RAK3172 Module Quick Start Guide

This guide covers the following topics:

- RAK3172 as a Stand-Alone Device Using RUI3
- RAK3172 as a LoRa/LoRaWAN Modem via AT Command
- Connecting to The Things Network (TTN)
- Connecting with Chirpstack
- LoRa P2P Mode

Prerequisites

What Do You Need?

Before going through the steps in the installation guide of the RAK3172 WisDuo LPWAN Module, make sure to prepare the necessary items listed below:

Hardware

- RAK3172 WisDuo LPWAN Module
- Computer
- USB to UART TTL adapter

Software

- Download and install the Arduino IDE .

> ⚠ WARNING
>
> ***If you are using Windows 10***. Do ***NOT*** install the Arduino IDE from the Microsoft App Store. Instead, install the original Arduino IDE from the Arduino official website. The Arduino app from the Microsoft App Store has problems using third-party Board Support Packages.

- Add RAK3172 as a supported board in Arduino IDE by updating Board Manager URLs in **Preferences** settings of Arduino IDE with the JSON URL below.

```json
https://raw.githubusercontent.com/RAKWireless/RAKwireless-Arduino-BSP-Index/main/package_rakwirel
```

After that, you can then add **RAKwireless RUI STM32 Boards** via Arduino board manager.

- RAK Serial Port Tool

List of Acronyms

Acronym	Definition
DFU	Device Firmware Upgrade
JTAG	Joint Test Action Group
LoRa	Long Range
OTAA	Over-The-Air-Activation (OTAA)
ABP	Activation-By-Personalization (ABP)
TTN	The Things Network
DEVEUI	Device EUI (Extended Unique Identification)
APPEUI	Application EUI (Extended Unique Identification)
APPKEY	Application Key
DEVADDR	Device Address
NWKSKEY	Network Session Key
APPSKEY	Application Session Key
P2P	Point-to-Point
MSB	Most Significant Bit
LNS	LoRaWAN Network Service

Product Configuration

RAK3172 as a Stand-Alone Device Using RUI3

Hardware Setup

The RAK3172 requires a few hardware connections before you can make it work. The bare minimum requirement is to have the power section properly configured, reset button, antenna, and USB connection.

⚠️ **WARNING**

Firmware update is done via UART2 pins. If you will connect the module to an external device that will be interfacing with UART2, take extra precautions in your board design to ensure you can still perform FW update to it. There should be a way in your board design that can disconnect the external device to RAK3172 UART2 before connecting the module to the PC (via USB-UART converter) for the FW update process.

An alternative option to update firmware aside from UART2 is to use SWD pins (SWCLK & SWDIO). This method will require you to use external tools like ST-LINK and RAKDAP1.

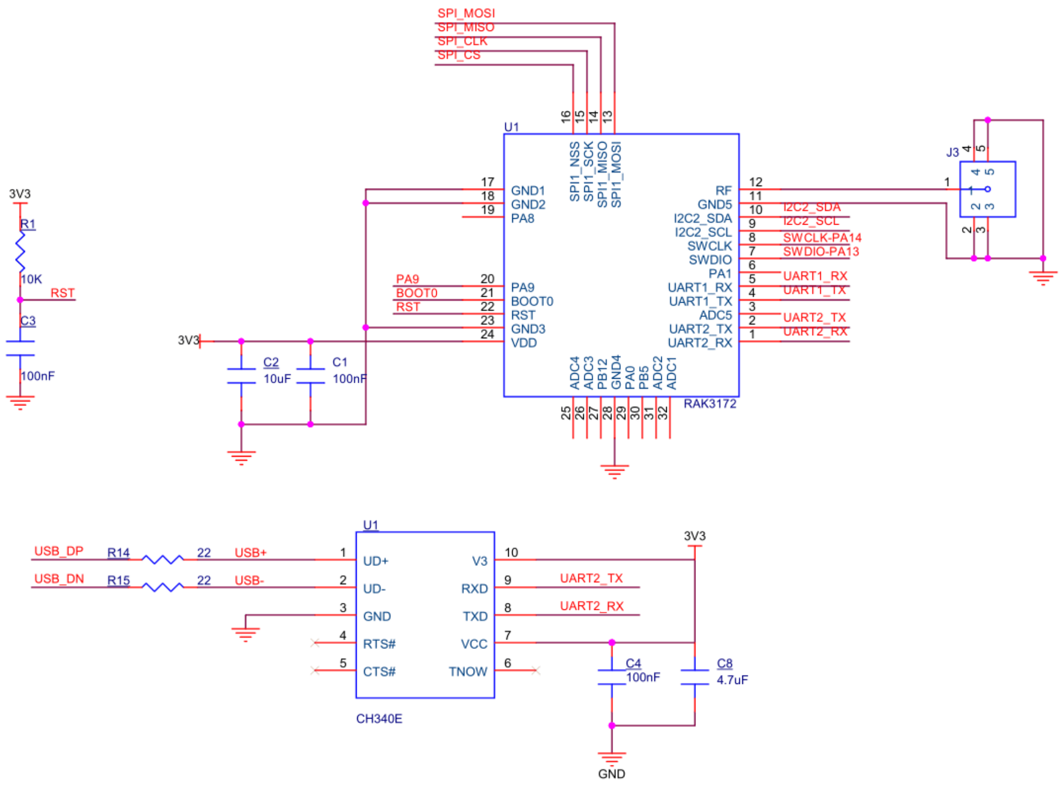


Figure 1: RAK3172 Minimum Schematic

Ensure that the antenna is properly connected to have a good LoRa signal. Also, note that you can damage the RF section of the chip if you power the module without an antenna connected to the IPEX connector.

Figure 2: LoRa Antenna

RAK3172 has a module variant with an IPEX connector where you can connect the LoRa antenna, as shown in **Figure 3**. If the RAK3172 module you ordered is the variant with no IPEX connector, you need to ensure that there is an external antenna connected to the **RF pin** (Pin 12) of the module.

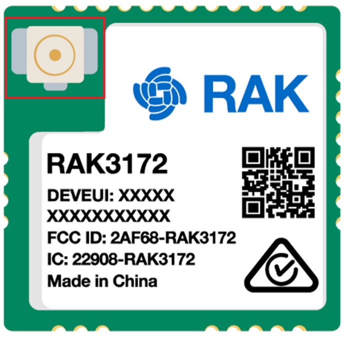


Figure 3: IPEX Connector of RAK3172 for LoRa Antenna

📝 NOTE

Detailed information about the RAK3172 LoRa antenna can be found on the antenna datasheet .

⚠️ WARNING

When using the LoRa transceiver, make sure that an antenna is always connected. Using this transceiver without an antenna can damage the module.

Software Setup

The default firmware of RAK3172 is based on RUI3, which allows you to develop your own custom firmware to connect sensors and other peripherals to it. To develop using your custom firmware using the Arduino IDE, you need to first add RAKwireless RUI STM32 boards to the Arduino board manager, which will be discussed in this guide. You can then use [RUI3 APIs] (https://docs.rakwireless.com/RUI3/Arduino-API/) for your intended application. You can upload the custom firmware via UART. The AT commands of RAK3172 are still available even if you compile custom firmware via RUI3. You can send AT commands via a UART2 connection.

RAK3172 RUI3 Board Support Package in Arduino IDE

If you don't have an Arduino IDE yet, you can download it on the Arduino official website and follow the installation procedure in the miscellaneous section of this document.

📝 NOTE

For Windows 10 and up users: If your Arduino IDE is installed from the Microsoft App Store, you need to reinstall your Arduino IDE by getting it from the Arduino official website. The Arduino app from the Microsoft App Store has problems using third-party Board Support Packages.

Once the Arduino IDE has been successfully installed, you can now configure the IDE to add the RAK3172 to its board selection by following these steps.

1. Open Arduino IDE and go to **File** > **Preferences**.

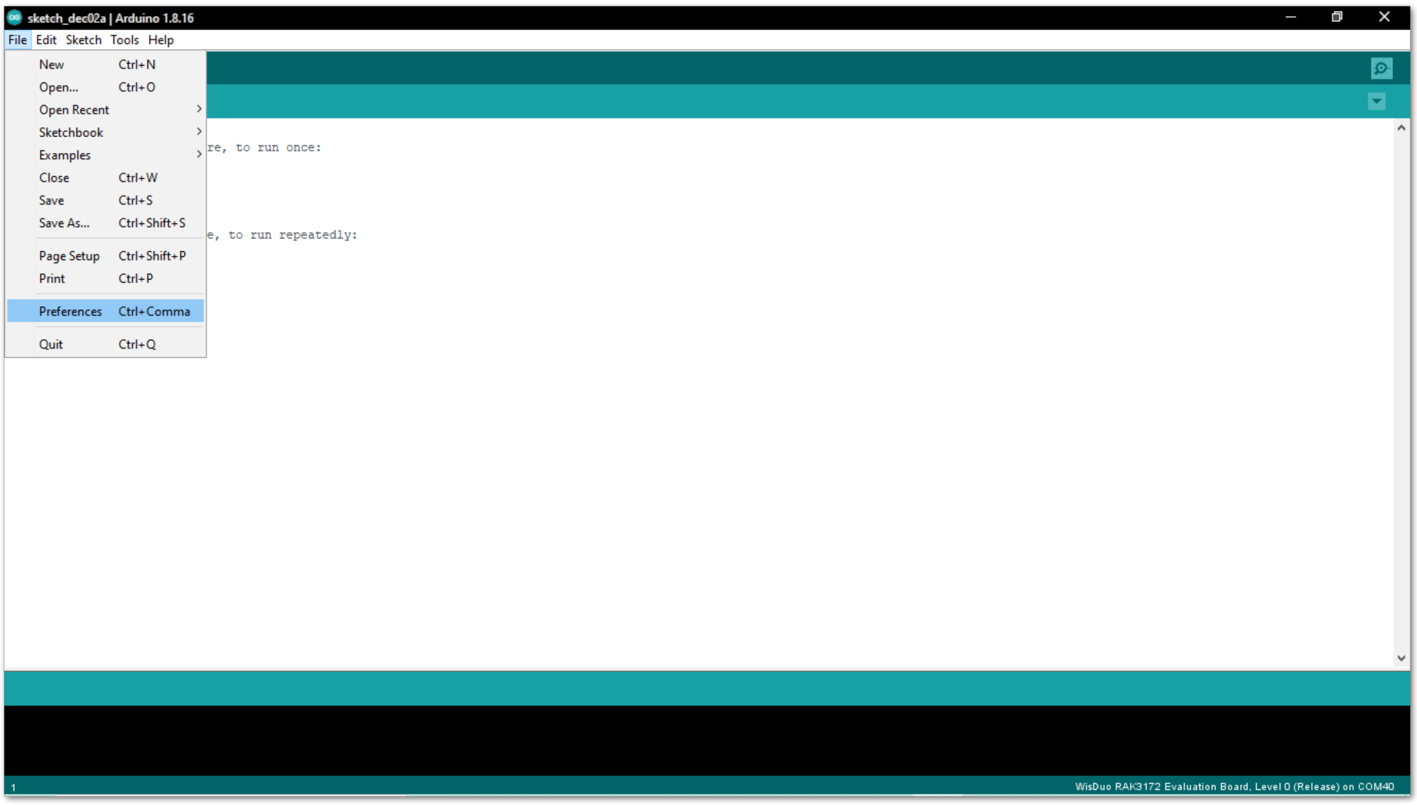
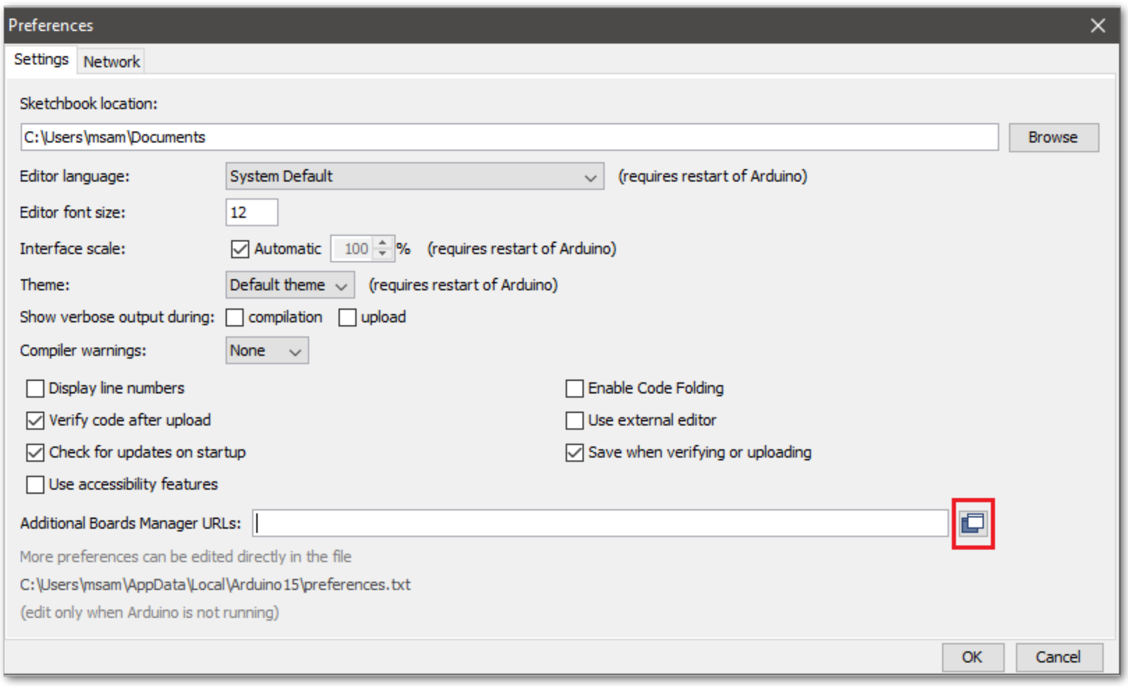


Figure 4: Arduino preferences

2. To add the RAK3172 to your Arduino Boards list, edit the **Additional Board Manager URLs**. Click the icon, as shown in **Figure 5**.



Sketchbook location: C:\Users\msam\Documents

Editor language: System Default (requires restart of Arduino)

Editor font size: 12

Interface scale: ☑ Automatic 100 % (requires restart of Arduino)

Theme: Default theme (requires restart of Arduino)

Show verbose output during: □ compilation □ upload

Compiler warnings: None

□ Display line numbers
☑ Verify code after upload
☑ Check for updates on startup
□ Use accessibility features
□ Enable Code Folding
□ Use external editor
☑ Save when verifying or uploading

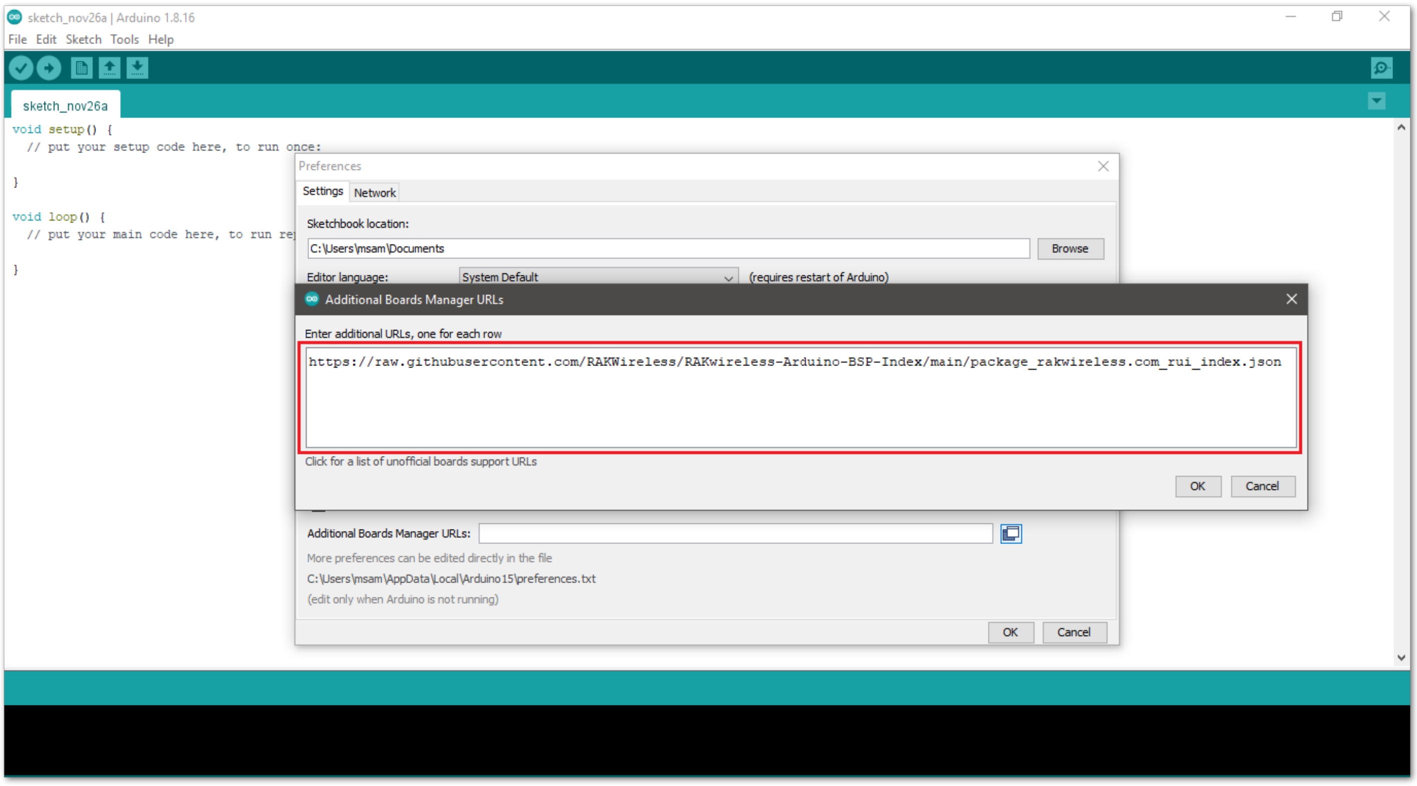
Additional Boards Manager URLs:

More preferences can be edited directly in the file
C:\Users\msam\AppData\Local\Arduino15\preferences.txt
(edit only when Arduino is not running)

Figure 5: Modifying Additional Board Manager URLs

3. Copy the URL below and paste it on the field, as shown in **Figure 6**. If there are other URLs already there, just add them on the next line. After adding the URL, click **OK**.

```json
https://raw.githubusercontent.com/RAKWireless/RAKwireless-Arduino-BSP-Index/main/package_rakwirel
```


sketch_nov26a | Arduino 1.8.16
File Edit Sketch Tools Help
sketch_nov26a
Preferences
Settings
Network
Browse
Additional Boards Manager URLs
Click for a list of unofficial boards support URLs
OK
Cancel
OK
Cancel

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run re

}
```

Sketchbook location: C:\Users\msam\Documents

Editor language: System Default (requires restart of Arduino)

Enter additional URLs, one for each row

```
https://raw.githubusercontent.com/RAKWireless/RAKwireless-Arduino-BSP-Index/main/package_rakwireless.com_rui_index.json
```

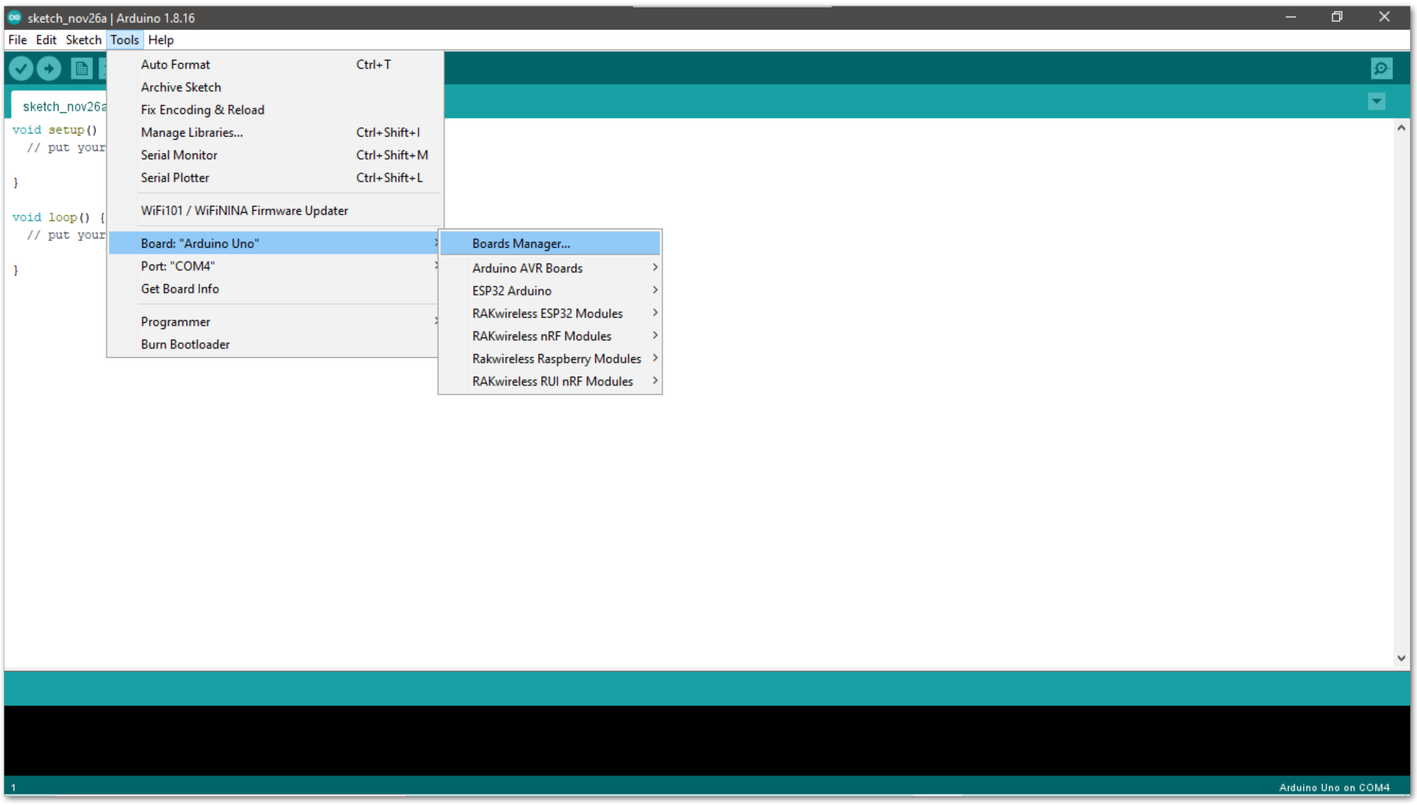
Additional Boards Manager URLs:

More preferences can be edited directly in the file
C:\Users\msam\AppData\Local\Arduino15\preferences.txt
(edit only when Arduino is not running)

Figure 6: Add additional board manager URLs

4. Restart the Arduino IDE.

5. Open the Boards Manager from the Tools Menu.

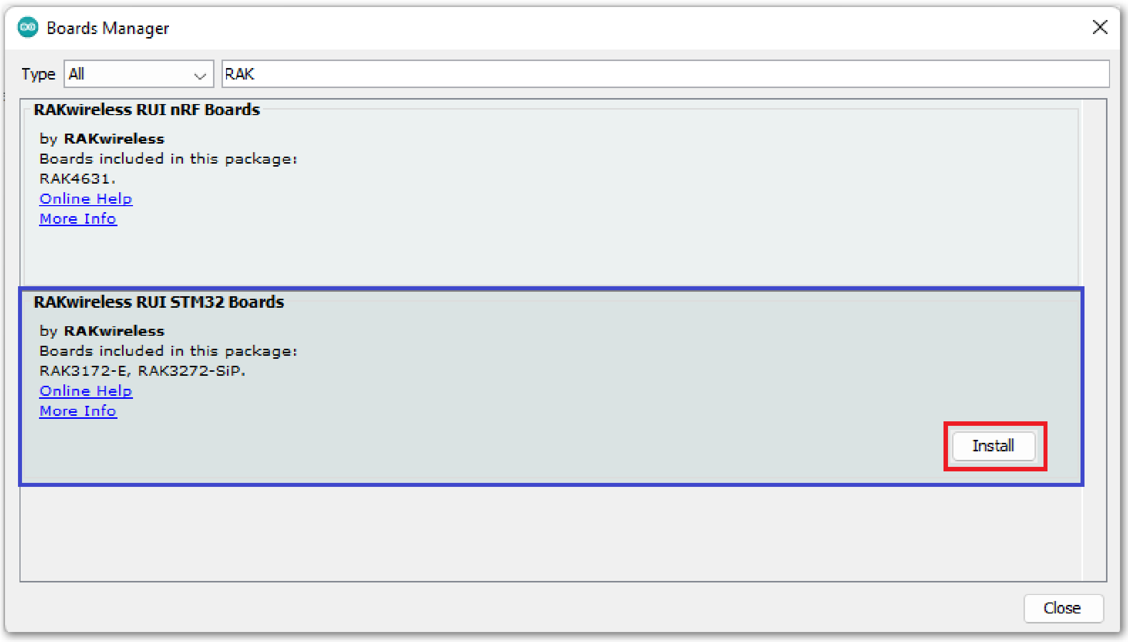

sketch_nov26a | Arduino 1.8.16
File Edit Sketch Tools Help
sketch_nov26a
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino Uno"
Port: "COM4"
Get Board Info
Programmer
Burn Bootloader
Boards Manager...
Arduino AVR Boards
ESP32 Arduino
RAKwireless ESP32 Modules
RAKwireless nRF Modules
Rakwireless Raspberry Modules
RAKwireless RUI nRF Modules
1
Arduino Uno on COM4

```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Figure 7: Opening Arduino boards manager

6. Write `RAK` in the search bar, as shown in **Figure 8**. This will show the available RAKwireless module boards that you can add to your Arduino Board list.

