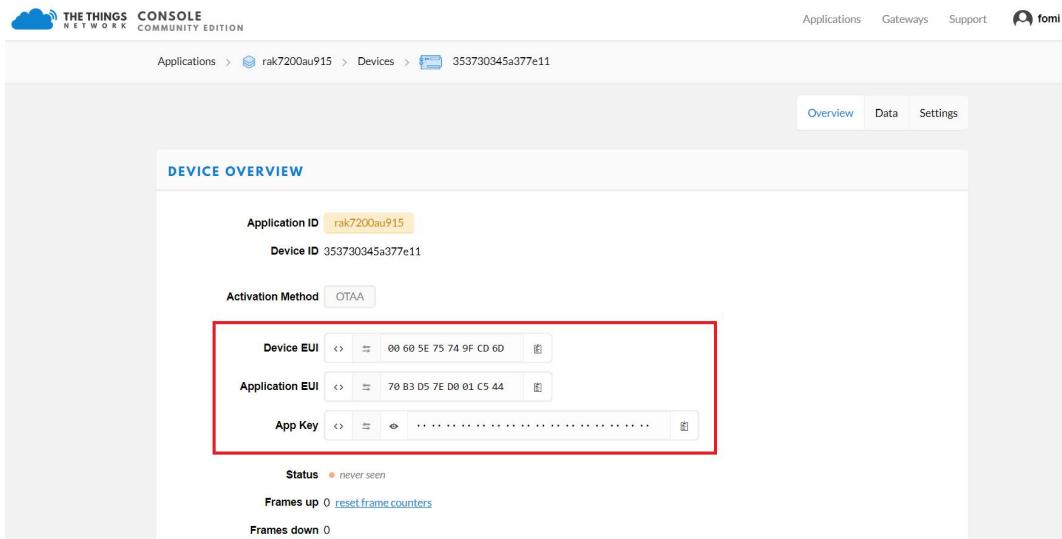
Application ID	rak7200au915
Device ID	353730345a377e11
Activation Method	OTAA
Device EUI	<input type="text" value="00 60 5E 75 74 9F CD 6D"/>
Application EUI	<input type="text" value="70 B3 D5 7E D0 01 C5 44"/>
App Key	<input type="text" value="..."/>
Status	<span>never seen</span>
Frames up	0 <a href="#">reset frame counters</a>
Frames down	0

**Overview** **Data** **Settings**

## 4.1 Join in OTAA mode

As you see in the above page, the default activation method is OTAA.

These three parameters will be used on RAK7200:

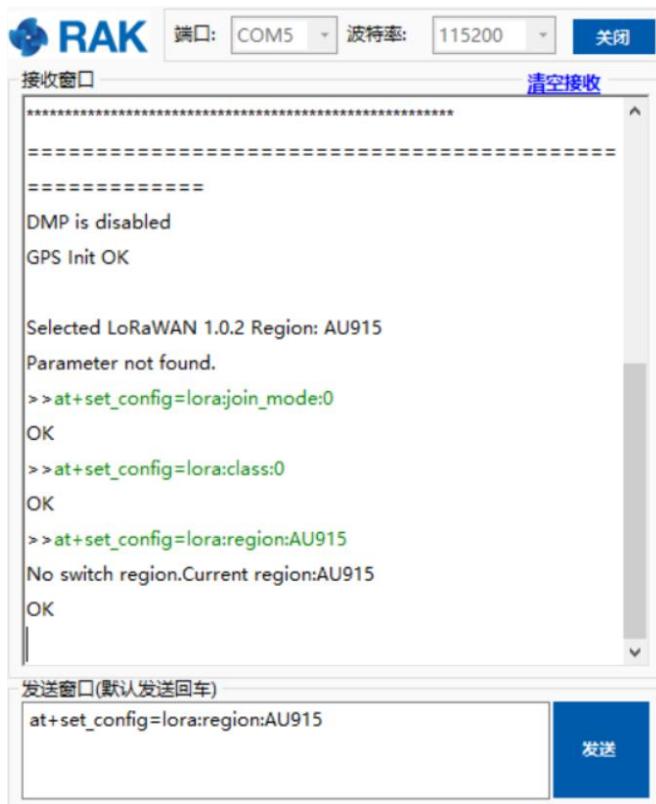


The screenshot shows the 'DEVICE OVERVIEW' section of the The Things Network Console. The 'Activation Method' is set to 'OTAA'. The 'Device EUI', 'Application EUI', and 'App Key' fields are highlighted with a red box. The 'Device EUI' value is 00 60 5E 75 74 9F CD 60. The 'Application EUI' value is 70 B3 D5 7E D8 01 C5 44. The 'App Key' field contains a long string of hex digits. Below the fields, the 'Status' is listed as 'never seen'.

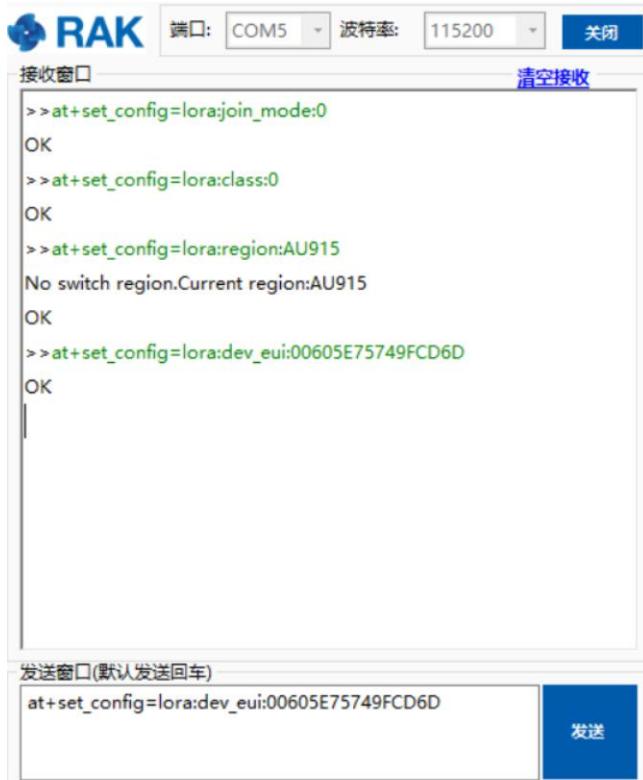
OK! Now, let's join in OTAA mode and AU915 frequency for example!

The default LoRa work mode is LoRaWAN 1.0.2, the default LoRa join mode is OTAA, and the default LoRa class is Class A.

Set the frequency/region to AU915:



Set the Device EUI:



Set the Application EUI:

端口: COM5 波特率: 115200 关闭

接收窗口 清空接收

```
>> at+set_config=lora:join_mode:0
OK
>> at+set_config=lora:class:0
OK
>> at+set_config=lora:region:AU915
No switch region.Current region:AU915
OK
>> at+set_config=lora:dev_eui:00605E75749FCD6D
OK
>> at+set_config=lora:app_eui:70B3D57ED001C544
OK
```

发送窗口(默认发送回车)

```
at+set_config=lora:app_eui:70B3D57ED001C544
```

发送

Set the Application Key:

端口: COM5 波特率: 115200 关闭

接收窗口 清空接收

```
>>
at+set_config=lora:app_key:4E2003296FC5CD26F46E940A6DAFA9D1
OK
```

发送窗口(默认发送回车)

```
at
+set_config=lora:app_key:4E2003296FC5CD26F46E940A6
DAFA9D1
```

发送

Join in OTAA mode:



Join successfully! You can see that RAK7200 is sending data frequently.