7. Click on the area highlighted in blue to select **RAKwireless RUI STM32 Boards**. Install the latest version of the **RAKwireless RUI STM32 Boards** by clicking on **Install** button.



Boards Manager

Type All RAK

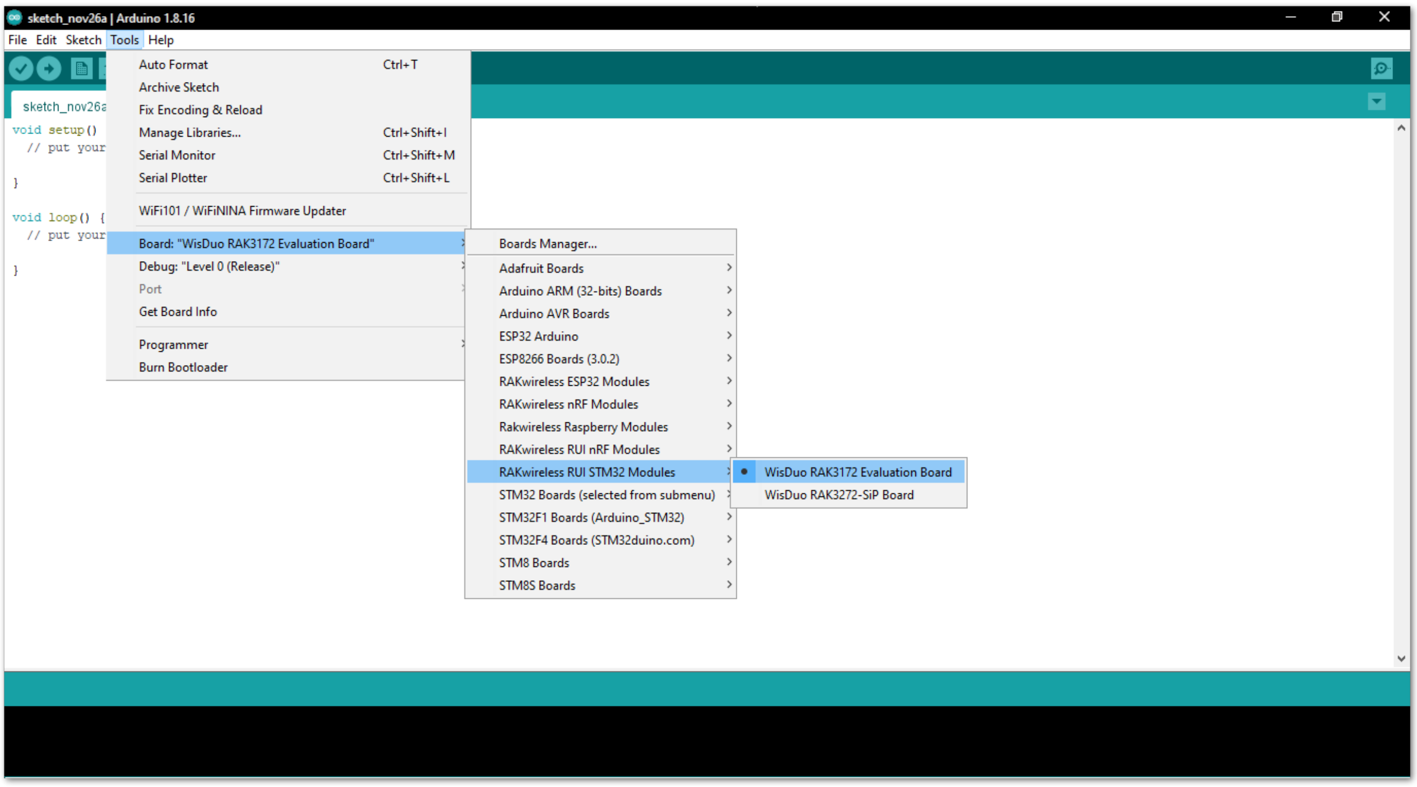
RAKwireless RUI nRF Boards
by RAKwireless
Boards included in this package:
RAK4631.
Online Help
More Info

RAKwireless RUI STM32 Boards
by RAKwireless
Boards included in this package:
RAK3172-E, RAK3272-SiP.
Online Help
More Info

Figure 8: Installing RAKwireless RUI STM32 Boards

Configure RAK3172 on Boards Manager

8. Once the BSP is installed, select **Tools** > **Boards Manager** > **RAKWireless RUI STM32 Modules** > **WisDuo RAK3172 Evaluation Board**. The RAK3172 Evaluation board uses the RAK3172 WisDuo module.



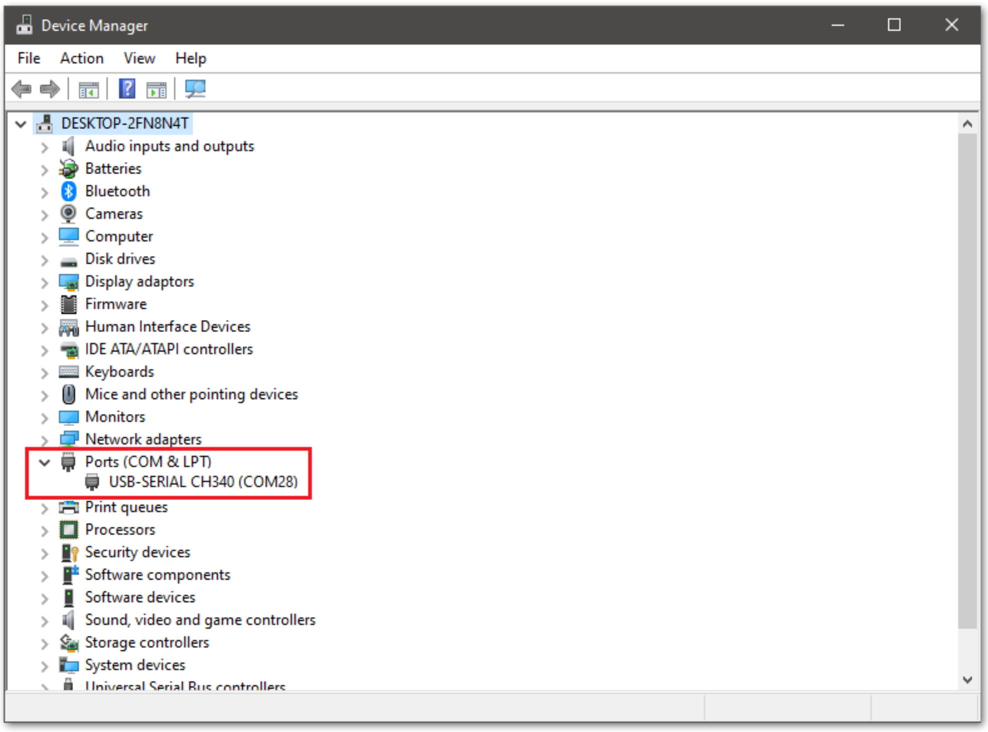
sketch_nov26a | Arduino 1.8.16
File Edit Sketch Tools Help
sketch_nov26a
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "WisDuo RAK3172 Evaluation Board"
Debug: "Level 0 (Release)"
Port
Get Board Info
Programmer
Burn Bootloader
Boards Manager...
Adafruit Boards
Arduino ARM (32-bits) Boards
Arduino AVR Boards
ESP32 Arduino
ESP8266 Boards (3.0.2)
RAKwireless ESP32 Modules
RAKwireless nRF Modules
Rakwireless Raspberry Modules
RAKwireless RUI nRF Modules
RAKwireless RUI STM32 Modules
STM32 Boards (selected from submenu)
STM32F1 Boards (Arduino_STM32)
STM32F4 Boards (STM32duino.com)
STM8 Boards
STM8S Boards
● WisDuo RAK3172 Evaluation Board
WisDuo RAK3272-SiP Board

```
void setup() {
  // put your
}

void loop() {
  // put your
}
```

Figure 9: Selecting RAK3172 Module

RAK3172 COM Port on Device Manager

Connect the RAK3172 via UART and check RAK3172 COM Port using Windows **Device Manager**. Double-click the reset button if the module is not detected.



Device Manager

File Action View Help

DESKTOP-2FN8N4T
- Audio inputs and outputs
- Batteries
- Bluetooth
- Cameras
- Computer
- Disk drives
- Display adaptors
- Firmware
- Human Interface Devices
- IDE ATA/ATAPI controllers
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Ports (COM & LPT)
  - USB-SERIAL CH340 (COM28)
- Print queues
- Processors
- Security devices
- Software components
- Software devices
- Sound, video and game controllers
- Storage controllers
- System devices
- Universal Serial Bus controllers

Figure 10: Device manager ports (COM & LPT)

Compile an Example with Arduino LED Breathing

1. After completing the steps on adding your RAK3172 to the Arduino IDE, you can now try to run a simple program to test your setup. You need to add two LEDs to the bare minimum schematic of the RAK3172 module, as shown in **Figure 11**.

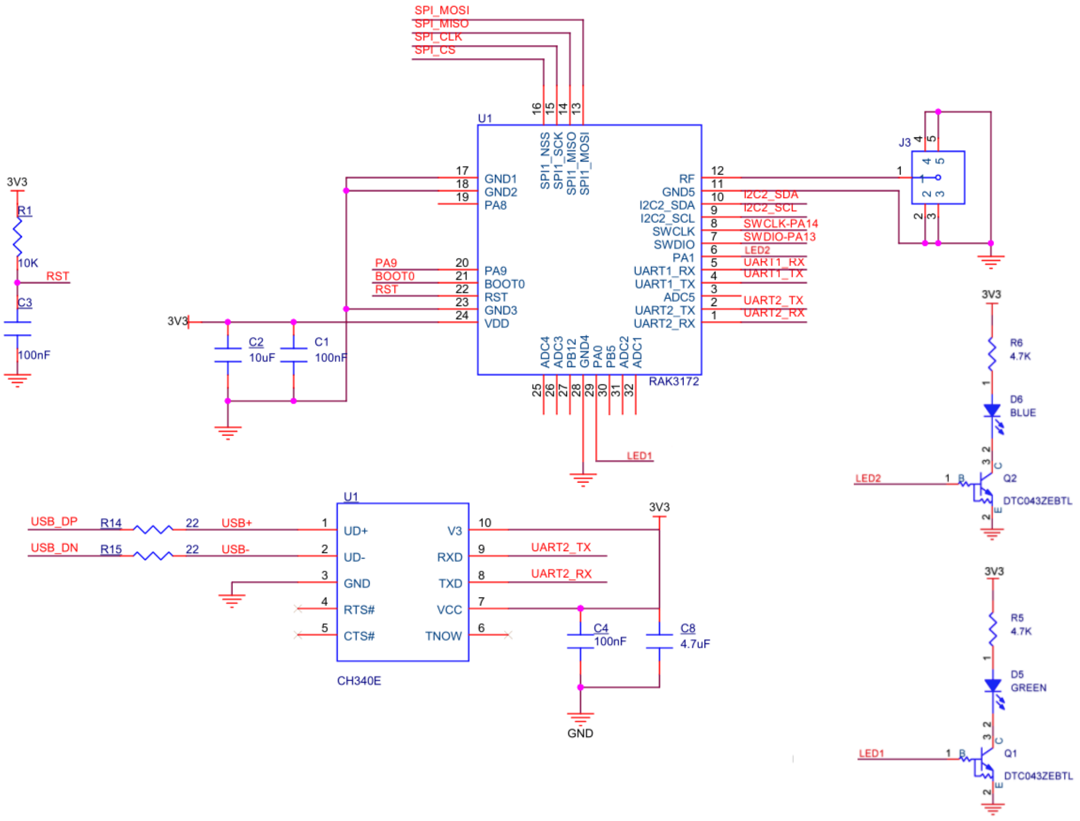


Figure 11: RAK3172 with two LEDs

2. Launch Arduino IDE and configure WisDuo RAK3172 Evaluation Board on board selection. See **Figure 9**.
3. Connect the RAK3172 via UART and check RAK3172 COM Port. See **Figure 10**.
4. Open the **Tools** Menu and select a COM port. **COM28** is currently used.

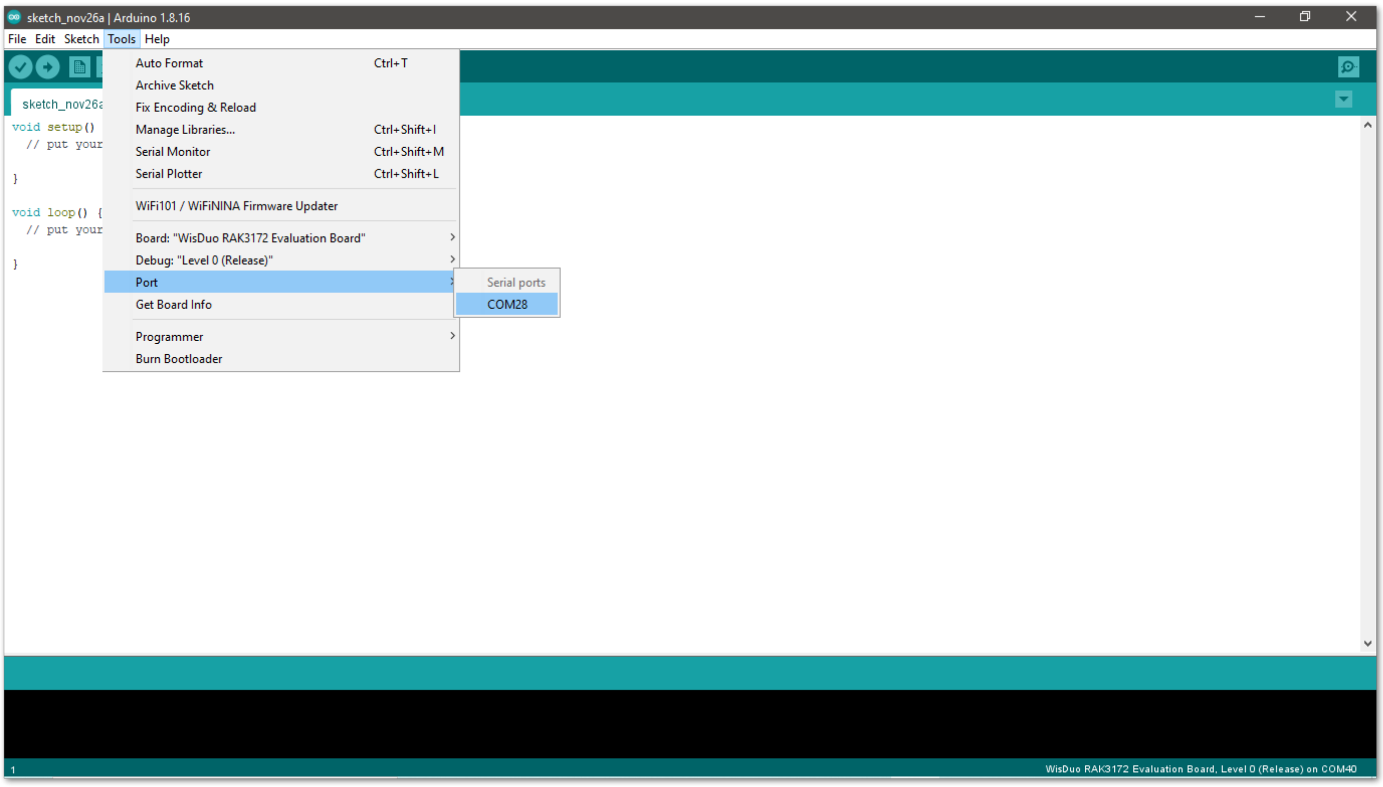

sketch_nov26a | Arduino 1.8.16
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "WisDuo RAK3172 Evaluation Board"
Debug: "Level 0 (Release)"
Port
Get Board Info
Programmer
Burn Bootloader
Serial ports
COM28
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM40

Figure 12: Select COM port

5. You can see the serial monitor icon and click it to connect the COM port.

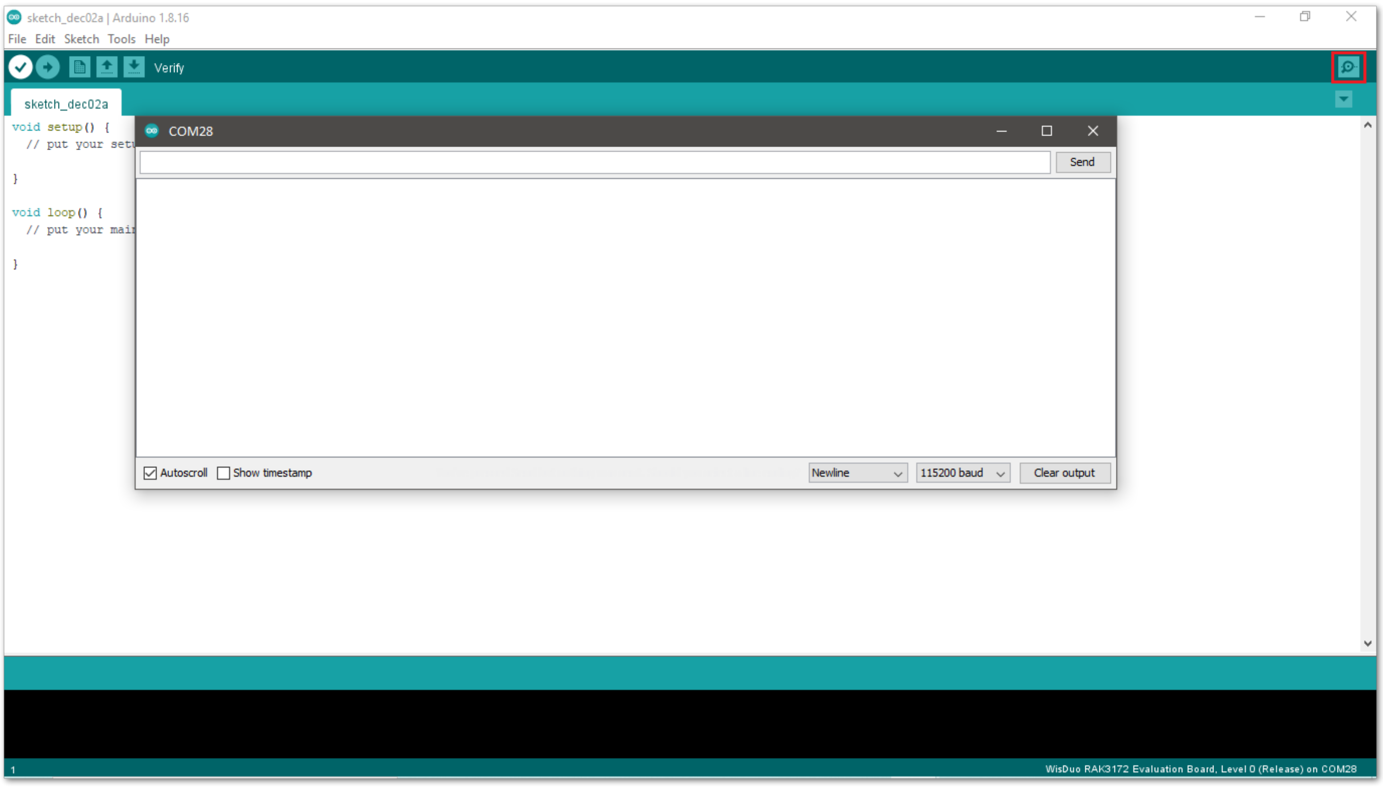
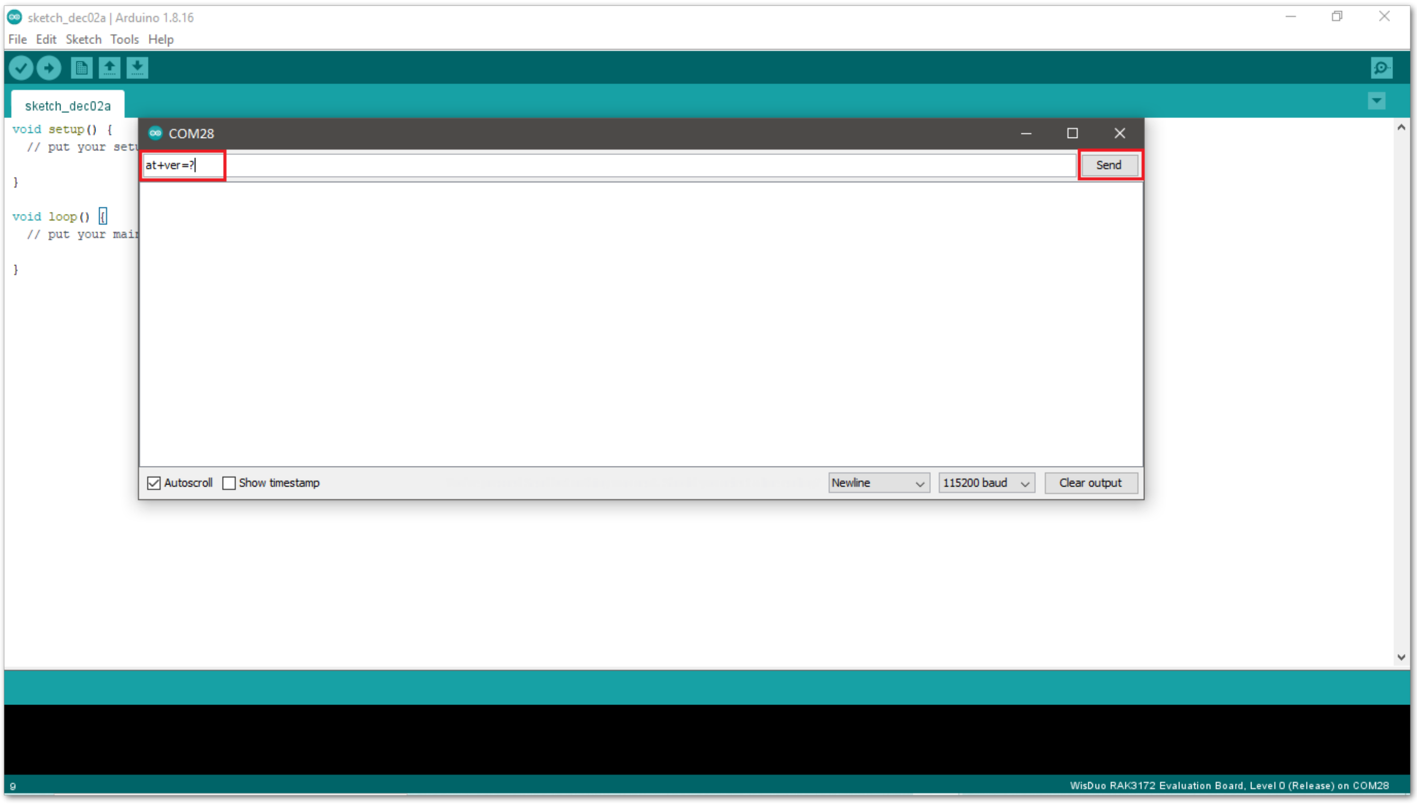

sketch_dec02a | Arduino 1.8.16
File Edit Sketch Tools Help
Verify
sketch_dec02a
COM28
Send
☑ Autoscroll ☐ Show timestamp
Newline
115200 baud
Clear output
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM28

Figure 13: Open Arduino serial monitor

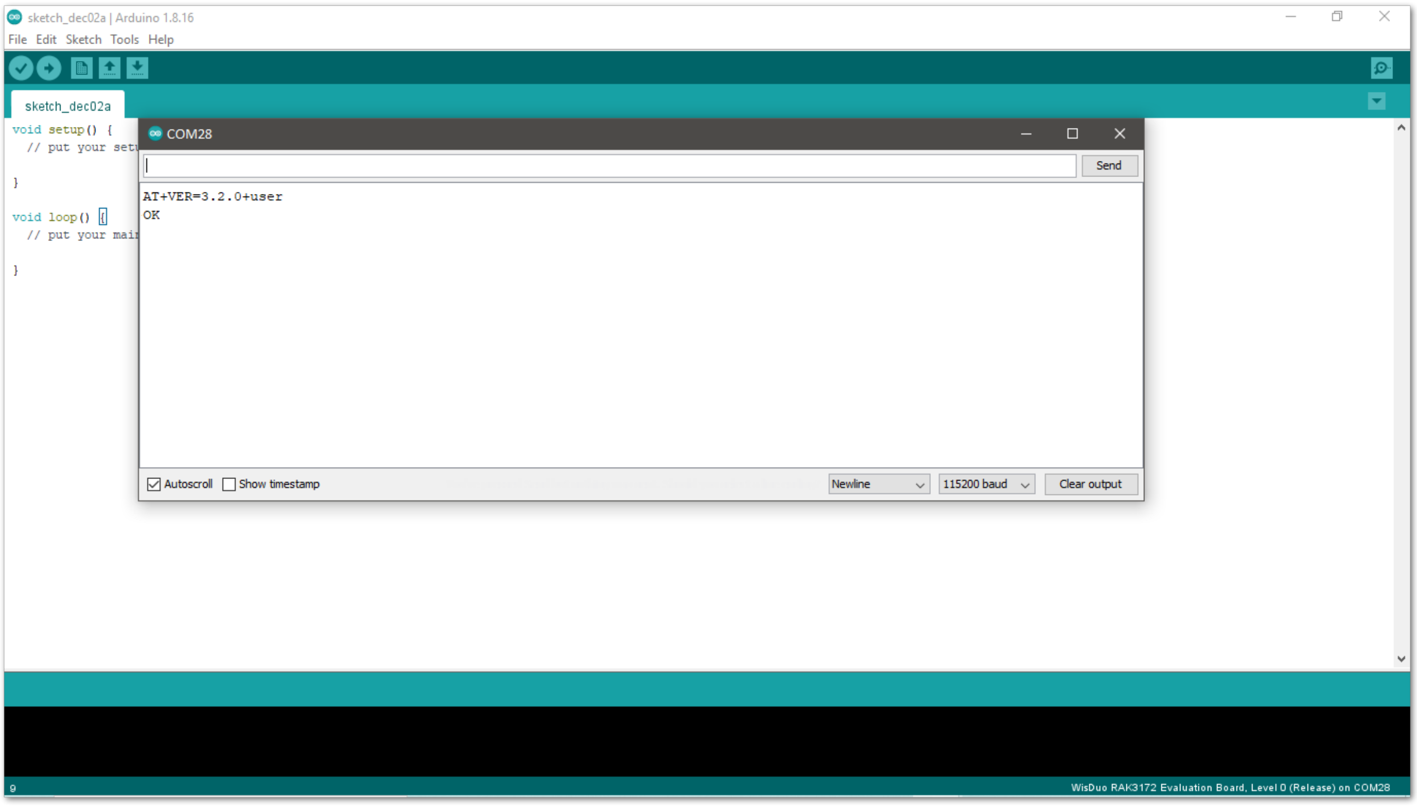
6. If the connection is successful, you can send AT Commands to RAK3172. For example: To check the RUI version, type `AT+VER=?` on the text area, then click on the **Send** button, as shown in **Figure 14**.


sketch_dec02a | Arduino 1.8.16
File Edit Sketch Tools Help
sketch_dec02a
COM28
Send
Autoscroll
Show timestamp
Newline
115200 baud
Clear output
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM28

```
void setup() {
  // put your setu
}

void loop() {
  // put your mai
}
```

```
at+ver=?
```

Figure 14: Send AT command


sketch_dec02a | Arduino 1.8.16
File Edit Sketch Tools Help
sketch_dec02a
COM28
Send
Autoscroll
Show timestamp
Newline
115200 baud
Clear output
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM28

```
AT+VER=3.2.0+user
OK
```

Figure 15: Arduino serial monitor COM28

7. Open **Arduino_Led_Breathing** example code.

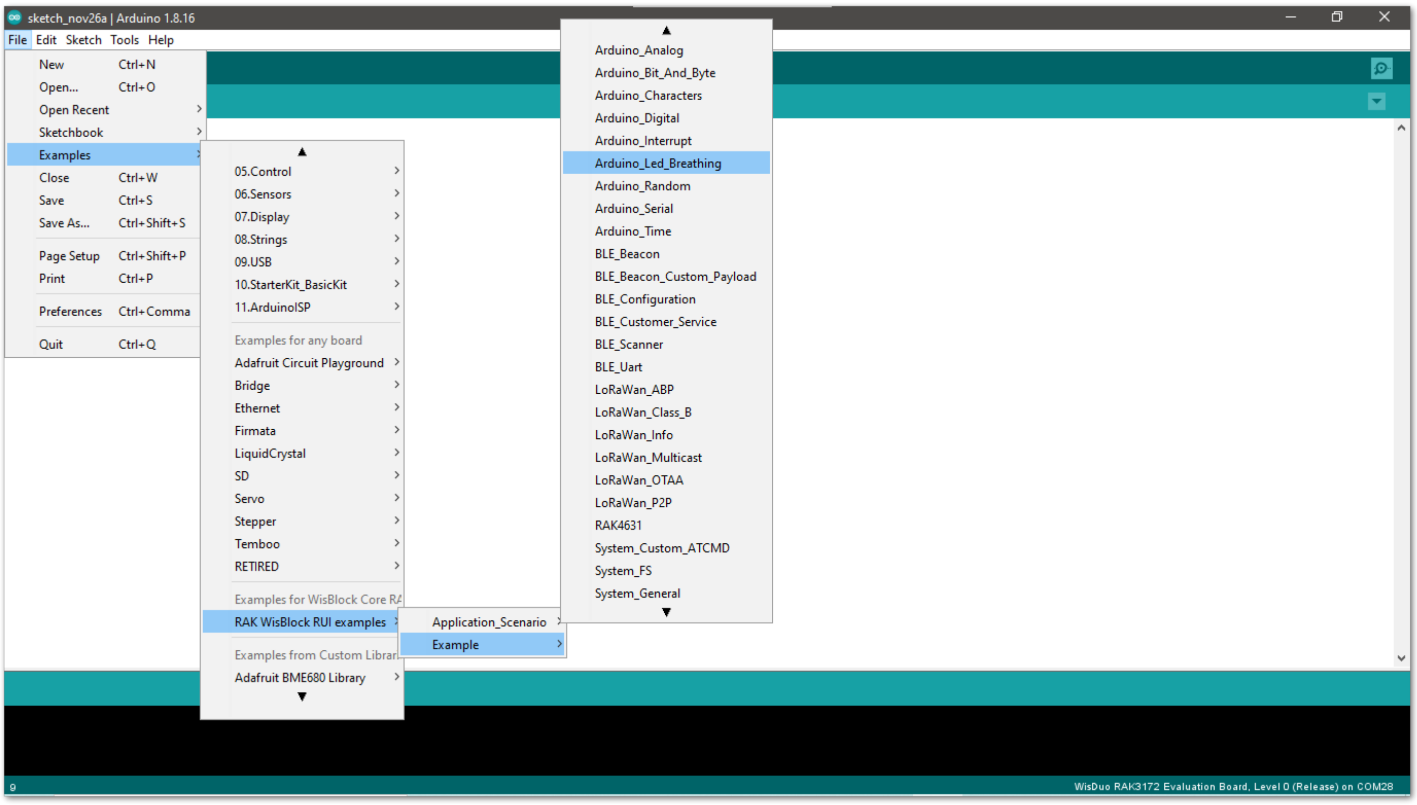
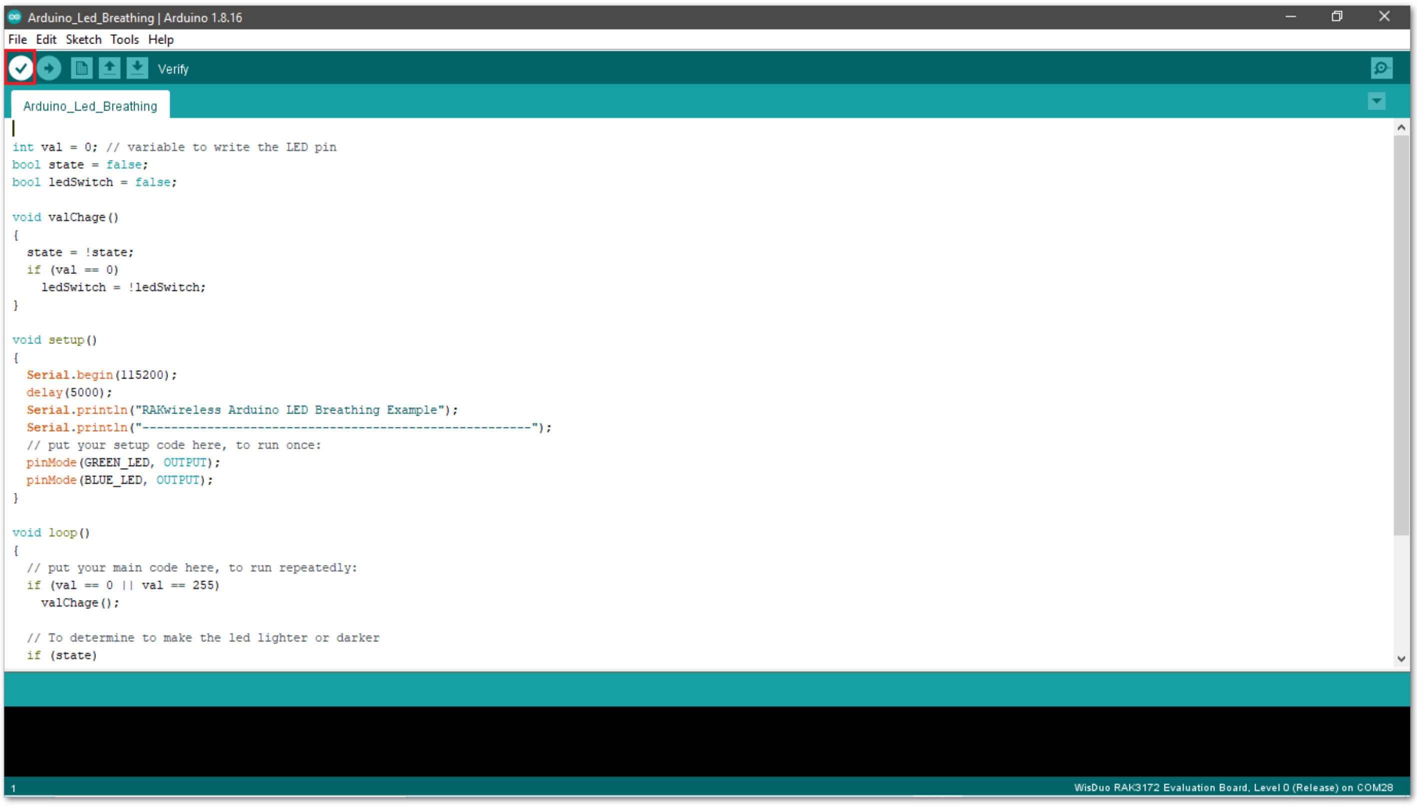


Figure 16: Arduino Led Breathing example

8. Click on the **Verify** icon to check if you have successfully compiled the example code.

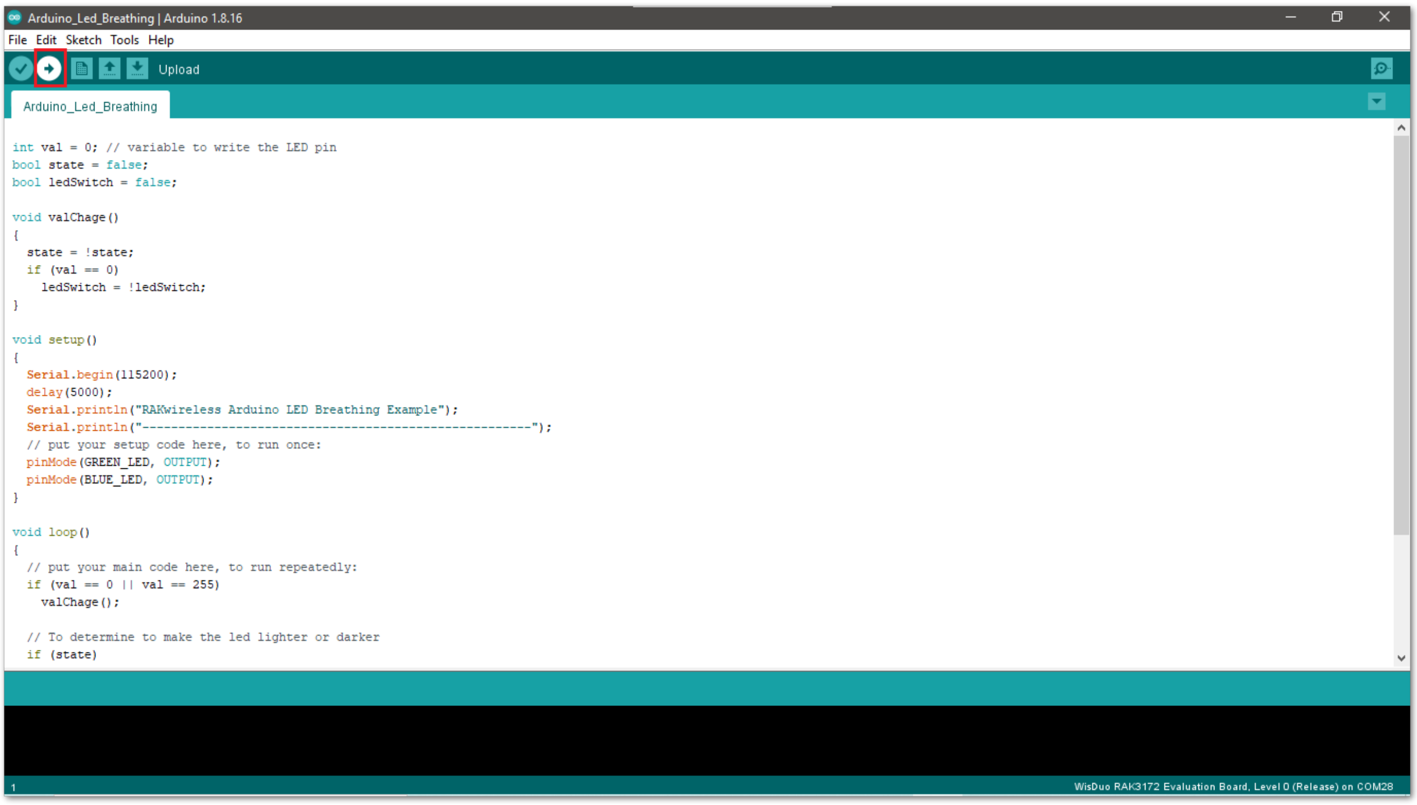

Arduino_Led_Breathing | Arduino 1.8.16
File Edit Sketch Tools Help
Verify
Arduino_Led_Breathing
1
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM28

```
int val = 0; // variable to write the LED pin
bool state = false;
bool ledSwitch = false;

void valChage()
{
  state = !state;
  if (val == 0)
    ledSwitch = !ledSwitch;
}

void setup()
{
  Serial.begin(115200);
  delay(5000);
  Serial.println("RAKwireless Arduino LED Breathing Example");
  Serial.println("------------------------------------------------------");
  // put your setup code here, to run once:
  pinMode(GREEN_LED, OUTPUT);
  pinMode(BLUE_LED, OUTPUT);
}

void loop()
{
  // put your main code here, to run repeatedly:
  if (val == 0 || val == 255)
    valChage();

  // To determine to make the led lighter or darker
  if (state)
```

Figure 17: Verify the example code

9. Click the **Upload** icon to send the compiled firmware to your RAK3172 module.

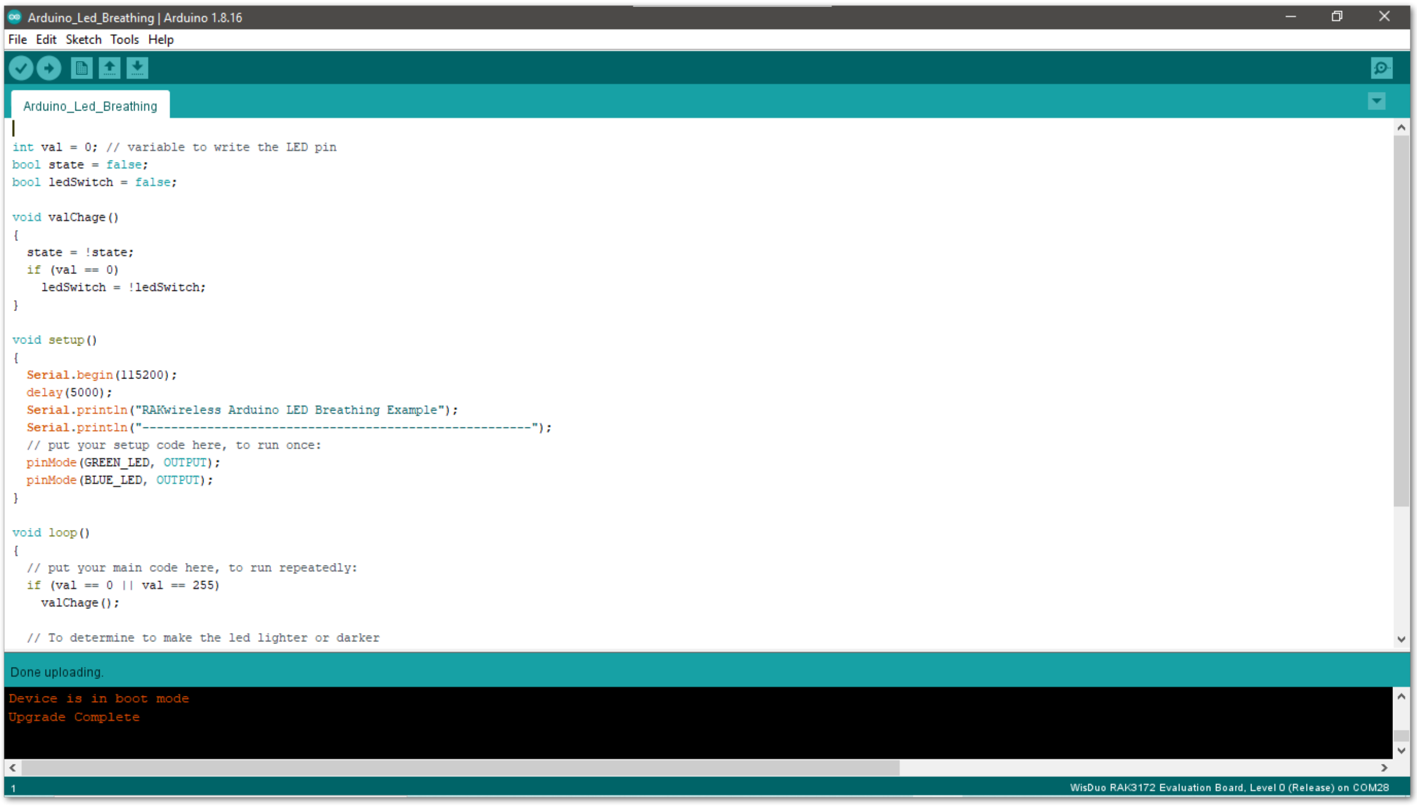
> 📝 **NOTE:**
>
> RAK3172 should automatically go to BOOT mode when the firmware is uploaded via Arduino IDE.
>
> If BOOT mode is not initiated, pull to ground the RESET pin twice (or double click the reset button if available) to force BOOT mode.