Now, you can see the sensor data send from RAK7200 on TTN website as follow:

THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications > lorabuttontest123 > Devices > 01a7c01e4db9b44d > Data

Overview Data Settings

APPLICATION DATA

Filters: uplink, downlink, activation, ack, error

time	counter	port	
23:08:13	0		
23:08:10	42	2 confirmed	payload: 07 02 01 A0 03 71 00 54 FF 61 03 F0 05 86 FF FD 00 00 00 01 06 73 25 2E 02 67 01 34
23:07:46	0		
23:07:43	41	2 confirmed	payload: 07 02 01 9E 03 71 00 55 FF 70 04 04 05 86 FF FD 00 00 00 01 06 73 25 2E 02 67 01 34
23:07:39	0		
23:07:37	40	2 confirmed	payload: 07 02 01 9B 03 71 00 5B FF 62 03 FB 05 86 FF FD 00 00 00 01 06 73 25 2E 02 67 01 34

Great! That's all about OTAA mode.

## 4.2 Join in ABP mode

(to add)

## 5. How to connect with LoRaServer?

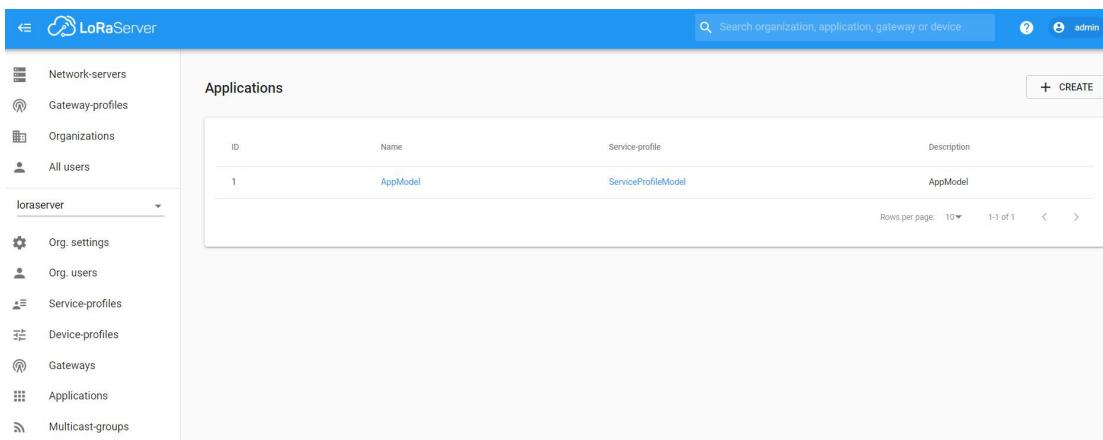
The LoRaServer is an open source project which you can find in <https://www.loraserver.io/>.

You can use RAK7200 to connect with LoRaServer according to the following steps:

In this document, i assume that you are using RAK LoRa gateway and its built-in LoRaServer, or you are using RAK cloud testing LoRaServer. But if you are using a LoRaServer which is built by yourself, maybe you need to configure it by yourself too.

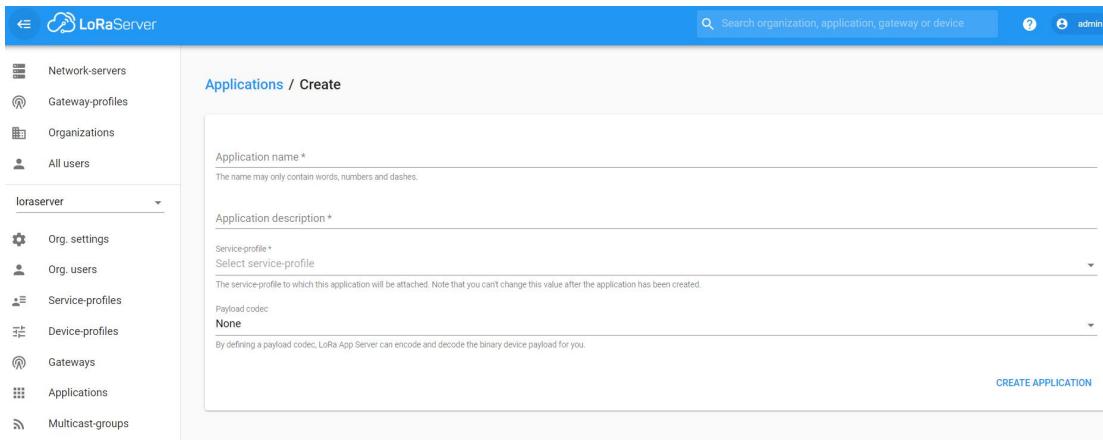
OK! Let's get start!

Open the web page of the LoRaServer which you want to connect with and login.

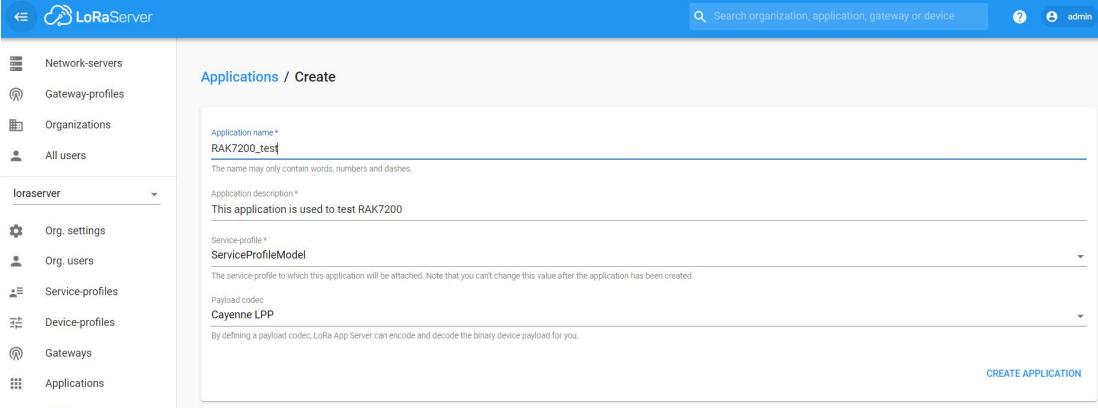


The screenshot shows the LoRaServer application management interface. On the left, there is a sidebar with various navigation options: Network-servers, Gateway-profiles, Organizations, All users, and a dropdown menu for 'loraserver' containing: Org. settings, Org. users, Service-profiles, Device-profiles, Gateways, Applications, and Multicast-groups. The main content area is titled 'Applications' and displays a table with one row. The table columns are ID, Name, Service-profile, and Description. The single entry is ID: 1, Name: AppModel, Service-profile: ServiceProfileModel, and Description: AppModel. At the bottom right of the table, there are buttons for 'Rows per page' (set to 10), '1-1 of 1', and navigation arrows. A 'CREATE' button is located at the top right of the table area.

By default, there is already one or more items in this page, you can use it or create a new item. Now, let's create a new item by click the "CREATE" button, and fill in them.



The screenshot shows the 'Applications / Create' form. The sidebar on the left is identical to the previous screenshot. The main form has several input fields: 'Application name \*' (with a note: 'The name may only contain words, numbers and dashes.'), 'Application description \*', 'Service profile \*' (with a dropdown menu labeled 'Select service-profile'), 'Payload codec' (with a dropdown menu labeled 'None'), and a note: 'By defining a payload codec, LoRa App Server can encode and decode the binary device payload for you.' At the bottom right of the form, there is a 'CREATE APPLICATION' button.