Arduino_Led_Breathing | Arduino 1.8.16
File Edit Sketch Tools Help
Upload
Arduino_Led_Breathing
1
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM28

```
int val = 0; // variable to write the LED pin
bool state = false;
bool ledSwitch = false;

void valChage()
{
  state = !state;
  if (val == 0)
    ledSwitch = !ledSwitch;
}

void setup()
{
  Serial.begin(115200);
  delay(5000);
  Serial.println("RAKwireless Arduino LED Breathing Example");
  Serial.println("------------------------------------------------------");
  // put your setup code here, to run once:
  pinMode(GREEN_LED, OUTPUT);
  pinMode(BLUE_LED, OUTPUT);
}

void loop()
{
  // put your main code here, to run repeatedly:
  if (val == 0 || val == 255)
    valChage();

  // To determine to make the led lighter or darker
  if (state)
```

Figure 18: Upload the example code

10. If the upload is successful, you will see the **Upgrade Complete** message.


Arduino_Led_Breathing | Arduino 1.8.16
File Edit Sketch Tools Help
Arduino_Led_Breathing
1
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM28

```
int val = 0; // variable to write the LED pin
bool state = false;
bool ledSwitch = false;

void valChage()
{
  state = !state;
  if (val == 0)
    ledSwitch = !ledSwitch;
}

void setup()
{
  Serial.begin(115200);
  delay(5000);
  Serial.println("RAKwireless Arduino LED Breathing Example");
  Serial.println("------------------------------------------------------");
  // put your setup code here, to run once:
  pinMode(GREEN_LED, OUTPUT);
  pinMode(BLUE_LED, OUTPUT);
}

void loop()
{
  // put your main code here, to run repeatedly:
  if (val == 0 || val == 255)
    valChage();

  // To determine to make the led lighter or darker
```

```
Done uploading.
Device is in boot mode
Upgrade Complete
```

Figure 19: Device programmed successfully

11. After the Device Programmed is completed, you will see that LEDs are blinking.

RAK3172 I/O Pins and Peripherals

This section discusses how to use and access RAK3172 pins using the RUI3 API. It shows basic code on using digital I/O, analog input, UART, and I2C.

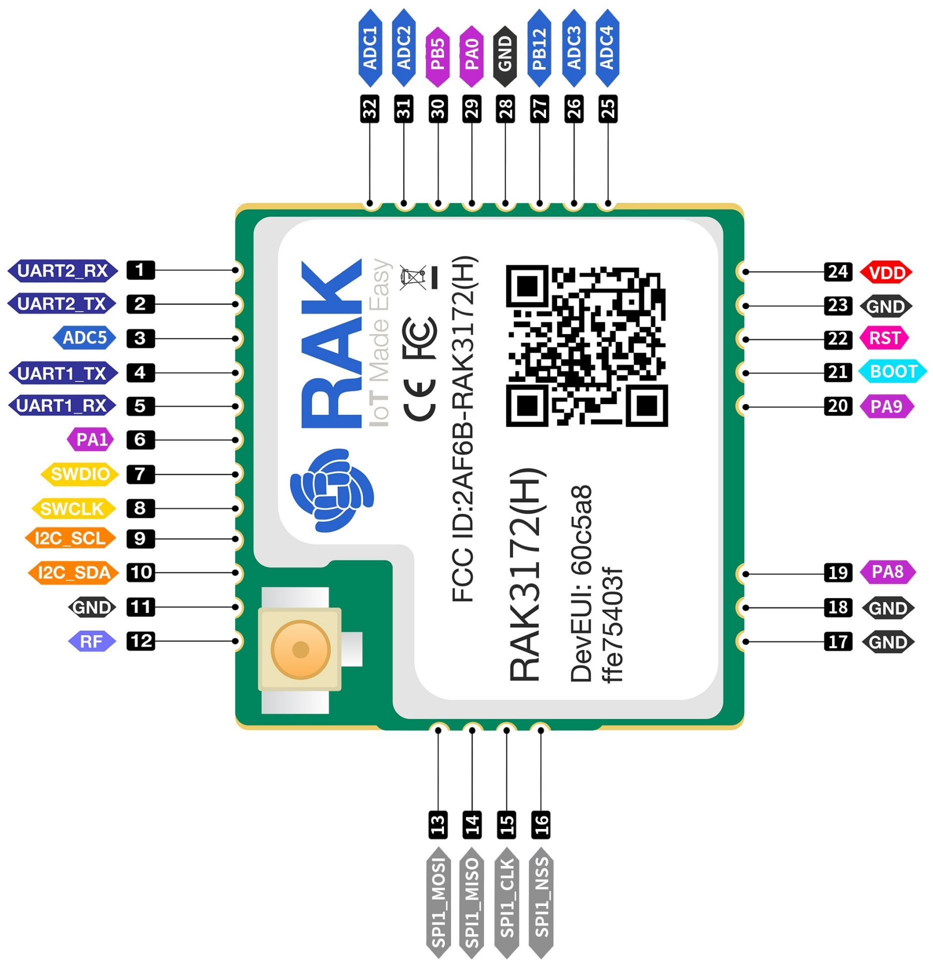


Figure 20: Available Peripherals and Digital I/O pins in RAK3172 module

How to Use Digital I/O

You can use any of the pins below as Digital Pin.

Pin Name	Alternative Pin Usage
PA0	
PA1	
PA4	SPI
PA5	SPI
PA6	SPI
PA7	SPI
PA8	
PA9	I2C_SCL
PA15	
PB2	
PB3	ADC1
PB4	ADC2
PB5	
PB6	UART1_TX
PB7	UART1_RX

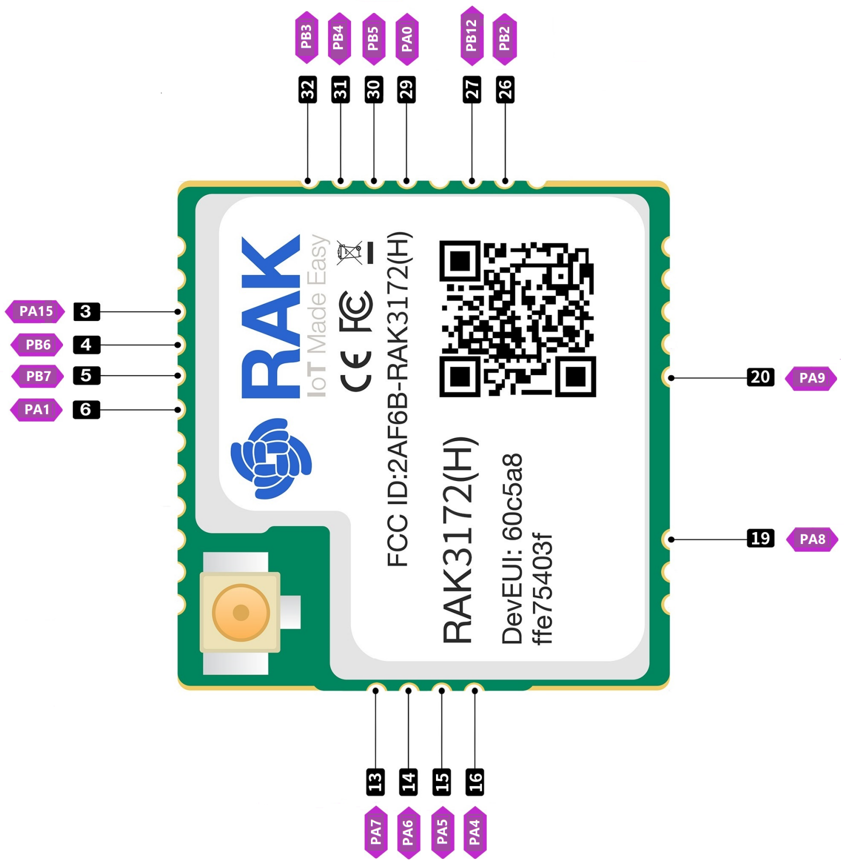


Figure 21: Available Digital I/O pins in RAK3172 module

The pins listed below must not be used.

Pin name	Pin Usage
PA2	UART2_TX
PA3	UART2_RX
PA13	SWDIO
PA14	SWCLK
PB8	RAK3172 Internal
PB12	Internal 10k pull-up resistor for RAK3172 high frequency variant (8xx - 9xx Mhz) or pull-down resistor for RAK3172 low frequency variant (4xx Mhz)

- Use Arduino digitalRead to read the value from a specified Digital I/O pin, either HIGH or LOW.
- Use Arduino digitalWrite to write a HIGH or a LOW value to a Digital I/O pin.

> 📝 **NOTE:**
>
> The GPIO Pin Name is the one to be used on the digitalRead and digitalWrite and NOT the pin numbers.

Example code

```c
void setup()
{
  pinMode(PA0, OUTPUT); //Change the P0_04 to any digital pin you want. Also, you can set this to
}

void loop()
{
  digitalWrite(PA0,HIGH); //Change the PA0 to any digital pin you want. Also, you can set this to
  delay(1000); // delay for 1 second
  digitalWrite(PA0,LOW); //Change the PA0 to any digital pin you want. Also, you can set this to
  delay(1000); // delay for 1 second
}
```

How to Use Analog Input

You can use any of the pins below as Analog Input.

Analog Port	Pin Name
ADC1	PB3
ADC2	PB4

Use Arduino analogRead to read the value from the specified Analog Input pin.

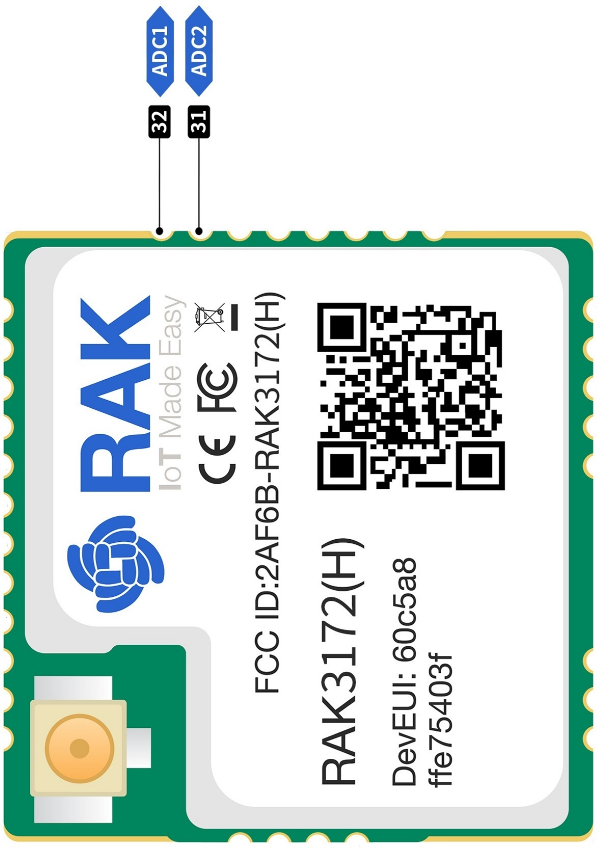


Figure 22: Available Analog pins in RAK3172

Example code

```c
#define analogPin PB3

int val = 0;  // variable to store the value read

void setup()
{
  Serial.begin(115200);
}

void loop()
{
  val = analogRead(analogPin);  // read the input pin
  Serial.println(val);          // debug value
  delay(100);
}
```

How to Use Serial Interfaces

UART

There are two UART peripherals available on the RAK3172 module. There are also different Serial Operating Modes possible in RUI3, namely Binary Mode , AT Mode , and Custom Mode .

Serial Port	Serial Instance Assignment	Default Mode
UART1 (pins 4, 5)	Serial1	Custom Mode
UART2 (pins 1, 2)	Serial	AT Command

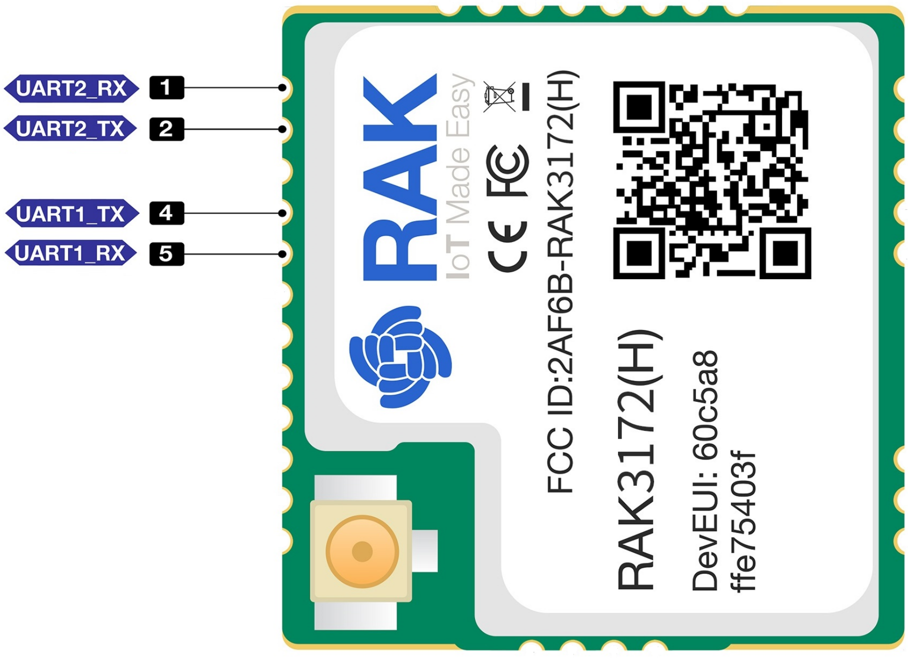


Figure 23: Available UART pins in RAK3172

Example Code

```c
void setup()
{
  Serial1.begin(115200); // use Serial1 for UART1 and Serial for UART2
                         // you can designate separate baudrate for each.
  Serial.begin(115200);
}

void loop()
{
  Serial1.println("RAK3172 UART1 TEST!");
  Serial.println("RAK3172 UART2 TEST!");
  delay(1000); // delay for 1 second
}
```

I2C

There is one I2C peripheral available on RAK3172.

I2C Pin Number	I2C Pin Name
PA12	I2C_SCL
PA11	I2C_SDA

- Use Arduino Wire library to communicate with I2C devices.

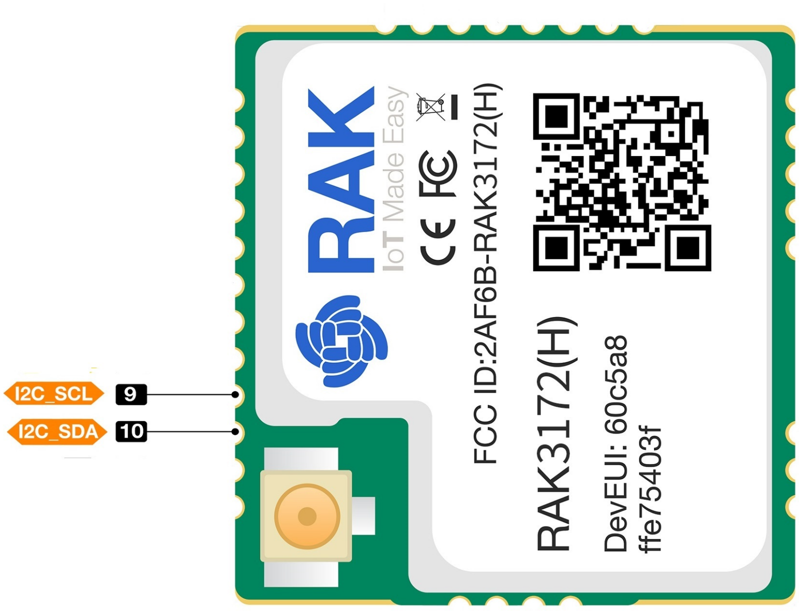


Figure 24: Available I2C pins in RAK3172

Example Code

Make sure you have an I2C device connected to specified I2C pins to run the I2C scanner code below:

```c
#include <Wire.h>

void setup()
{
  Wire.begin();

  Serial.begin(115200);
  while (!Serial);
  Serial.println("\nI2C Scanner");
}


void loop()
{
  byte error, address;
  int nDevices;

  Serial.println("Scanning...");

  nDevices = 0;
  for(address = 1; address < 127; address++ )
  {
    // The i2c_scanner uses the return value of
    // the Write.endTransmission to see if
    // a device did acknowledge to the address.
    Wire.beginTransmission(address);
    error = Wire.endTransmission();

    if (error == 0)
    {
      Serial.print("I2C device found at address 0x");
      if (address<16)
        Serial.print("0");
      Serial.print(address,HEX);
      Serial.println("  !");

      nDevices++;
    }
    else if (error==4)
    {
      Serial.print("Unknown error at address 0x");
      if (address<16)
        Serial.print("0");
      Serial.println(address,HEX);
    }
  }
  if (nDevices == 0)
    Serial.println("No I2C devices found\n");
  else
    Serial.println("done\n");

  delay(5000);           // wait 5 seconds for next scan
}
```

The Arduino Serial Monitor shows the I2C device found.

```
17:29:15.690 -> Scanning...
17:29:15.738 -> I2C device found at address 0x28  !
17:29:15.831 -> done
17:29:15.831 ->
17:29:20.686 -> Scanning...
17:29:20.733 -> I2C device found at address 0x28  !
17:29:20.814 -> done
17:29:20.814 ->
```

SPI

If your RUI3 project uses SPI, then PA4 to PA7 pins are reserved for RUI3 SPI interface.

> 📝 **NOTE:**
>
> PA13 and PA14 pins are reserved for SWD debug interface. Check the Connect to the RAK3172 section.

LoRaWAN Example

This example illustrates how to program RAK3172 module as a stand-alone LoRaWAN end-device via RUI3 Arduino APIs . To use RAK3172 module as a LoRaWAN end-device, it needs to be within reach of a working **LoRaWAN gateway** registered to a **LoRaWAN network server (LNS)** or with a built-in network server.

> 📝 **NOTE:**
>
> If you are new to LoRaWAN, here are a few good references about LoRaWAN and gateways:
>
> - LoRaWAN 101
> - What is a LoRaWAN Gateway
> - How do LoRaWAN Gateways work?
> - Things to Consider When Picking A LoRaWAN Gateway
>
> RAKwireless LoRaWAN gateway models like WisGate Edge have built-in network servers. It is also common that the LoRaWAN network server is external or in the cloud. The popular LoRaWAN network server in the cloud that you can use for free (but offers enterprise service, too) is TTN .
>
> To correctly run this example, it is necessary to configure the LoRaWAN parameters and create an OTAA application on your LoRaWAN gateway.

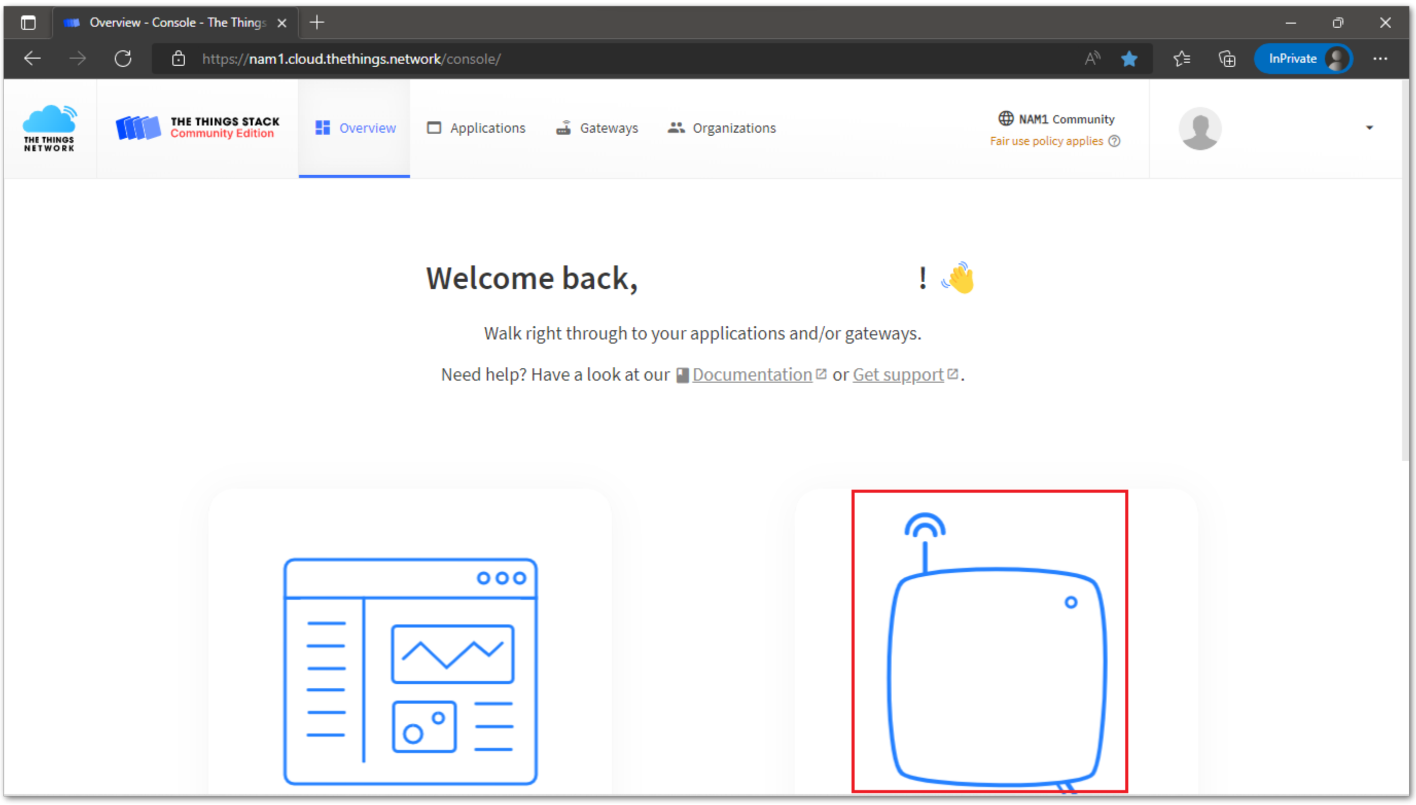
Register the LoRaWAN Gateway on TTNv3 Community Edition

After configuring your gateway, you need to register it in TTNv3:

1. Log in to the TTNv3 Network Server with a web browser.

- Europe
- North America
- Australia

2. Navigate to the **Console** page and click on **gateway** icon, as shown in **Figure 25**.



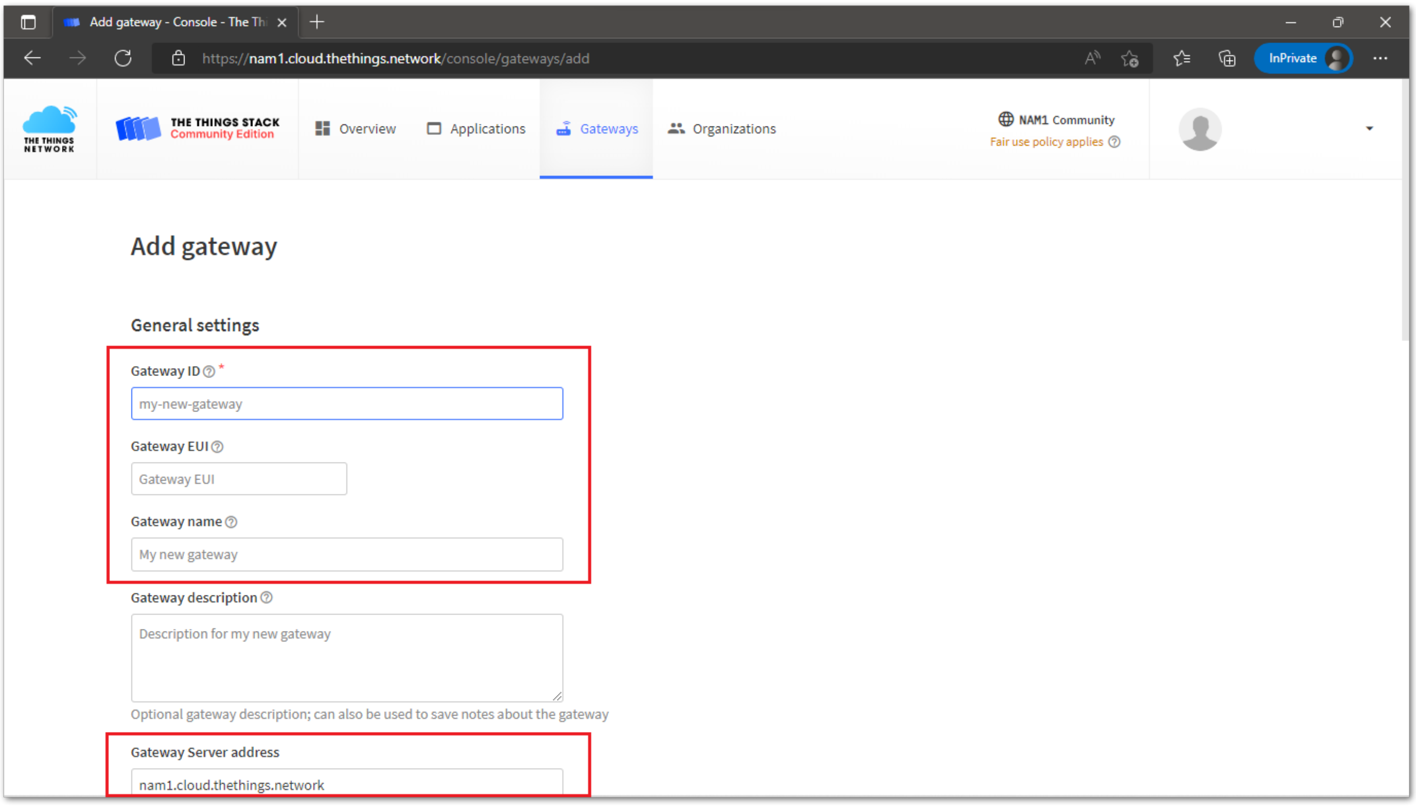
Welcome back, !

Walk right through to your applications and/or gateways.

Need help? Have a look at our Documentation or Get support.

Figure 25: TTNv3 gateway registration and configuration

3. On **General Settings**, enter the **Gateway ID**, **Gateway EUI**, and **Gateway Name**. This information is available in your LoRaWAN gateway configuration. Select the **Gateway Server address** according to the region where the LoRaWAN gateway will be installed.



Add gateway

General settings

Gateway ID *

my-new-gateway

Gateway EUI

Gateway EUI

Gateway name

My new gateway

Gateway description

Description for my new gateway

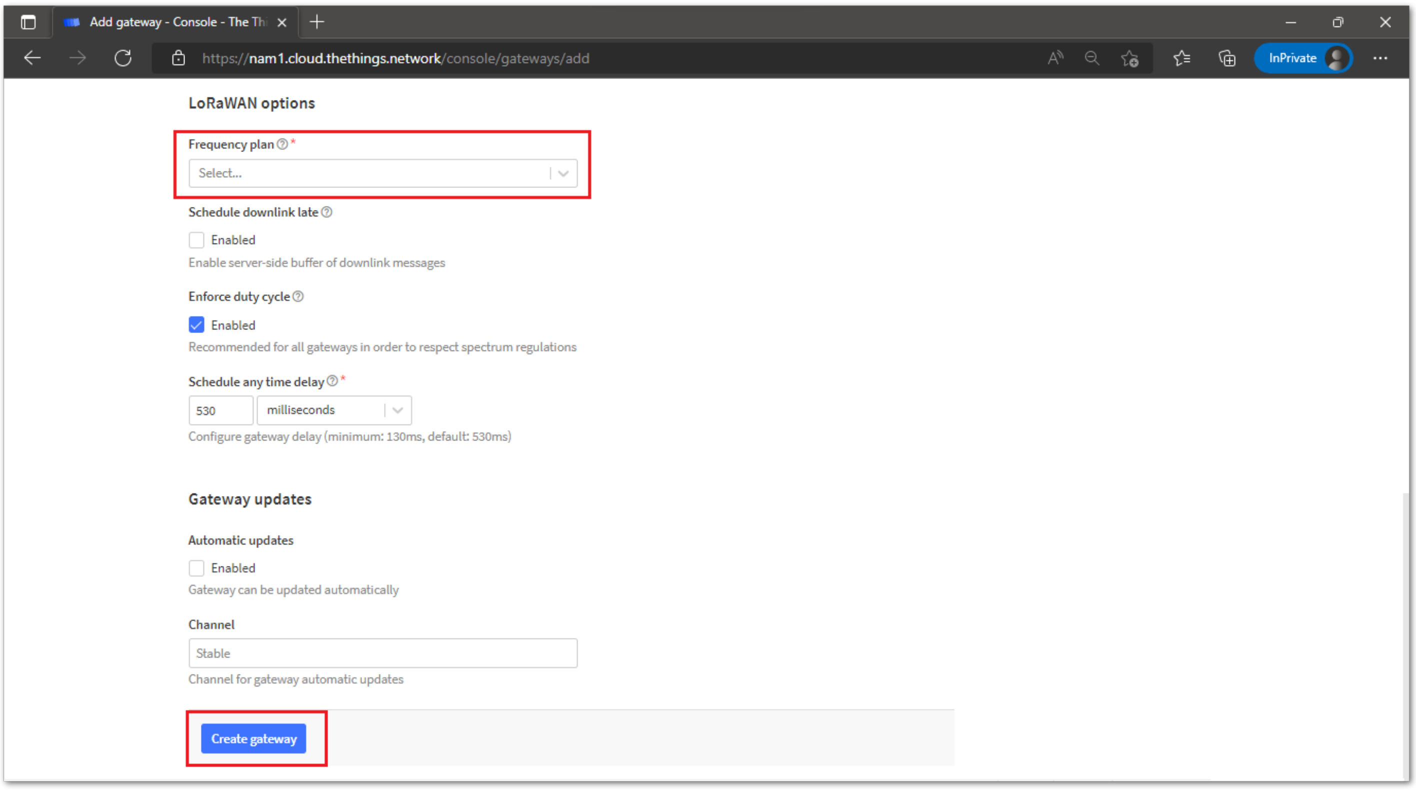
Optional gateway description; can also be used to save notes about the gateway

Gateway Server address

nam1.cloud.thethings.network

Figure 26: TTNv3 gateway registration and configuration

4. Select the **Frequency plan** for your region (with used by TTN), then click on the **Create gateway** button. This will add a new gateway to TTNv3.



Add gateway - Console - The Th
https://nam1.cloud.thethings.network/console/gateways/add
InPrivate

LoRaWAN options

Frequency plan *
Select...

Schedule downlink late
□ Enabled
Enable server-side buffer of downlink messages

Enforce duty cycle
☑ Enabled
Recommended for all gateways in order to respect spectrum regulations

Schedule any time delay *
530 milliseconds
Configure gateway delay (minimum: 130ms, default: 530ms)

Gateway updates

Automatic updates
□ Enabled
Gateway can be updated automatically

Channel
Stable
Channel for gateway automatic updates

Figure 27: TTNv3 add new Gateway

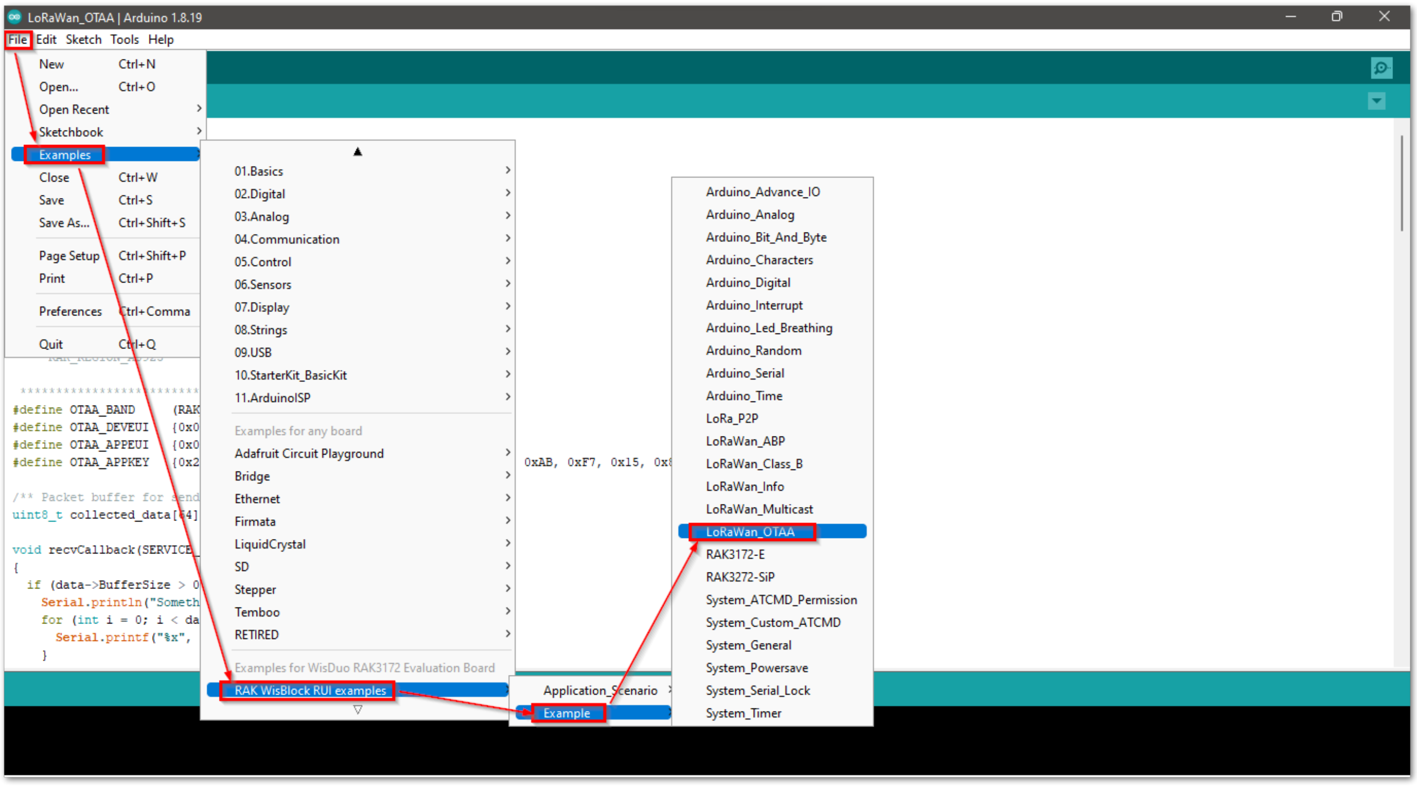
Register the Device on TTNv3

The next step is to follow the procedure described in the section TTN OTAA Device Registration.

Uploading LoRaWAN Example to RAK3172

After a successful registration of the RAK3172 device on the LNS, you can now proceed with running the LoRaWAN OTAA demo application example.

1. Launch Arduino IDE and configure WisDuo RAK3172 Evaluation Board on board selection. See **Figure 9**.
2. Connect the RAK3172 via UART and check RAK3172 COM Port. See **Figure 10**.
3. Open the example code under **RAK WisBlock RUI examples**: **File** -> **Examples** -> **RAK WisBlock RUI examples** -> **Example** -> **LoRaWan_OTAA**.



```
#define OTAA_BAND     (RAK
#define OTAA_DEVEUI   {0x0
#define OTAA_APPEUI   {0x0
#define OTAA_APPKEY   {0x2                    0xAB, 0xF7, 0x15, 0x

/** Packet buffer for send
uint8_t collected_data[64]

void recvCallback(SERVICE_
{
  if (data->BufferSize > 0
    Serial.println("Someth
    for (int i = 0; i < da
      Serial.printf("%x",
    }
```

Figure 28: OTAA LoRaWAN application example

4. In the example code, you need to modify the device EUI **OTAA_DEVEUI**, the application EUI **OTAA_APPEUI**, and the application key **OTAA_APPKEY**.

- The **OTAA_DEVEUI** parameter should match the device EUI registered in your network server. Note that your RAK3172 module has a sticker with a QR code printed on it. You can use the QR code information to configure the **OTAA_DEVEUI** parameter. The **OTAA_DEVEUI** format is MSB first.

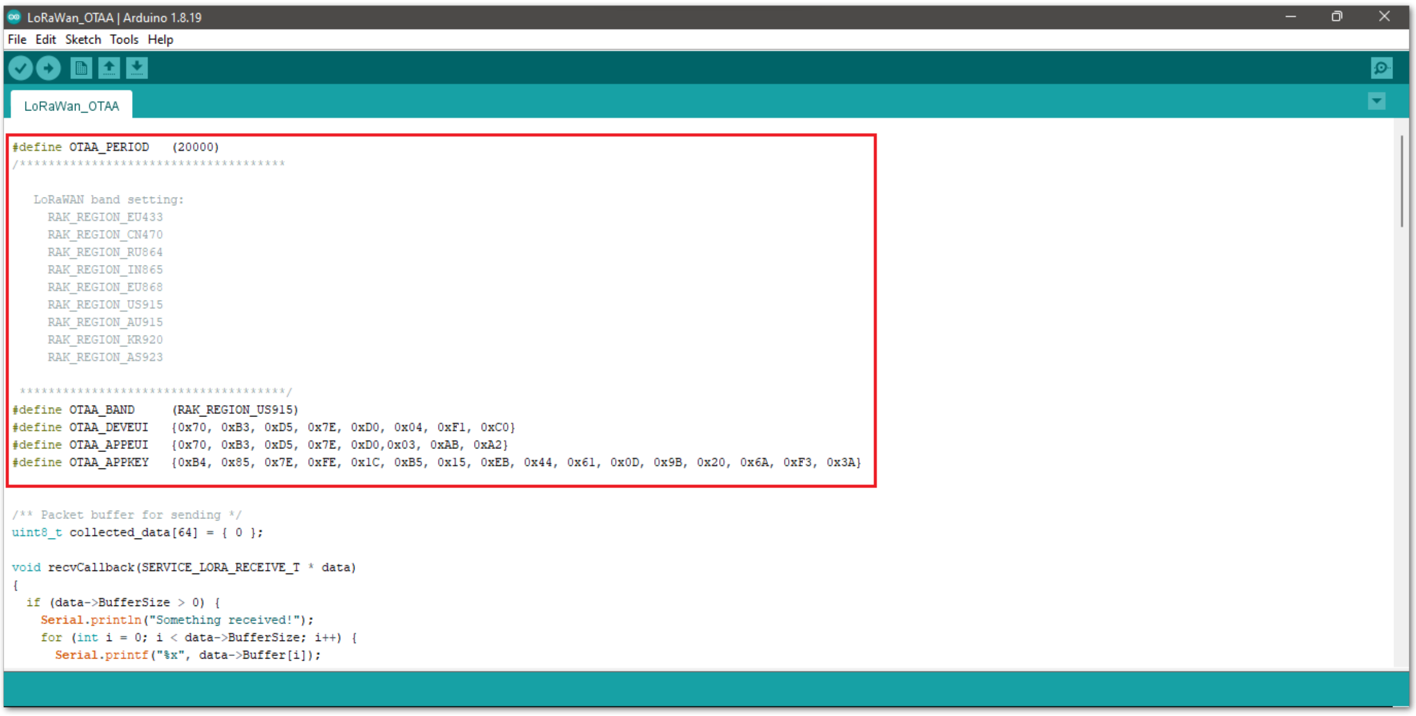
```c
  // OTAA Device EUI MSB
#define OTAA_DEVEUI   {0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x04, 0xF1, 0xC0}
```

- The **OTAA_APPEUI** parameter. This parameter should also be the same as the **APPEUI** in the network server you configured.

```c
  // OTAA Application EUI MSB
#define OTAA_APPEUI   {0x70, 0xB3, 0xD5, 0x7E, 0xD0,0x03, 0xAB, 0xA2}
```

- Another important parameter to change is the application key **OTAA_APPKEY**. This parameter should also be the same as the **APPKEY** in the network server you configured. The **OTAA_APPKEY** format is MSB first.

```c
// OTAA Application Key MSB
#define OTAA_APPKEY   {0xB4, 0x85, 0x7E, 0xFE, 0x1C, 0xB5, 0x15, 0xEB, 0x44, 0x61, 0x0D, 0x9B, 0x
```



```c
#define OTAA_PERIOD   (20000)
/***********************************

   LoRaWAN band setting:
     RAK_REGION_EU433
     RAK_REGION_CN470
     RAK_REGION_RU864
     RAK_REGION_IN865
     RAK_REGION_EU868
     RAK_REGION_US915
     RAK_REGION_AU915
     RAK_REGION_KR920
     RAK_REGION_AS923

 ***********************************/
#define OTAA_BAND     (RAK_REGION_US915)
#define OTAA_DEVEUI   {0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x04, 0xF1, 0xC0}
#define OTAA_APPEUI   {0x70, 0xB3, 0xD5, 0x7E, 0xD0,0x03, 0xAB, 0xA2}
#define OTAA_APPKEY   {0xB4, 0x85, 0x7E, 0xFE, 0x1C, 0xB5, 0x15, 0xEB, 0x44, 0x61, 0x0D, 0x9B, 0x20, 0x6A, 0xF3, 0x3A}

/** Packet buffer for sending */
uint8_t collected_data[64] = { 0 };

void recvCallback(SERVICE_LORA_RECEIVE_T * data)
{
  if (data->BufferSize > 0) {
    Serial.println("Something received!");
    for (int i = 0; i < data->BufferSize; i++) {
      Serial.printf("%x", data->Buffer[i]);
```

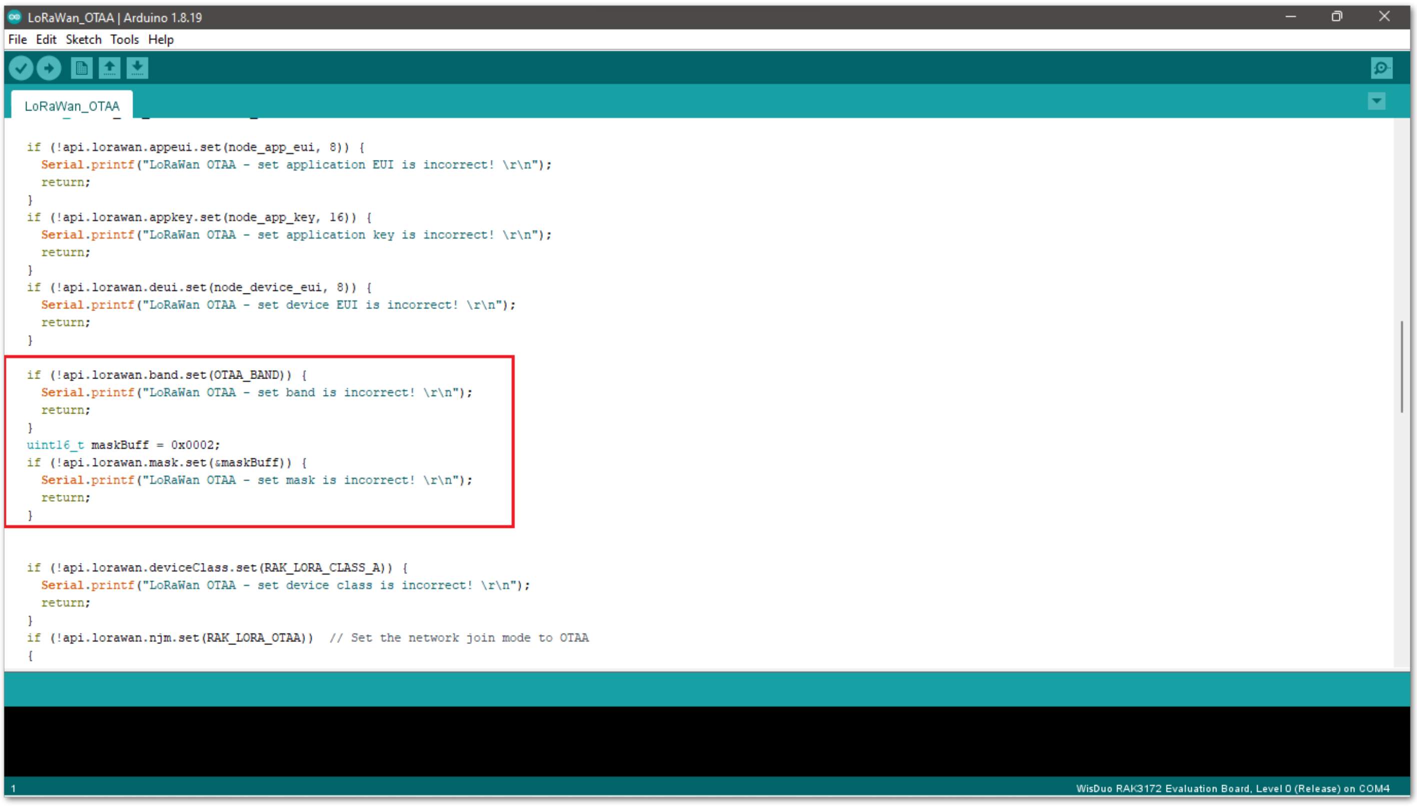
Figure 29: Configuring DEVEUI, APPEUI and APPKEY

3. Depending on the Regional Band you selected, you might need to configure the sub-band of your RAK module to match the gateway and LoRaWAN network server. This is especially important for Regional Bands like US915, AU915, and CN470.