Applications / Create

Application name \*  
RAK7200\_test

The name may only contain words, numbers and dashes.

Application description \*  
This application is used to test RAK7200

Service-profile \*  
ServiceProfileModel

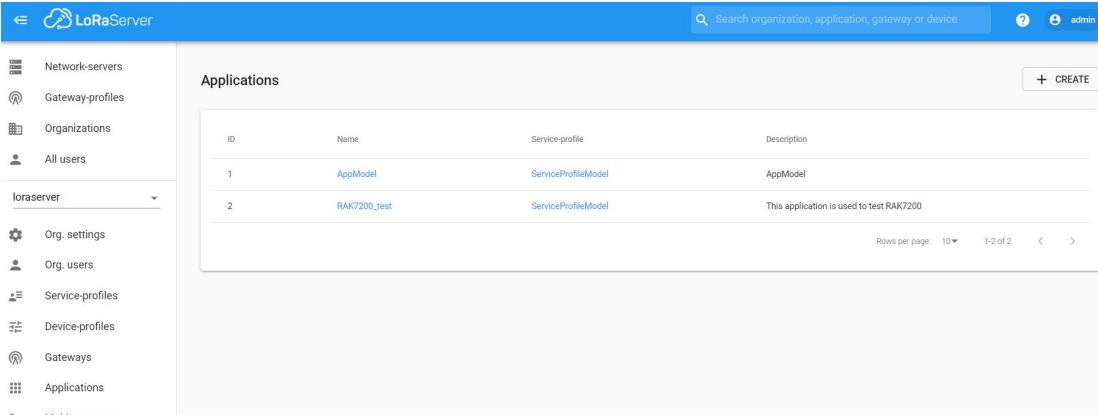
The service/profile to which this application will be attached. Note that you can't change this value after the application has been created.

Payload codec  
Cayenne LPP

By defining a payload codec, LoRa App Server can encode and decode the binary device payload for you.

**CREATE APPLICATION**

“CREATE APPLICATION”.

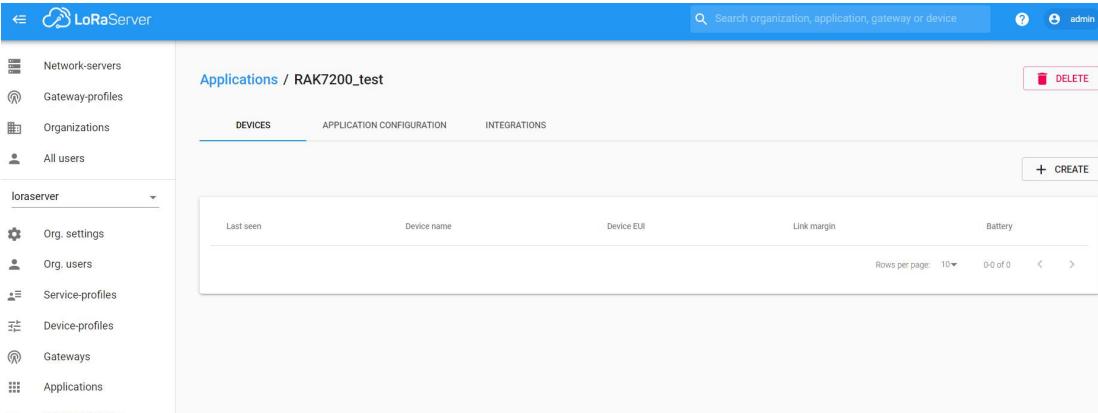


Applications

ID	Name	Service-profile	Description
1	AppModel	ServiceProfileModel	AppModel
2	RAK7200_test	ServiceProfileModel	This application is used to test RAK7200

+ CREATE

Click the new item name “RAK7200\_test”:



Applications / RAK7200\_test

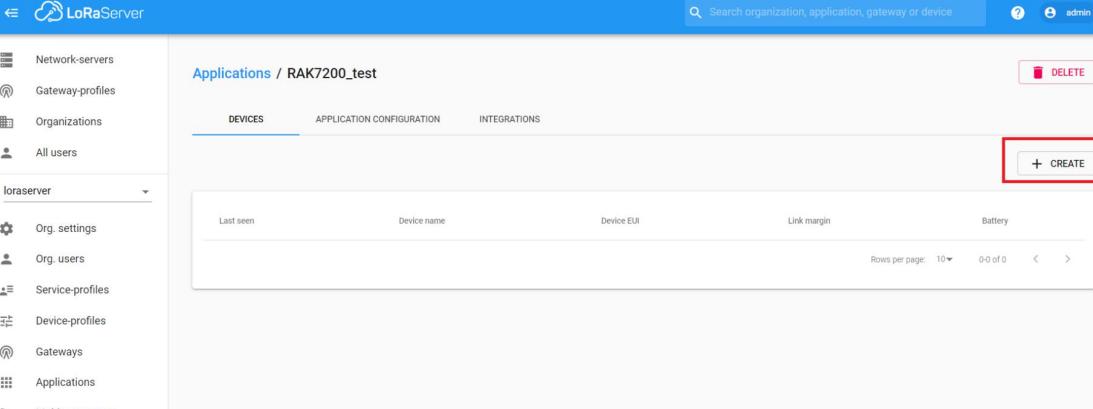
**DEVICES** APPLICATION CONFIGURATION INTEGRATIONS

**CREATE**

Last seen	Device name	Device EUI	Link margin	Battery

Rows per page: 10 1-0 of 0 < >

Add a LoRa node device into LoRaServer by clicking the “CREATE” button:



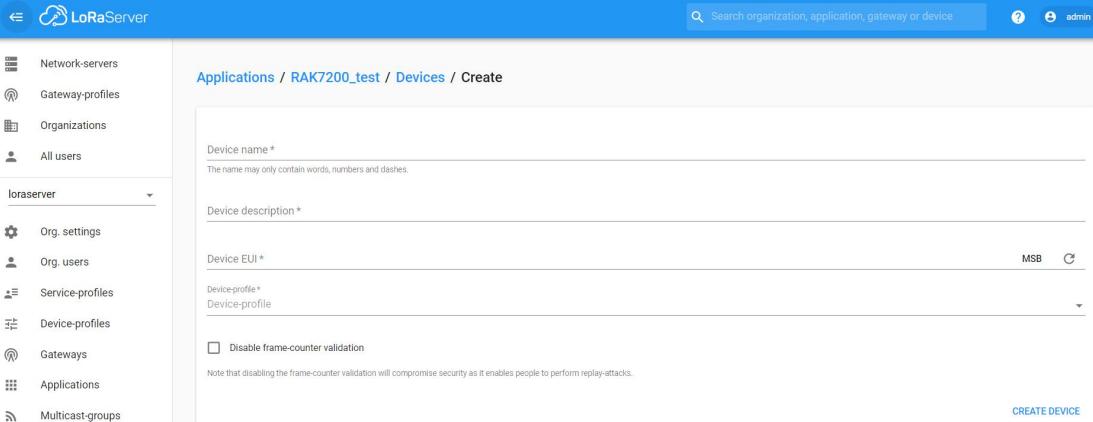
Applications / RAK7200\_test

**DEVICES** APPLICATION CONFIGURATION INTEGRATIONS

+ CREATE

Last seen	Device name	Device EU	Link margin	Battery

Rows per page: 10 ▾ 0-0 of 0 < >



Applications / RAK7200\_test / Devices / Create

Device name \*

The name may only contain words, numbers and dashes.

Device description \*

Device EUI \*

Device-profile \*

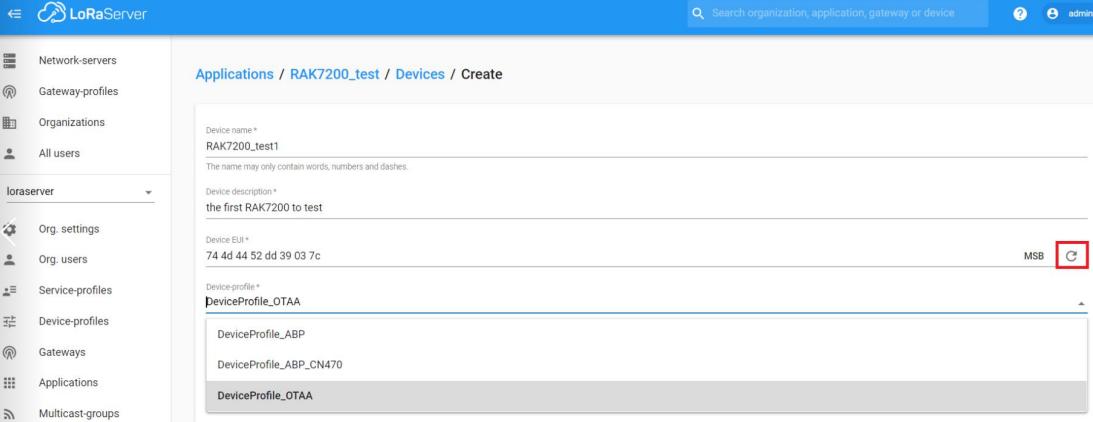
Device-profile

Disable frame-counter validation

Note that disabling the frame-counter validation will compromise security as it enables people to perform replay-attacks.

CREATE DEVICE

Fill in them. You can generate a Device EUI automatically by click the following icon, or you can write a correct Device EUI in the edit box.



Applications / RAK7200\_test / Devices / Create

Device name \*

RAK7200\_test1

The name may only contain words, numbers and dashes.

Device description \*

the first RAK7200 to test

Device EUI \*

74 4d 44 52 dd 39 03 7c

Device-profile \*

DeviceProfile\_OTAA

DeviceProfile\_ABP

DeviceProfile\_ABP\_CN470

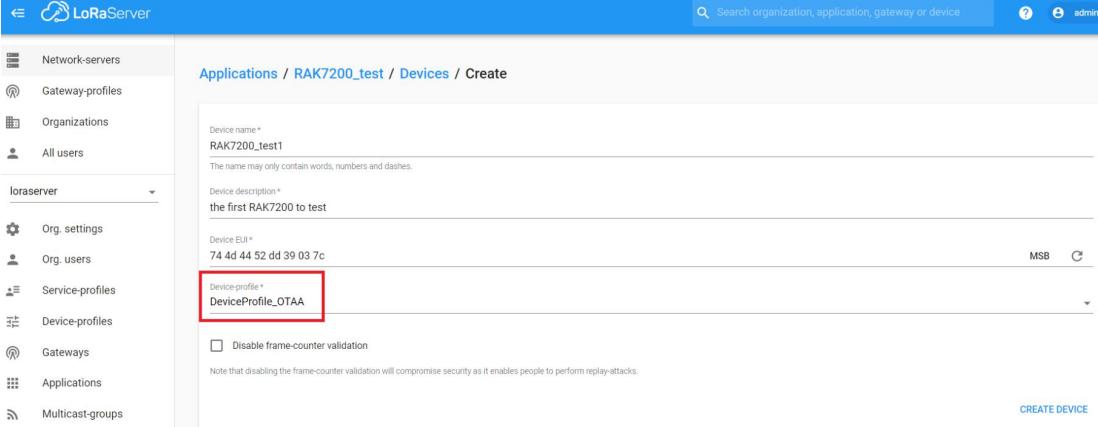
DeviceProfile\_OTAA

MSB 

Note: If you want to join in OTAA mode, you should select “**DeviceProfile\_OTAA**” in the “Device-profile” item. If you want to join in ABP mode and CN470 frequency, you should select “**DeviceProfile\_ABP\_CN470**” in the “Device-profile” item. If you want to join in ABP mode and other frequencies except AS923 and CN470, you should select “**DeviceProfile\_ABP**” in the “Device-profile” item. What about AS923 in ABP mode? Sorry! LoRaServer can not support it now.

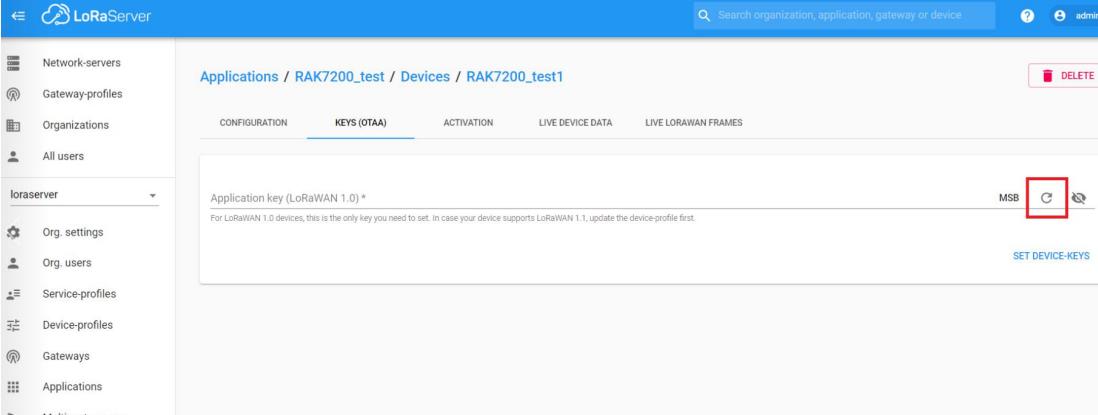
## 5.1 Join in OTAA mode

If you select “DeviceProfile\_OTAA”, it means you want to join LoRaServer in OTAA mode.

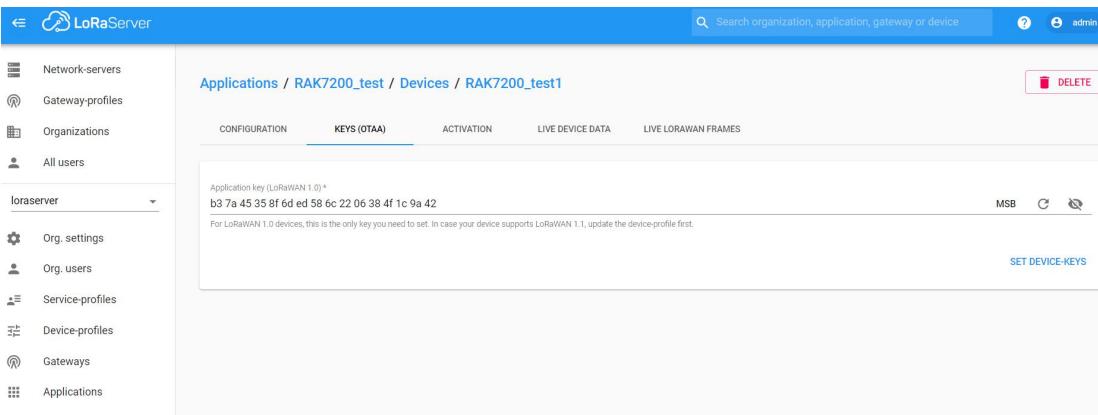


The screenshot shows the 'Devices / Create' section of the LoRaServer interface. The 'Device profile' dropdown menu is open, and the option 'DeviceProfile\_OTAA' is selected and highlighted with a red box. Other options like 'DeviceProfile\_APSK' and 'DeviceProfile\_LORA' are also visible in the list.