This guide uses US915 regional band, so you need to update the band in the code as well. In addition to that, you also need to set up the channel mask (channels 8 to 15 are the most commonly used channels in the US915 band).

```c
if (!api.lorawan.band.set(OTAA_BAND)) {
  Serial.printf("LoRaWan OTAA - set band is incorrect! \r\n");
  return;
}
uint16_t maskBuff = 0x0002;
if (!api.lorawan.mask.set(&maskBuff)) {
  Serial.printf("LoRaWan OTAA - set mask is incorrect! \r\n");
  return;
}
```

> 📝 **NOTE:**
>
> RAK3172 supports the following regions:
>
> - RAK_REGION_EU433 = 0
> - RAK_REGION_CN470 = 1
> - RAK_REGION_RU864 = 2
> - RAK_REGION_IN865 = 3
> - RAK_REGION_EU868 = 4
> - RAK_REGION_US915 = 5
> - RAK_REGION_AU915 = 6
> - RAK_REGION_KR920 = 7
> - RAK_REGION_AS923 = 8
> - RAK_REGION_AS923-2 = 9
> - RAK_REGION_AS923-3 = 10
> - RAK_REGION_AS923-4 = 11


LoRaWan_OTAA | Arduino 1.8.19
File Edit Sketch Tools Help
LoRaWan_OTAA
1
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM4

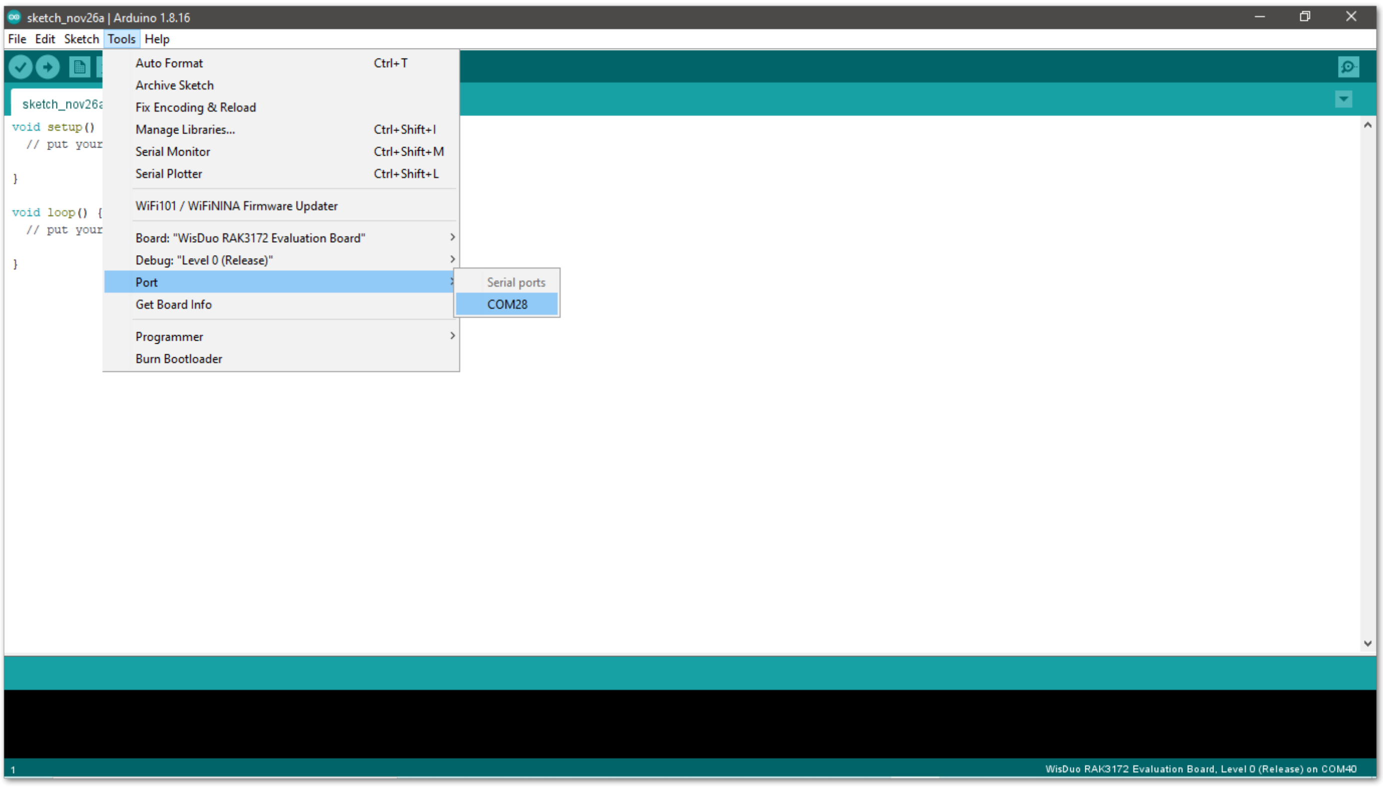
```c
  if (!api.lorawan.appeui.set(node_app_eui, 8)) {
    Serial.printf("LoRaWan OTAA - set application EUI is incorrect! \r\n");
    return;
  }
  if (!api.lorawan.appkey.set(node_app_key, 16)) {
    Serial.printf("LoRaWan OTAA - set application key is incorrect! \r\n");
    return;
  }
  if (!api.lorawan.deui.set(node_device_eui, 8)) {
    Serial.printf("LoRaWan OTAA - set device EUI is incorrect! \r\n");
    return;
  }

  if (!api.lorawan.band.set(OTAA_BAND)) {
    Serial.printf("LoRaWan OTAA - set band is incorrect! \r\n");
    return;
  }
  uint16_t maskBuff = 0x0002;
  if (!api.lorawan.mask.set(&maskBuff)) {
    Serial.printf("LoRaWan OTAA - set mask is incorrect! \r\n");
    return;
  }


  if (!api.lorawan.deviceClass.set(RAK_LORA_CLASS_A)) {
    Serial.printf("LoRaWan OTAA - set device class is incorrect! \r\n");
    return;
  }
  if (!api.lorawan.njm.set(RAK_LORA_OTAA))  // Set the network join mode to OTAA
  {
```

Figure 30: Updating to US915 and setting up channel mask

📝 **NOTE:**

- Make sure you have configured the correct RAK board before uploading the code. See Configure RAK3172 on Boards Manager section.
- Also, check RAK3172 COM Port on Device Manager section.

4. Open the **Tools** Menu and select a COM port. **COM28** is currently used.

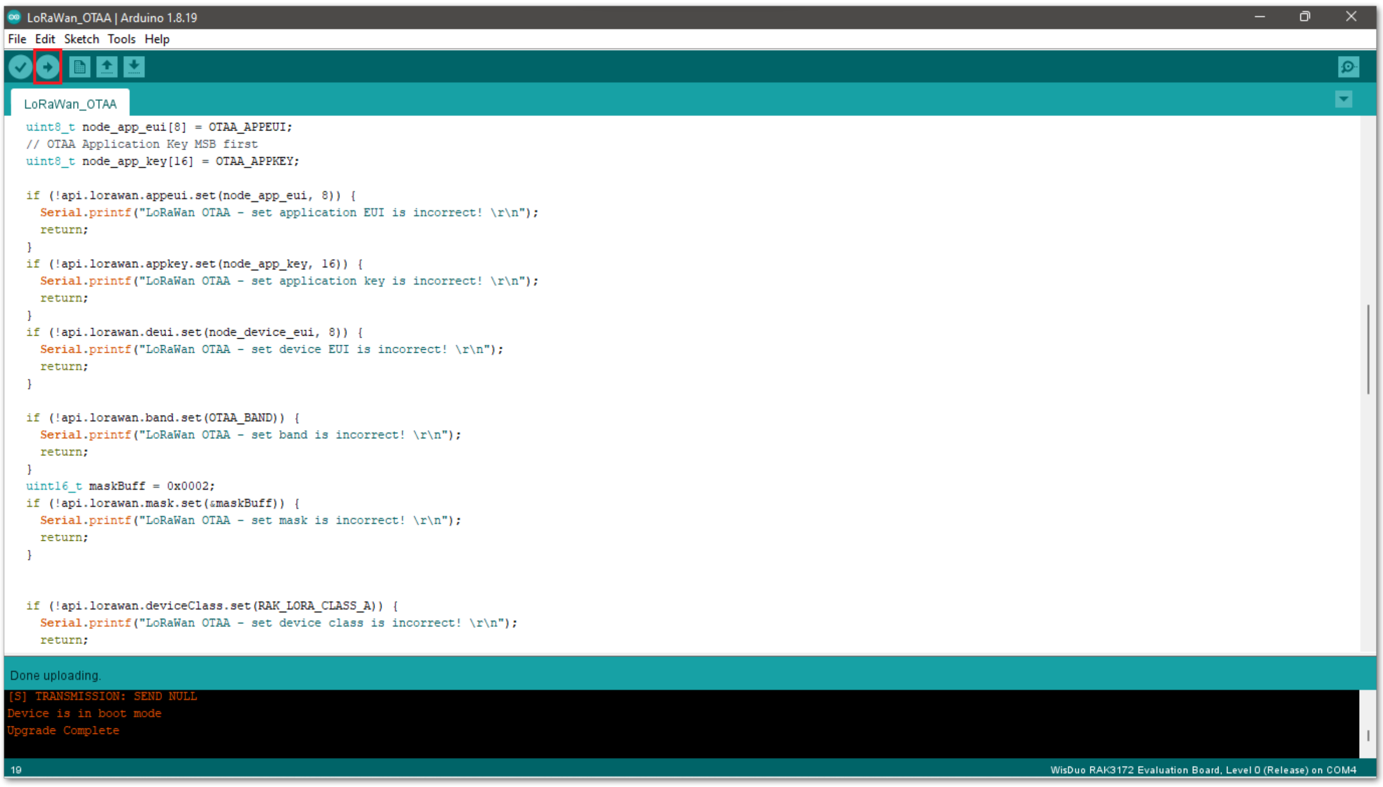


sketch_nov26a | Arduino 1.8.16
File Edit Sketch Tools Help
sketch_nov26a
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "WisDuo RAK3172 Evaluation Board"
Debug: "Level 0 (Release)"
Port
Get Board Info
Programmer
Burn Bootloader
Serial ports
COM28
1
WisDuo RAK3172 Evaluation Board, Level 0 (Release) on COM40

```
void setup()
  // put your

}

void loop() {
  // put your

}
```

Figure 31: Select COM port

5. The last step is to upload the code by clicking the **Upload** icon on Arduino IDE. Take note that you should select the right board and COM port.



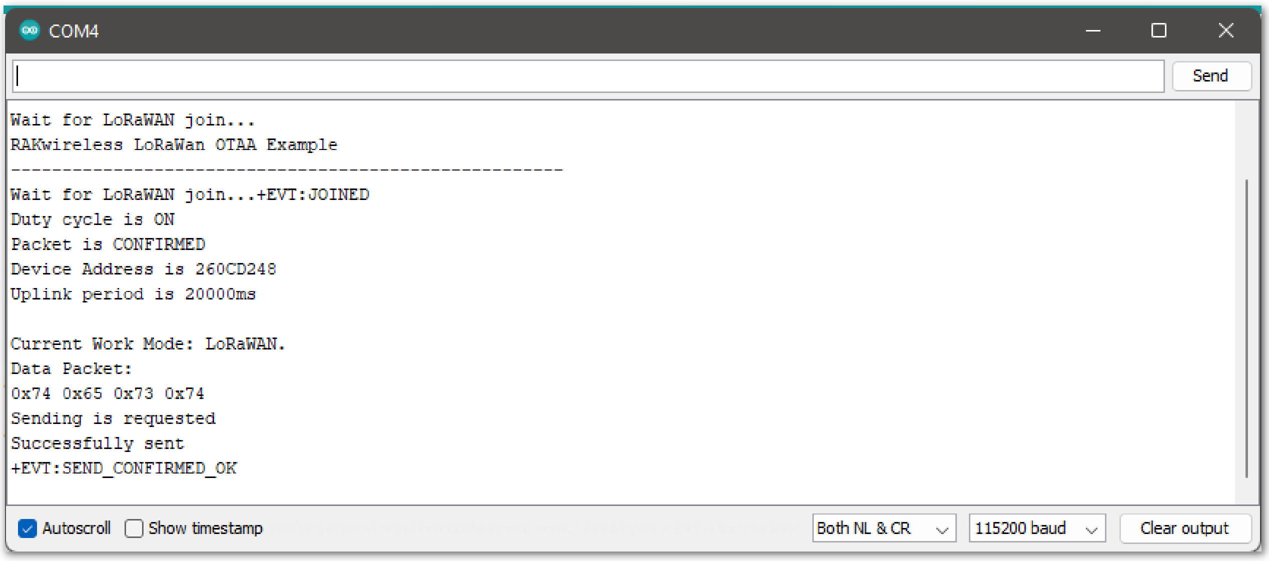
```
  uint8_t node_app_eui[8] = OTAA_APPEUI;
  // OTAA Application Key MSB first
  uint8_t node_app_key[16] = OTAA_APPKEY;

  if (!api.lorawan.appeui.set(node_app_eui, 8)) {
    Serial.printf("LoRaWan OTAA - set application EUI is incorrect! \r\n");
    return;
  }
  if (!api.lorawan.appkey.set(node_app_key, 16)) {
    Serial.printf("LoRaWan OTAA - set application key is incorrect! \r\n");
    return;
  }
  if (!api.lorawan.deui.set(node_device_eui, 8)) {
    Serial.printf("LoRaWan OTAA - set device EUI is incorrect! \r\n");
    return;
  }

  if (!api.lorawan.band.set(OTAA_BAND)) {
    Serial.printf("LoRaWan OTAA - set band is incorrect! \r\n");
    return;
  }
  uint16_t maskBuff = 0x0002;
  if (!api.lorawan.mask.set(&maskBuff)) {
    Serial.printf("LoRaWan OTAA - set mask is incorrect! \r\n");
    return;
  }


  if (!api.lorawan.deviceClass.set(RAK_LORA_CLASS_A)) {
    Serial.printf("LoRaWan OTAA - set device class is incorrect! \r\n");
    return;
```

Done uploading.

```
[S] TRANSMISSION: SEND NULL
Device is in boot mode
Upgrade Complete
```

Figure 32: Uploading the code

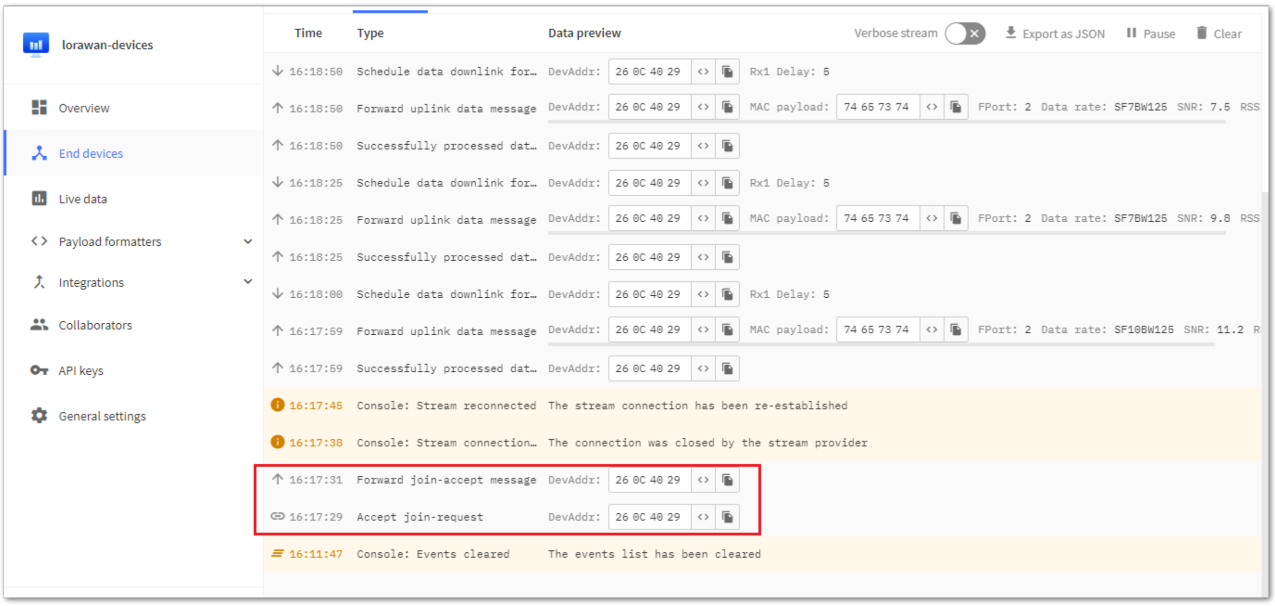
6. You should now be able to see the console logs using the serial monitor of Arduino IDE. Sometimes COM port will be disconnected, so you won't be able to see the terminal output immediately. You can reconnect the module or try to push the reset button to see the terminal output.



```
Wait for LoRaWAN join...
RAKwireless LoRaWan OTAA Example
------------------------------------------------------
Wait for LoRaWAN join...+EVT:JOINED
Duty cycle is ON
Packet is CONFIRMED
Device Address is 260CD248
Uplink period is 20000ms

Current Work Mode: LoRaWAN.
Data Packet:
0x74 0x65 0x73 0x74
Sending is requested
Successfully sent
+EVT:SEND_CONFIRMED_OK
```

Figure 33: Arduino serial monitor logs

7. Check on the LoRaWAN network TTN console logs if your device has been successfully joined with the `join-request` and `join-accept` messages.



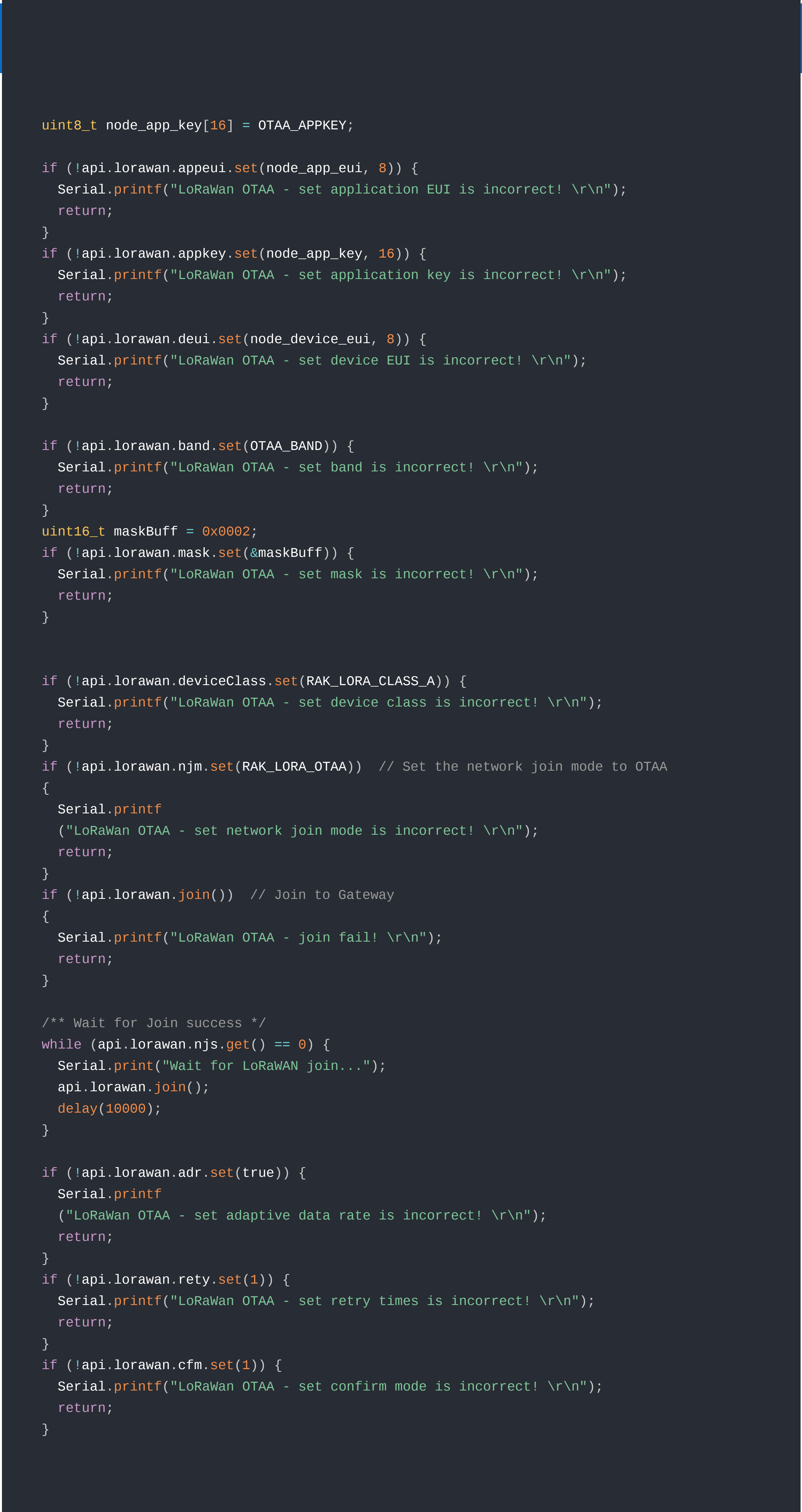
```
Time      Type                         Data preview
16:18:50  Schedule data downlink for…  DevAddr: 26 0C 40 29  Rx1 Delay: 5
16:18:50  Forward uplink data message  DevAddr: 26 0C 40 29  MAC payload: 74 65 73 74  FPort: 2  Data rate: SF7BW125  SNR: 7.5  RSS
16:18:50  Successfully processed dat…  DevAddr: 26 0C 40 29
16:18:25  Schedule data downlink for…  DevAddr: 26 0C 40 29  Rx1 Delay: 5
16:18:25  Forward uplink data message  DevAddr: 26 0C 40 29  MAC payload: 74 65 73 74  FPort: 2  Data rate: SF7BW125  SNR: 9.8  RSS
16:18:25  Successfully processed dat…  DevAddr: 26 0C 40 29
16:18:00  Schedule data downlink for…  DevAddr: 26 0C 40 29  Rx1 Delay: 5
16:17:59  Forward uplink data message  DevAddr: 26 0C 40 29  MAC payload: 74 65 73 74  FPort: 2  Data rate: SF10BW125  SNR: 11.2  R
16:17:59  Successfully processed dat…  DevAddr: 26 0C 40 29
16:17:45  Console: Stream reconnected  The stream connection has been re-established
16:17:38  Console: Stream connection…  The connection was closed by the stream provider
16:17:31  Forward join-accept message  DevAddr: 26 0C 40 29
16:17:29  Accept join-request          DevAddr: 26 0C 40 29
16:11:47  Console: Events cleared      The events list has been cleared
```

Figure 34: TTN console log

The modified `LoRaWAN_OTAA` project is available below.