“CREATE DEVICE”. Then generate the application key in this page. You can write it by yourself or generate it automatically by clicking the following icon:



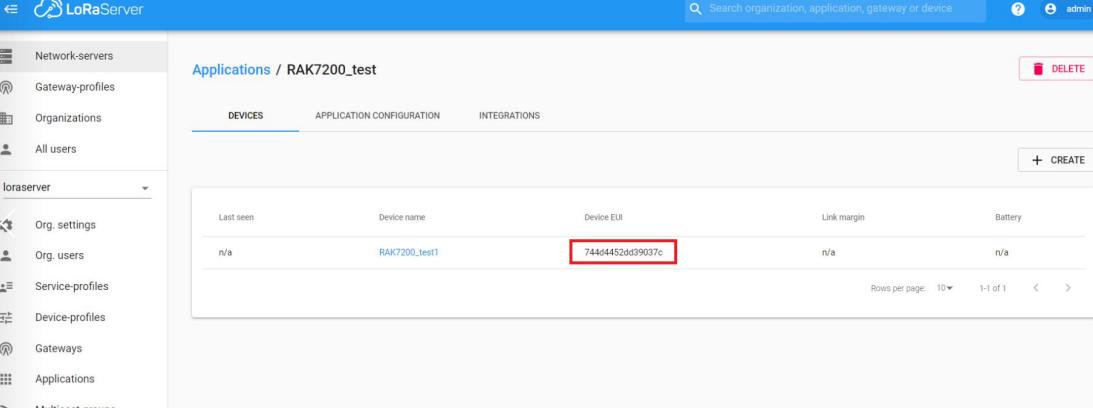
The screenshot shows the 'Edit Device' page with the 'KEYS (OTAA)' tab selected. The 'Application key (LoRaWAN 1.0)' field is empty. Below it, there is a note: 'For LoRaWAN 1.0 devices, this is the only key you need to set. In case your device supports LoRaWAN 1.1, update the device-profile first.' At the bottom right of the form, there is a blue 'SET DEVICE-KEYS' button, which is highlighted with a red box.



The screenshot shows the 'Edit Device' page with the 'KEYS (OTAA)' tab selected. The 'Application key (LoRaWAN 1.0)' field now contains the hex value 'b3 7a 45 35 8f 6d ed 5b 6c 22 06 38 4f 1c 9a 42'. Below it, there is a note: 'For LoRaWAN 1.0 devices, this is the only key you need to set. In case your device supports LoRaWAN 1.1, update the device-profile first.' At the bottom right of the form, there is a blue 'SET DEVICE-KEYS' button, which is highlighted with a red box.

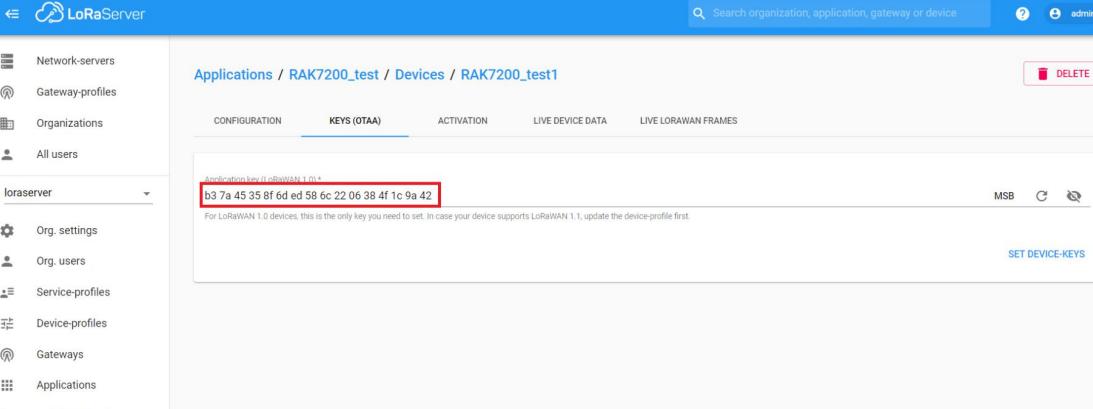
“SET DEVICE-KEYS”. That’s OK! You’ve complete the configuration on LoRaServer.

As you see, the Device EUI which will be set into RAK7200 as “dev\_eui” is this one:



The screenshot shows the LoRaServer application management interface. On the left, there's a sidebar with various navigation options like Network-servers, Gateway-profiles, Organizations, and Applications. Under Applications, it shows 'RAK7200\_test'. The main area has tabs for DEVICES, APPLICATION CONFIGURATION, and INTEGRATIONS. The DEVICES tab is selected, displaying a table with columns: Last seen, Device name, Device EUI, Link margin, and Battery. A single row is present for 'RAK7200\_test1', with its Device EUI ('74464452dd39037c') highlighted with a red box.

The Application Key which will be set into RAK7200 as “app\_key” is this one:

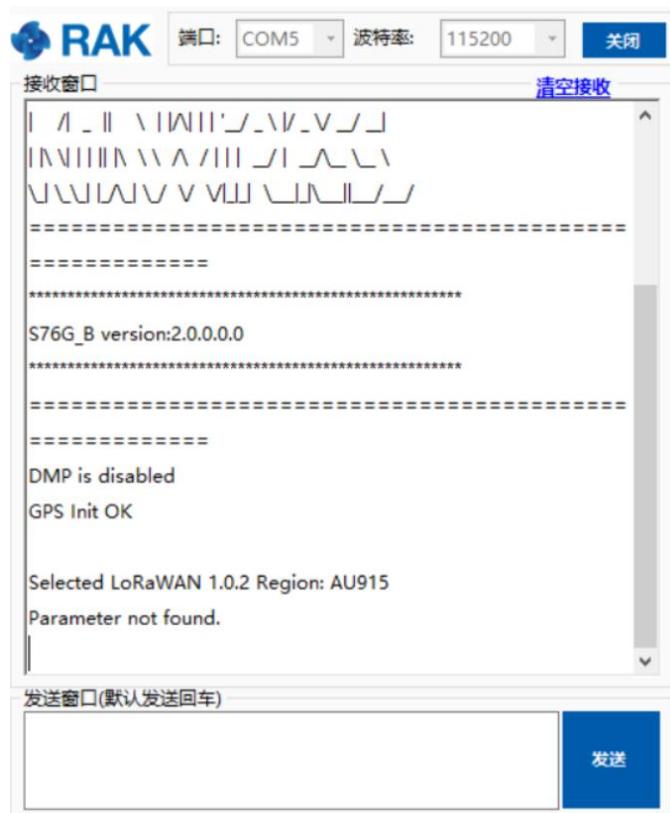


The screenshot shows the LoRaServer device configuration interface. It's a detailed view of the 'RAK7200\_test1' device under 'RAK7200\_test'. The top navigation bar includes a back arrow, a cloud icon, and the text 'LoRaServer'. The sidebar on the left is identical to the previous screenshot. The main content area has tabs for CONFIGURATION, KEYS (OTAA), ACTIVATION, LIVE DEVICE DATA, and LIVE LORAWAN FRAMES. The KEYS (OTAA) tab is selected. It contains a sub-section for 'Application key (LoRaWAN 1.0.x)' with the value 'b3 7a 45 35 8f 6d ed 58 6c 22 06 38 4f 1c 9a 42' highlighted with a red box. Below this, a note says 'For LoRaWAN 1.0 devices, this is the only key you need to set. In case your device supports LoRaWAN 1.1, update the device-profile first.' There are buttons for 'MSB', 'LSB', and 'SET DEVICE-KEYS'.

The Application EUI which will be set into RAK7200 as “app\_eui” is useless for LoRaServer, and you can set it to any value with a correct format, for example: 7083D57ED001C1CF.

Next, let's configure RAK7200 by using AT command.

Connect your RAK7200 with a PC, power on and open RAK Serial Port Tool.

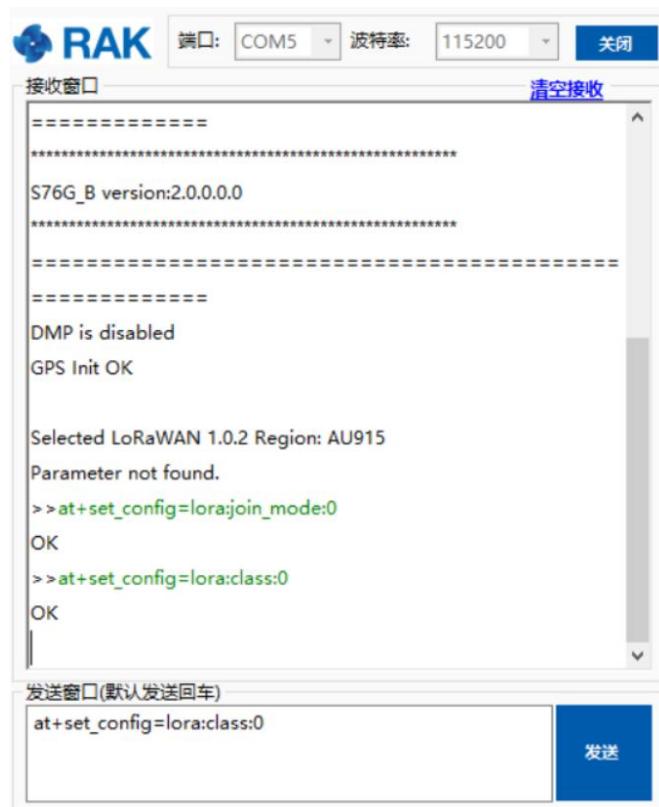


The default LoRa work mode is LoRaWAN 1.0.2.

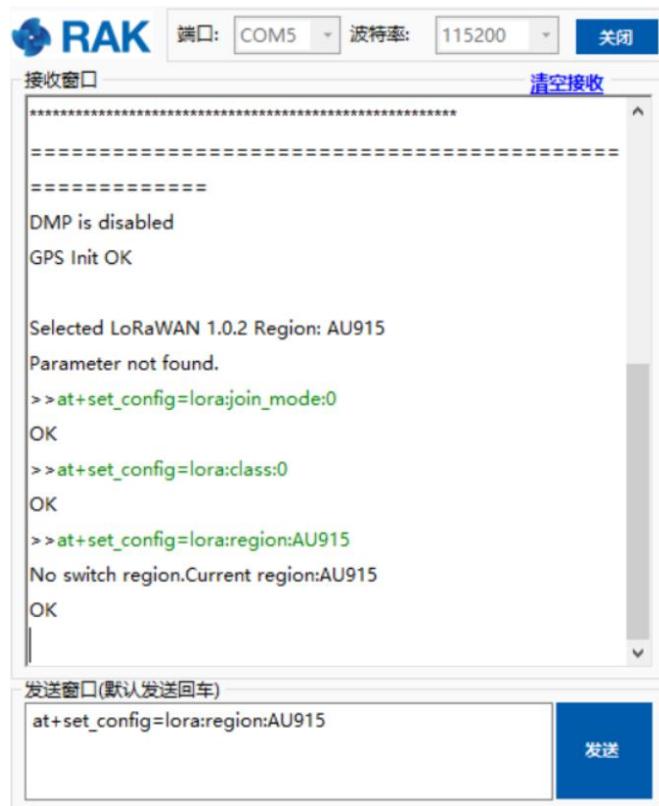
Set the LoRa join mode to OTAA:



Set the LoRa class to Class A:



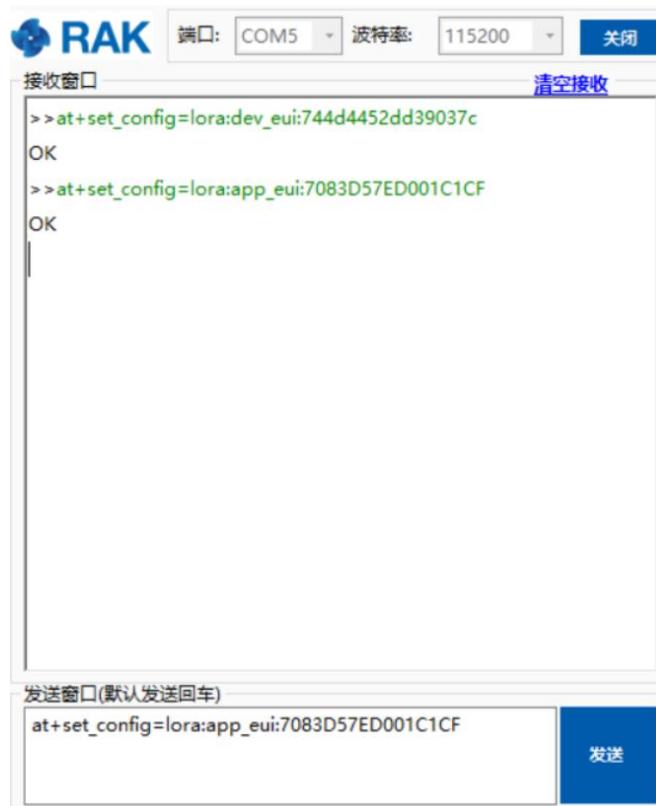
Set the frequency/region to AU915:



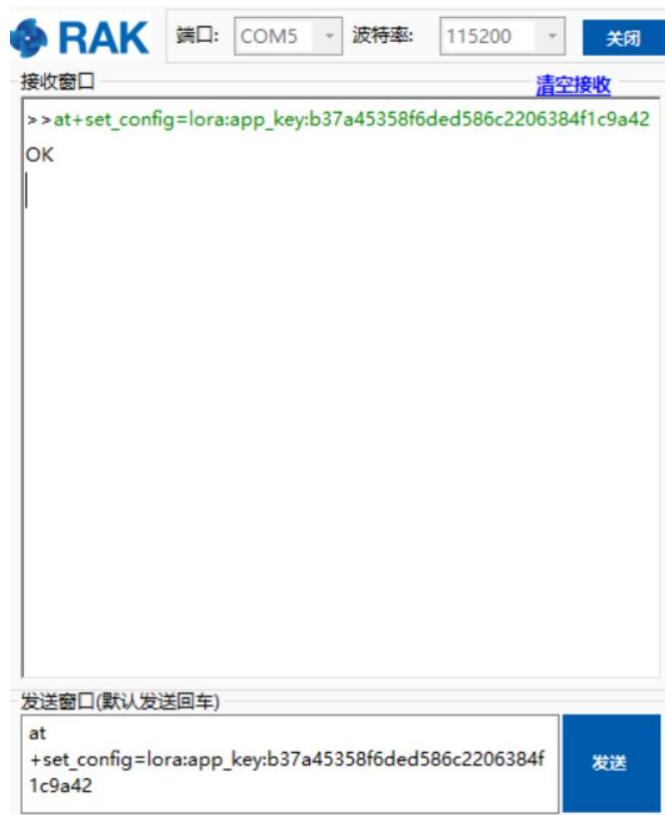
Set “dev\_eui”:



Set “app\_eui”:



Set “app\_key”:

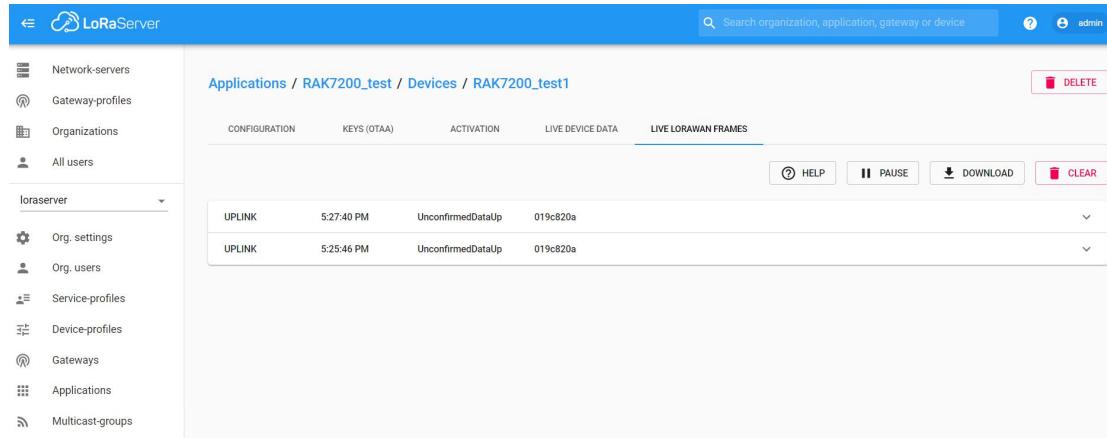


Start to join:



Join successfully!

You can see the data which RAK7200 sends on LoRaServer page:



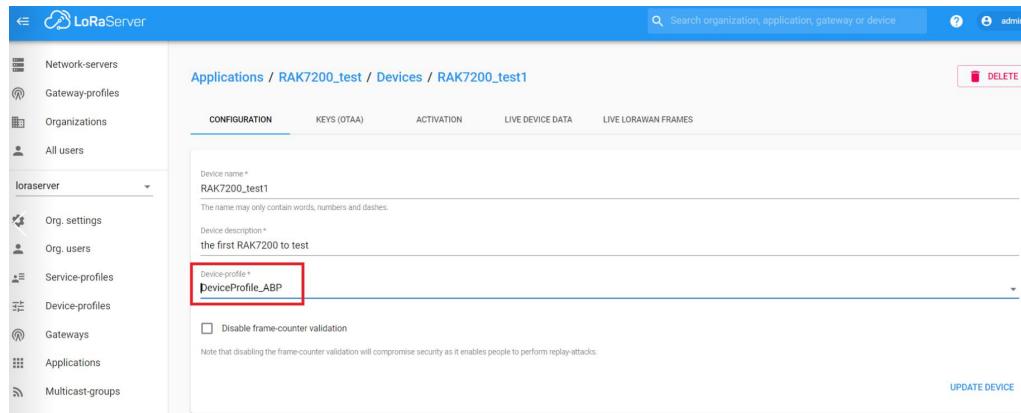
The screenshot shows the LoRaServer web interface. On the left, there's a sidebar with navigation links like Network-servers, Gateway-profiles, Organizations, All users, and a dropdown for 'loraserver' which includes Org. settings, Org. users, Service-profiles, Device-profiles, Gateways, Applications, and Multicast-groups. The main content area has a breadcrumb path: Applications / RAK7200\_test / Devices / RAK7200\_test1. Below the path are tabs: CONFIGURATION, KEYS (OTAA), ACTIVATION, LIVE DEVICE DATA, and LIVE LORAWAN FRAMES. The LIVE LORAWAN FRAMES tab is selected, showing two UPLINK entries. The first entry is at 5:27:40 PM with UnconfirmedDataUp and ID 019c820a. The second entry is at 5:25:46 PM with UnconfirmedDataUp and ID 019c820a. At the top right of the main area are buttons for DELETE, HELP, PAUSE, DOWNLOAD, and CLEAR.

The data format is LPP.

OK, that's all about "Join in OTAA mode" with LoRaServer.

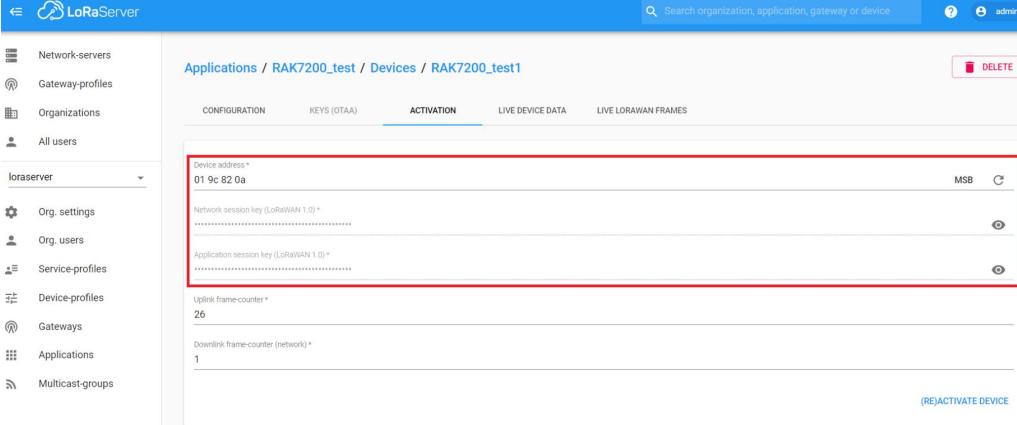
## 5.2 Join in ABP mode

If you select "DeviceProfile\_ABP" or "DeviceProfile\_ABP\_CN470", it means you want to join LoRaServer in OTAA mode.



This screenshot shows the 'ACTIVATION' configuration for a device. The 'Device profile' field is highlighted with a red box and contains 'DeviceProfile\_ABP'. Other fields shown include 'Device name' (RAK7200\_test1) and 'Device description' (the first RAK7200 to test). There's also a checkbox for 'Disable frame-counter validation' with a note below it stating that disabling it will compromise security. At the bottom right is a 'UPDATE DEVICE' button.

Then you can see that there are some parameters for ABP in the "ACTIVATION" item:



The screenshot shows the LoRaServer interface under the 'Applications / RAK7200\_test / Devices / RAK7200\_test1' path. The 'ACTIVATION' tab is selected. A red box highlights the 'Device address' field containing '01 9c 82 0a' and the 'Network session key (LoRaWAN 1.0)' field.

Next, let's use these parameters to set RAK7200 by using AT command.