C

```
#define OTAA_PERIOD   (20000)
/*************************************

   LoRaWAN band setting:
     RAK_REGION_EU433
     RAK_REGION_CN470
     RAK_REGION_RU864
     RAK_REGION_IN865
     RAK_REGION_EU868
     RAK_REGION_US915
     RAK_REGION_AU915
     RAK_REGION_KR920
     RAK_REGION_AS923

 *************************************/
#define OTAA_BAND     (RAK_REGION_US915)
#define OTAA_DEVEUI   {0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x04, 0xF1, 0xC0}
#define OTAA_APPEUI   {0x70, 0xB3, 0xD5, 0x7E, 0xD0,0x03, 0xAB, 0xA2}
#define OTAA_APPKEY   {0xB4, 0x85, 0x7E, 0xFE, 0x1C, 0xB5, 0x15, 0xEB, 0x44, 0x61, 0x0D, 0x9B, 0x


/** Packet buffer for sending */
uint8_t collected_data[64] = { 0 };

void recvCallback(SERVICE_LORA_RECEIVE_T * data)
{
  if (data->BufferSize > 0) {
    Serial.println("Something received!");
    for (int i = 0; i < data->BufferSize; i++) {
      Serial.printf("%x", data->Buffer[i]);
    }
    Serial.print("\r\n");
  }
}

void joinCallback(int32_t status)
{
  Serial.printf("Join status: %d\r\n", status);
}

void sendCallback(int32_t status)
{
  if (status == 0) {
    Serial.println("Successfully sent");
  } else {
    Serial.println("Sending failed");
  }
}

void setup()
{
  Serial.begin(115200, RAK_AT_MODE);

  Serial.println("RAKwireless LoRaWan OTAA Example");
  Serial.println("------------------------------------------------------");

  // OTAA Device EUI MSB first
  uint8_t node_device_eui[8] = OTAA_DEVEUI;
  // OTAA Application EUI MSB first
  uint8_t node_app_eui[8] = OTAA_APPEUI;
  // OTAA Application Key MSB first
```



```
uint8_t node_app_key[16] = OTAA_APPKEY;

if (!api.lorawan.appeui.set(node_app_eui, 8)) {
  Serial.printf("LoRaWan OTAA - set application EUI is incorrect! \r\n");
  return;
}
if (!api.lorawan.appkey.set(node_app_key, 16)) {
  Serial.printf("LoRaWan OTAA - set application key is incorrect! \r\n");
  return;
}
if (!api.lorawan.deui.set(node_device_eui, 8)) {
  Serial.printf("LoRaWan OTAA - set device EUI is incorrect! \r\n");
  return;
}

if (!api.lorawan.band.set(OTAA_BAND)) {
  Serial.printf("LoRaWan OTAA - set band is incorrect! \r\n");
  return;
}
uint16_t maskBuff = 0x0002;
if (!api.lorawan.mask.set(&maskBuff)) {
  Serial.printf("LoRaWan OTAA - set mask is incorrect! \r\n");
  return;
}


if (!api.lorawan.deviceClass.set(RAK_LORA_CLASS_A)) {
  Serial.printf("LoRaWan OTAA - set device class is incorrect! \r\n");
  return;
}
if (!api.lorawan.njm.set(RAK_LORA_OTAA))  // Set the network join mode to OTAA
{
  Serial.printf
  ("LoRaWan OTAA - set network join mode is incorrect! \r\n");
  return;
}
if (!api.lorawan.join())  // Join to Gateway
{
  Serial.printf("LoRaWan OTAA - join fail! \r\n");
  return;
}

/** Wait for Join success */
while (api.lorawan.njs.get() == 0) {
  Serial.print("Wait for LoRaWAN join...");
  api.lorawan.join();
  delay(10000);
}

if (!api.lorawan.adr.set(true)) {
  Serial.printf
  ("LoRaWan OTAA - set adaptive data rate is incorrect! \r\n");
  return;
}
if (!api.lorawan.rety.set(1)) {
  Serial.printf("LoRaWan OTAA - set retry times is incorrect! \r\n");
  return;
}
if (!api.lorawan.cfm.set(1)) {
  Serial.printf("LoRaWan OTAA - set confirm mode is incorrect! \r\n");
  return;
}
```

```
  /** Check LoRaWan Status*/
  Serial.printf("Duty cycle is %s\r\n", api.lorawan.dcs.get()? "ON" : "OFF");   // Check Duty Cyc
  Serial.printf("Packet is %s\r\n", api.lorawan.cfm.get()? "CONFIRMED" : "UNCONFIRMED");    // Ch
  uint8_t assigned_dev_addr[4] = { 0 };
  api.lorawan.daddr.get(assigned_dev_addr, 4);
  Serial.printf("Device Address is %02X%02X%02X%02X\r\n", assigned_dev_addr[0], assigned_dev_addr
  Serial.printf("Uplink period is %ums\r\n", OTAA_PERIOD);
  Serial.println("");
  api.lorawan.registerRecvCallback(recvCallback);
  api.lorawan.registerJoinCallback(joinCallback);
  api.lorawan.registerSendCallback(sendCallback);
}

void uplink_routine()
{
  /** Payload of Uplink */
  uint8_t data_len = 0;
  collected_data[data_len++] = (uint8_t) 't';
  collected_data[data_len++] = (uint8_t) 'e';
  collected_data[data_len++] = (uint8_t) 's';
  collected_data[data_len++] = (uint8_t) 't';

  Serial.println("Data Packet:");
  for (int i = 0; i < data_len; i++) {
    Serial.printf("0x%02X ", collected_data[i]);
  }
  Serial.println("");

  /** Send the data package */
  if (api.lorawan.send(data_len, (uint8_t *) & collected_data, 2, true, 1)) {
    Serial.println("Sending is requested");
  } else {
    Serial.println("Sending failed");
  }
}

void loop()
{
  static uint64_t last = 0;
  static uint64_t elapsed;

  if ((elapsed = millis() - last) > OTAA_PERIOD) {
    uplink_routine();

    last = millis();
  }
  //Serial.printf("Try sleep %ums..", OTAA_PERIOD);
  api.system.sleep.all(OTAA_PERIOD);
  //Serial.println("Wakeup..");
}
```

RAK3172 as a LoRa/LoRaWAN Modem via AT Command

AT Command via UART2

RAK3172 module can be configured using AT commands via the UART2 interface. You need a USB to UART TTL adapter to connect the RAK3172 to your computer's USB port and a serial terminal tool. You can use the RAK

Serial Port Tool so you can easily send AT commands and view the replies from the console output. The RAK Serial Port Tool commands still uses the RUI V2 AT commands by default. You can modify it to have RUI3 AT commands and then save it.

> ⚠ **WARNING**
>
> Firmware update and AT command functionality are done via UART2 pins. If you will connect the module to an external host MCU that will send AT commands via UART2, take extra precautions in your board design to ensure you can still perform FW update to it. There should be a way in your board design that can disconnect the host MCU UART to connect to RAK3172 UART2 before connecting the module to the PC (via USB-UART converter) for the FW update process.
>
> An alternative option to update firmware aside from UART2 is to use SWD pins (SWCLK & SWDIO). This method will require you to use external tools like ST-LINK and RAKDAP1.

Connect to the RAK3172

1. Connect the RAK3172 to the serial port of a general-purpose computer (USB port) using a USB to UART TTL adapter like RAKDAP1 , as shown in **Figure 35**.

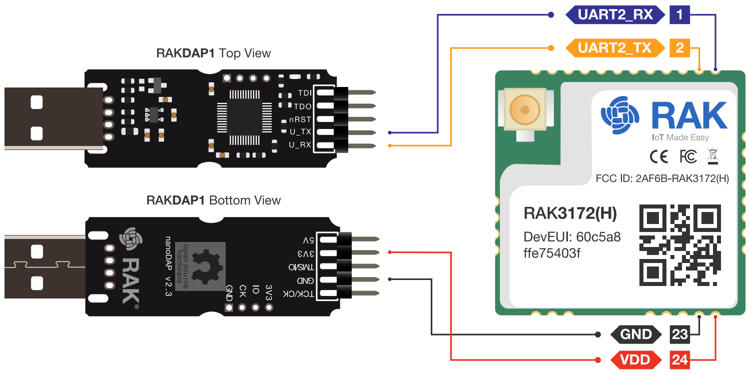


Figure 35: RAK3172 Module Connection

2. Any serial communication tool can be used; but, it is recommended to use the RAK Serial Port Tool .

3. Configure the serial communication tool by selecting the proper port detected by the computer and configure the link as follows:

- Baud Rate: **115200 baud**
- Data Bits: **8 bits**
- Stop Bits: **1 stop bit**
- Parity: **NONE**

RAK3172 Configuration for LoRaWAN or LoRa P2P

To enable the RAK3172 module as a LoRa P2P module or a LoRaWAN end-device, the module must be configured and parameters must be set by sending AT commands. You can configure the RAK3172 in two ways:

- LoRaWAN End-Device - RAK3172 as LoRaWAN IoT device.
- LoRa P2P - Point-to-point communication between two RAK3172 modules.

Configuring RAK3172 as LoRaWAN End-Device

To enable the RAK3172 module as a LoRaWAN end-device, a device must be registered first in the LoRaWAN network server. This guide will cover both TTN and Chirpstack LoRaWAN network servers and the associated AT commands for the RAK3172.

This guide covers the following topics:

- TheThingsNetwork Guide - How to log in, register new accounts and create new applications on TTN.
- RAK3172 TTN OTAA Guide - How to add OTAA device on TTN and what AT commands to use on RAK3172 OTAA activation.
- RAK3172 TTN ABP Guide - How to add ABP device on TTN and what AT commands to use on RAK3172 ABP activation.
- Chirpstack Guide - How to create new applications on Chirpstack.
- RAK3172 Chirpstack OTAA Guide - How to add OTAA device to Chirpstack and what AT commands to use on RAK3172 OTAA activation.
- RAK3172 Chirpstack ABP Guide - How to add ABP device on Chirpstack and what AT commands to use on RAK3172 ABP activation.

Connecting to The Things Network (TTN)

In this section, a quick tutorial guide will show how to connect the RAK3172 module to the TTN platform.

> 📝 **NOTE:**
>
> In this guide, you need to have a working gateway that is connected to TTN or you have to be within the coverage of a TTN community network.

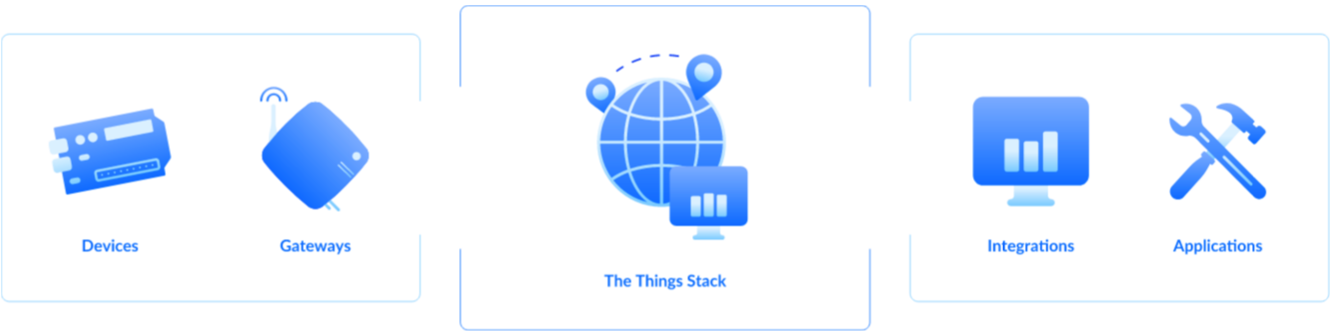


Figure 36: RAK3172 EVB in the context of the TTN

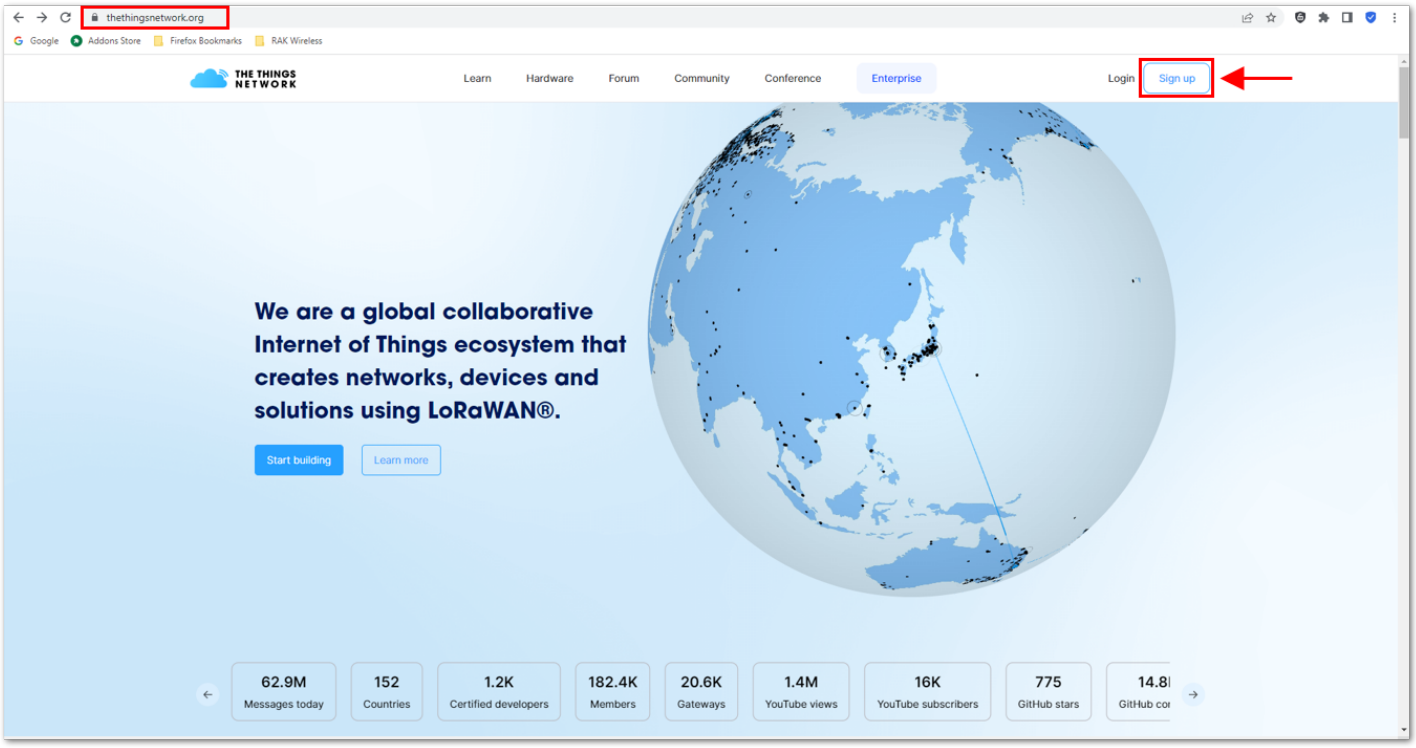
As shown in **Figure 36**, The Things Stack (TTN V3) is an open-source LoRaWAN Network Server suitable for global, geo-distributed public and private deployments, as well as for small local networks. The architecture follows the LoRaWAN Network Reference Model for standards compliance and interoperability. This project is actively maintained by The Things Industries .

LoRaWAN is a protocol for low-power wide-area networks. It allows for large-scale Internet of Things deployments where low-powered devices efficiently communicate with Internet-connected applications over long-range wireless connections.

The RAK3172 WisDuo module can be part of this ecosystem as a device, and the objective of this section is to demonstrate how simple it is to send data to The Things Stack using the LoRaWAN protocol. To achieve this, the RAK3172 WisDuo module must be located inside the coverage of a LoRaWAN gateway connected to The Things Stack server.

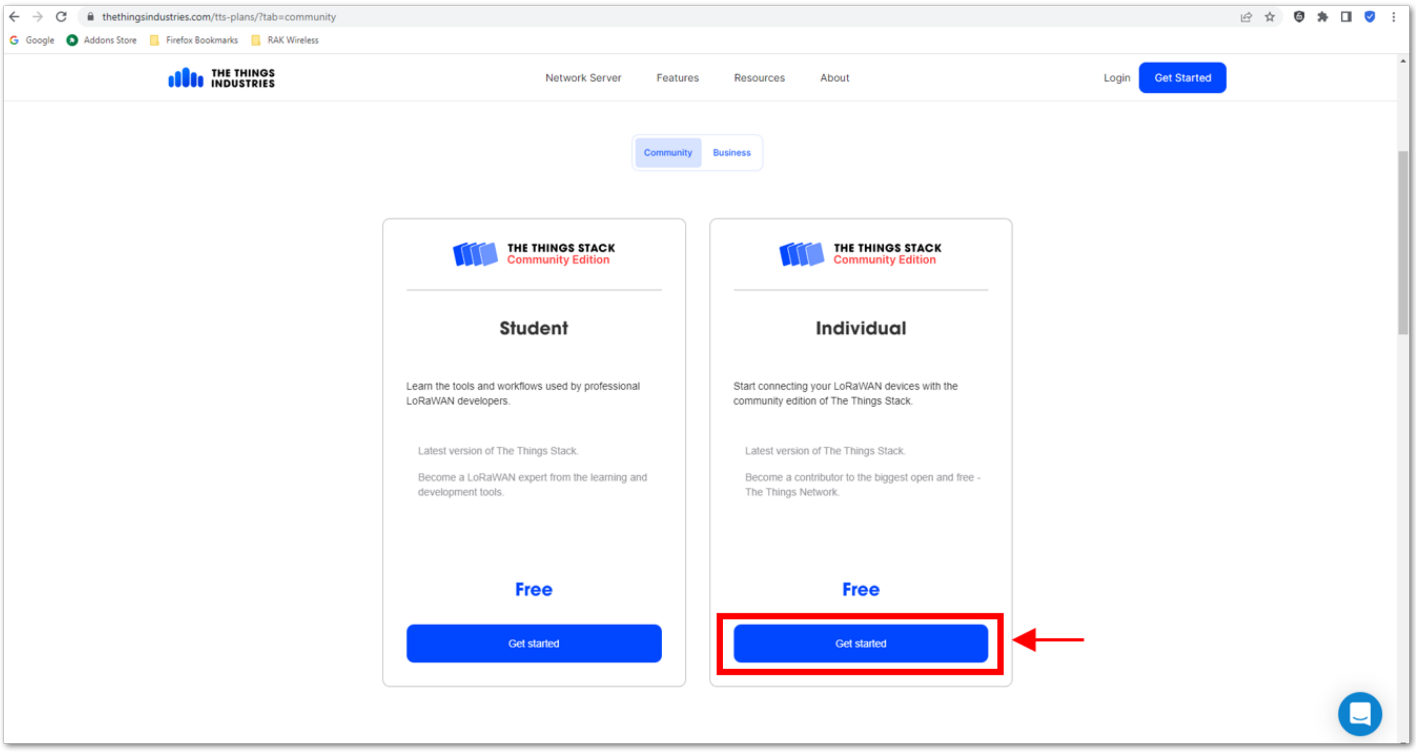
Registration to TTN and Creating LoRaWAN Applications

1. The first step is to go to The Things Network and sign up an account shown in **Figure 37**. Then select a cluster as shown in **Figure 39**.



We are a global collaborative Internet of Things ecosystem that creates networks, devices and solutions using LoRaWAN®.

Figure 37: Signing up an account in TTN



Student

Learn the tools and workflows used by professional LoRaWAN developers.

Latest version of The Things Stack.

Become a LoRaWAN expert from the learning and development tools.

Free

Get started

Individual

Start connecting your LoRaWAN devices with the community edition of The Things Stack.

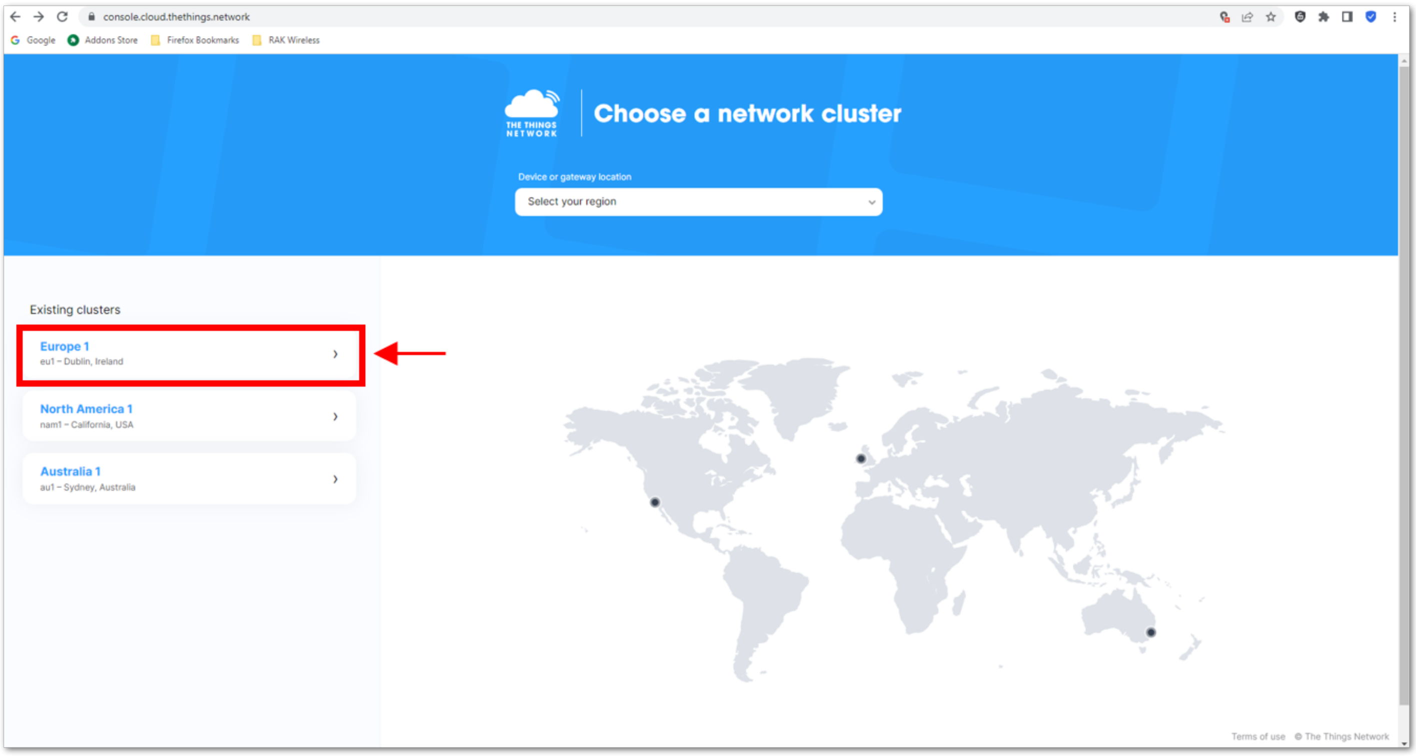
Latest version of The Things Stack.