Set LoRa join mode to ABP:



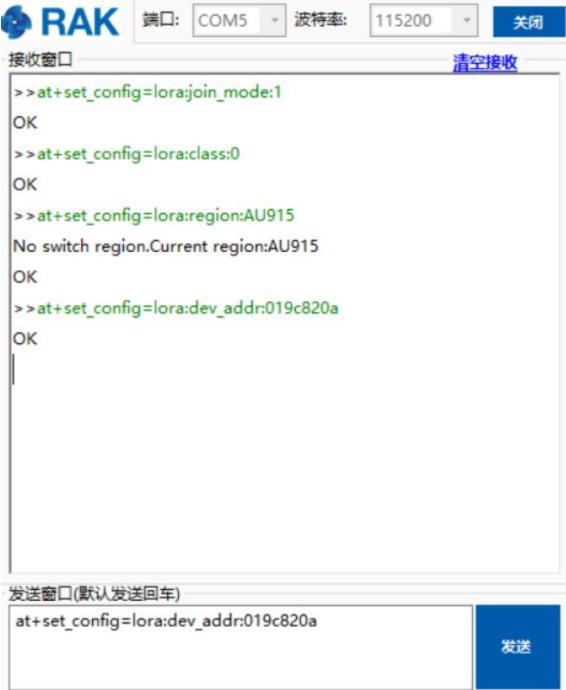
Set LoRa class to Class A:



Set the frequency/region to AU915:



Set "dev\_addr":



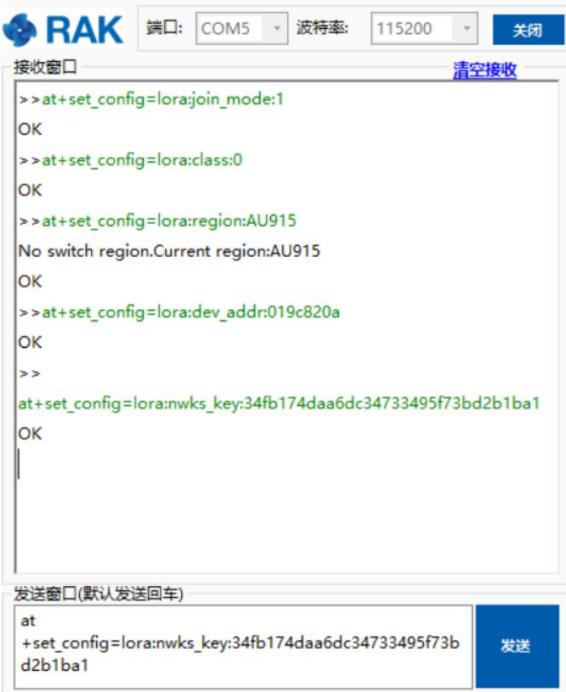
接收窗口

```
>>at+set_config=lora:join_mode:1
OK
>>at+set_config=lora:class:0
OK
>>at+set_config=lora:region:AU915
No switch region.Current region:AU915
OK
>>at+set_config=lora:dev_addr:019c820a
OK
```

发送窗口(默认发送回车)

**发送**

Set “nwks\_key”:



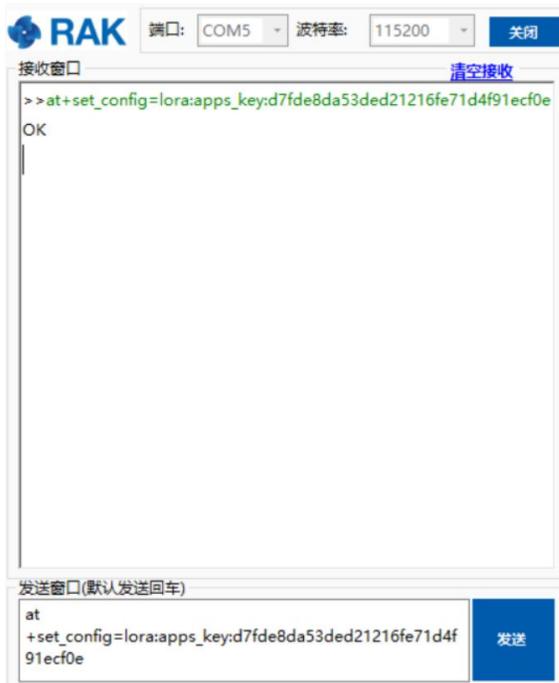
接收窗口

```
>>at+set_config=lora:join_mode:1
OK
>>at+set_config=lora:class:0
OK
>>at+set_config=lora:region:AU915
No switch region.Current region:AU915
OK
>>at+set_config=lora:dev_addr:019c820a
OK
>>
at+set_config=lora:nwks_key:34fb174daa6dc34733495f73bd2b1ba1
OK
```

发送窗口(默认发送回车)

**发送**

Set “apps\_key”:

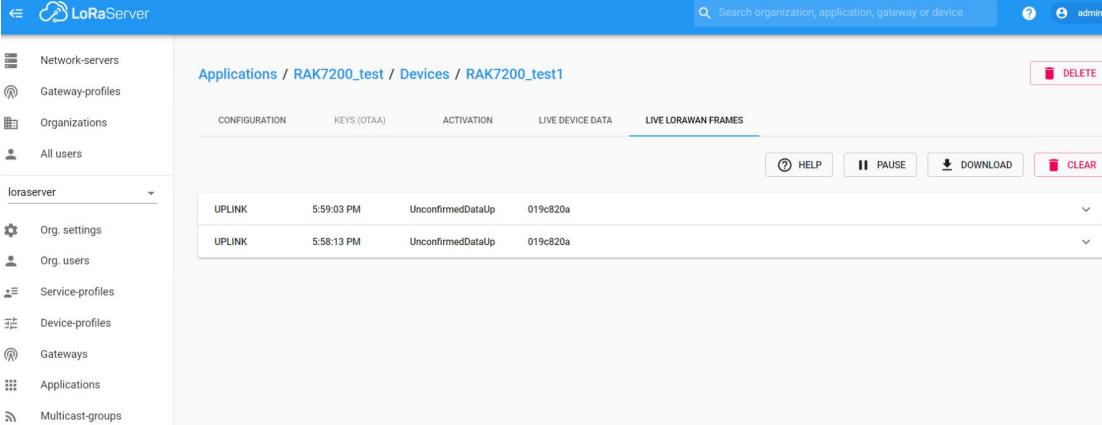


Start to join:



Great! Join and send data successfully!

You can see the data which is just sent from RAK7200 on LoRaServer page:



The screenshot shows the LoRaServer web interface. On the left is a sidebar with navigation links: Network-servers, Gateway-profiles, Organizations, All users, and a dropdown menu for 'loraserver' containing Org. settings, Org. users, Service-profiles, Device-profiles, Gateways, Applications, and Multicast-groups. The main content area has a breadcrumb path: Applications / RAK7200\_test / Devices / RAK7200\_test1. It features tabs for CONFIGURATION, KEYS (OTAA), ACTIVATION, LIVE DEVICE DATA, and LIVE LORAWAN FRAMES (which is selected). Below these tabs are buttons for HELP, PAUSE, DOWNLOAD, and CLEAR. A search bar at the top right contains the placeholder 'Search organization, application, gateway or device'. There is also a user icon and 'admin' text. The LIVE LORAWAN FRAMES section displays two rows of data:

UPLINK	5:59:03 PM	UnconfirmedDataUp	019c820a
UPLINK	5:58:13 PM	UnconfirmedDataUp	019c820a

That's all about “Join in ABP mode” with LoRAServer.

## 6. Revision History

Revision	Description	Date
1.0	Initial version	2019-07-16

## 7. Document Summary

Prepared by	Checked by	Approved by
Fomi	Penn&Fomi	



### About RAKwireless:

RAKwireless is the pioneer in providing innovative and diverse cellular and LoRa connectivity solutions for IoT edge devices. Its easy and modular design can be used in different IoT applications and accelerate time-to-market.

For more information, please visit RAKwireless website at [www.rakwireless.com](http://www.rakwireless.com).