Become a contributor to the biggest open and free - The Things Network.

Free

Get started

Figure 38: Signing up an account in TTN



Choose a network cluster

Device or gateway location

Select your region

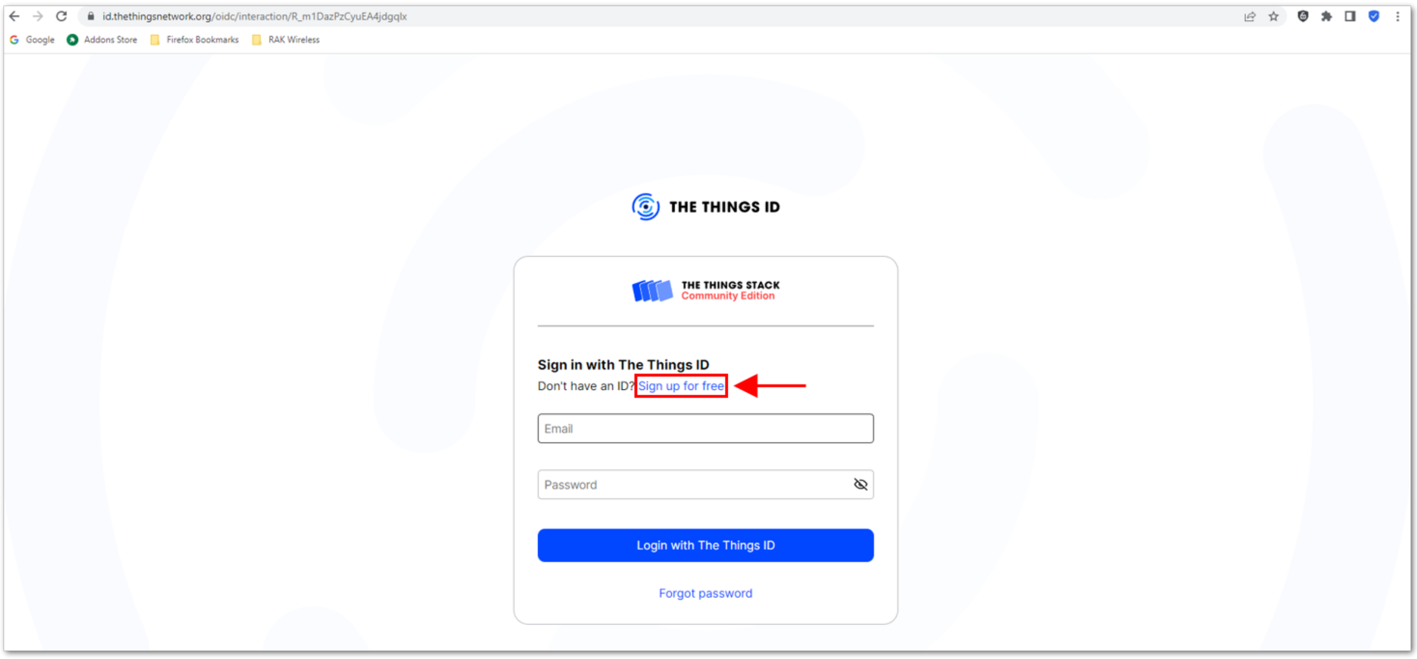
Existing clusters

Europe 1
eu1 – Dublin, Ireland

North America 1
nam1 – California, USA

Australia 1
au1 – Sydney, Australia

Figure 39: Selecting Cluster in TTN



Sign in with The Things ID

Don't have an ID? Sign up for free

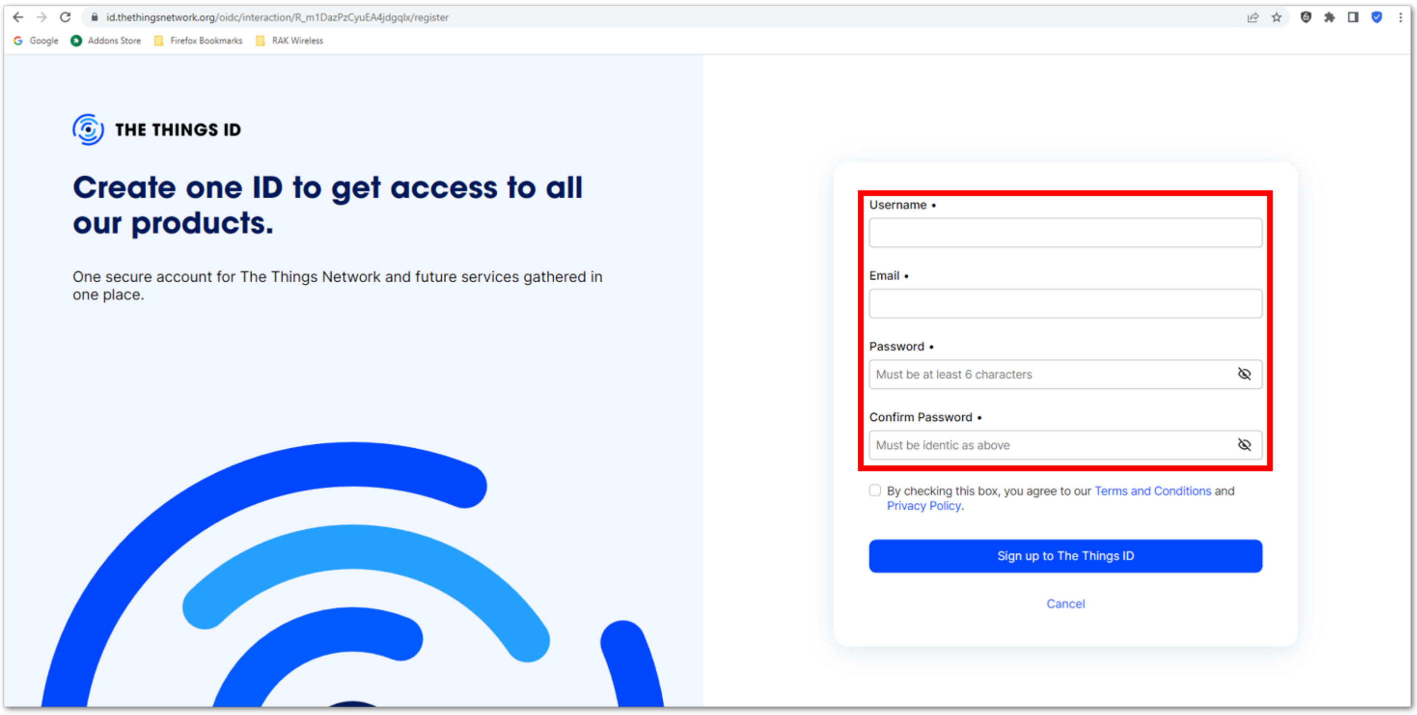
Email

Password

Login with The Things ID

Forgot password

Figure 40: Signing up through the Things ID


id.thethingsnetwork.org/oidc/interaction/R_m1DazPzCyuEA4jdgqlx/register
Google
Addons Store
Firefox Bookmarks
RAK Wireless
THE THINGS ID

Create one ID to get access to all our products.

One secure account for The Things Network and future services gathered in one place.

Username •

Email •

Password •
Must be at least 6 characters

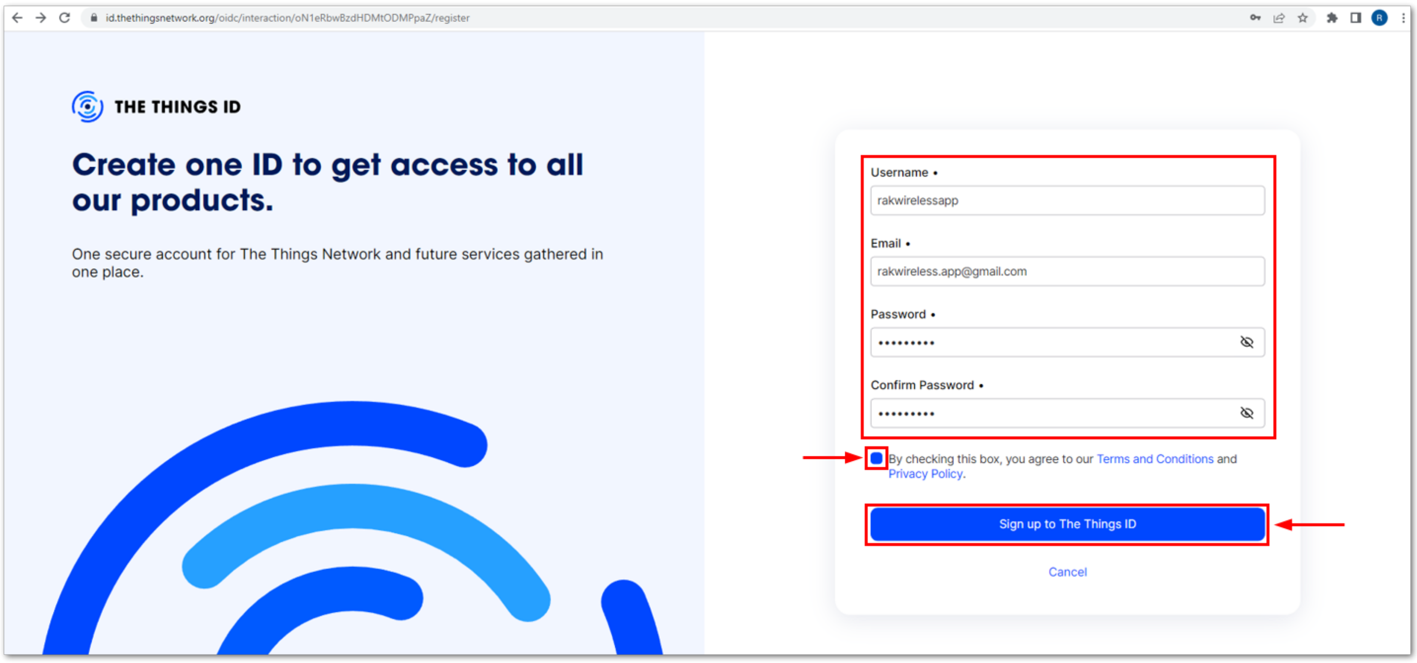
Confirm Password •
Must be identic as above

□ By checking this box, you agree to our Terms and Conditions and Privacy Policy.

Sign up to The Things ID

Cancel

Figure 41: Creation of an account through the Things ID



id.thethingsnetwork.org/oidc/interaction/oN1eRbwBzdHDMtODMPpaZ/register

THE THINGS ID

Create one ID to get access to all our products.

One secure account for The Things Network and future services gathered in one place.

Username •
rakwirelessapp

Email •
rakwireless.app@gmail.com

Password •
•••••••••

Confirm Password •
•••••••••

☑ By checking this box, you agree to our Terms and Conditions and Privacy Policy.

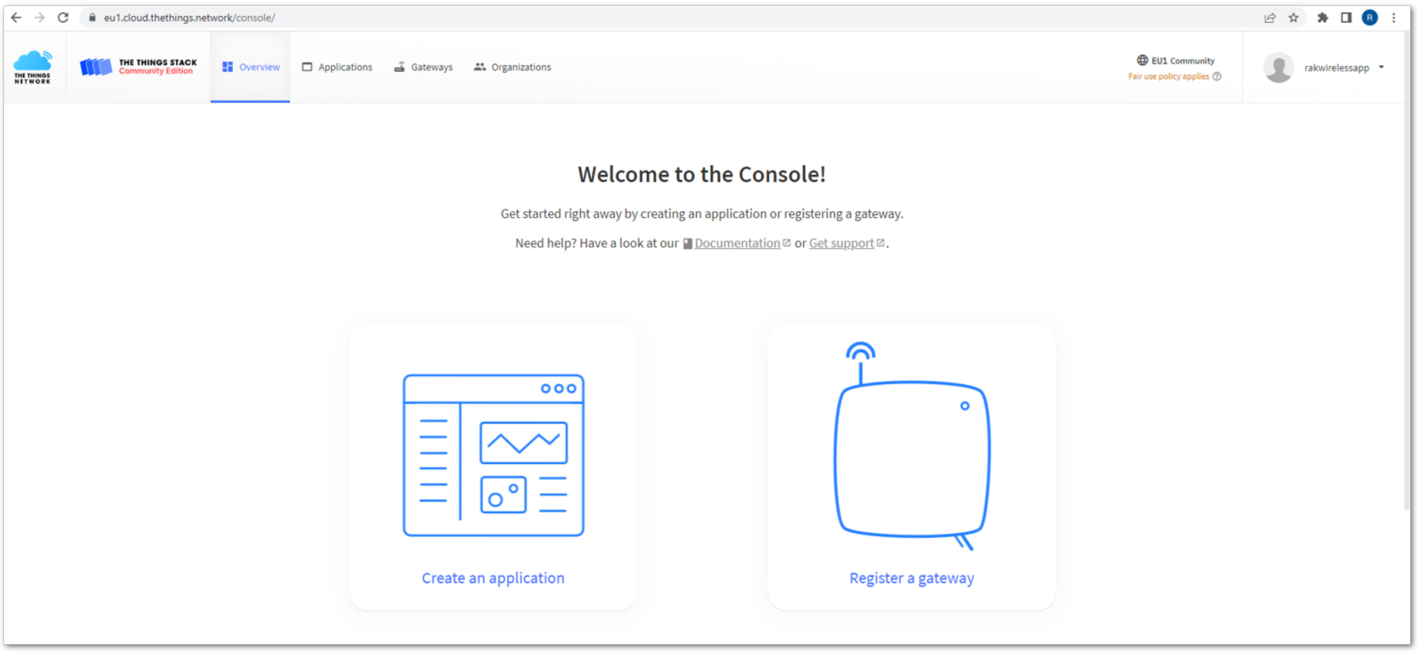
Sign up to The Things ID

Cancel

Figure 42: Creation of an account through the Things ID

You can use the same login credentials on the TTN V2 if you have one. If you have no account yet, you need to create one.

2. Now that you are logged in to the platform, the next step is to create an application. Click **Create an application**.



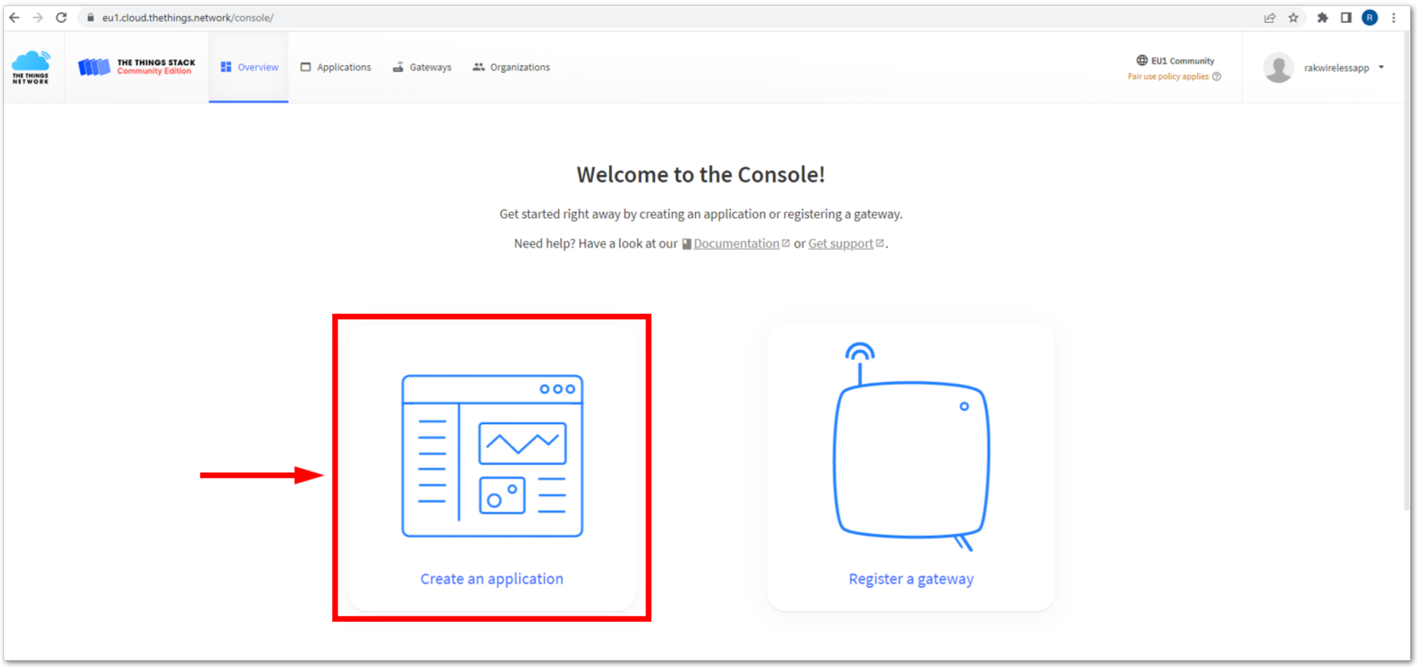
eu1.cloud.thethings.network/console/

Welcome to the Console!

Get started right away by creating an application or registering a gateway.

Need help? Have a look at our Documentation or Get support.

Figure 43: The Things Stack Platform



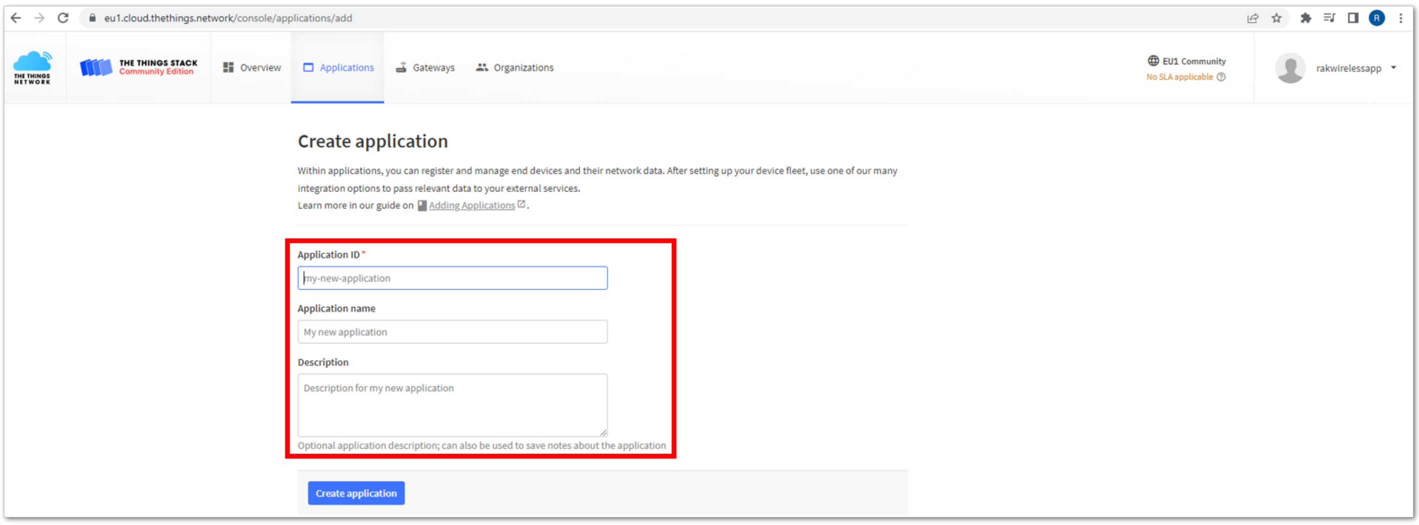
Welcome to the Console!

Get started right away by creating an application or registering a gateway.

Need help? Have a look at our Documentation or Get support.

Figure 44: Creating TTN application for your LoRaWAN devices

3. To have an application registered, input first the specific details and necessary information about your application then click **Create application**.



Create application

Within applications, you can register and manage end devices and their network data. After setting up your device fleet, use one of our many integration options to pass relevant data to your external services.
Learn more in our guide on Adding Applications.

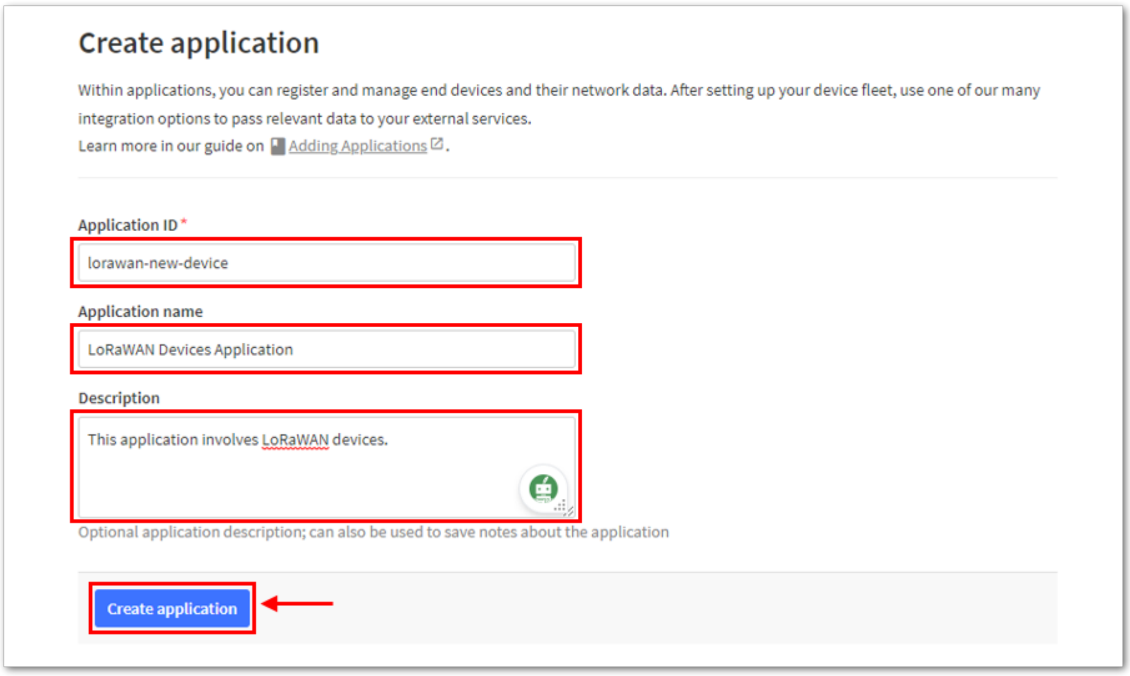
Application ID *
my-new-application

Application name
My new application

Description
Description for my new application

Optional application description; can also be used to save notes about the application

Figure 45: Details of the TTN application



Create application

Within applications, you can register and manage end devices and their network data. After setting up your device fleet, use one of our many integration options to pass relevant data to your external services.
Learn more in our guide on Adding Applications.

Application ID *
lorawan-new-device

Application name
LoRaWAN Devices Application

Description
This application involves LoRaWAN devices.

Optional application description; can also be used to save notes about the application

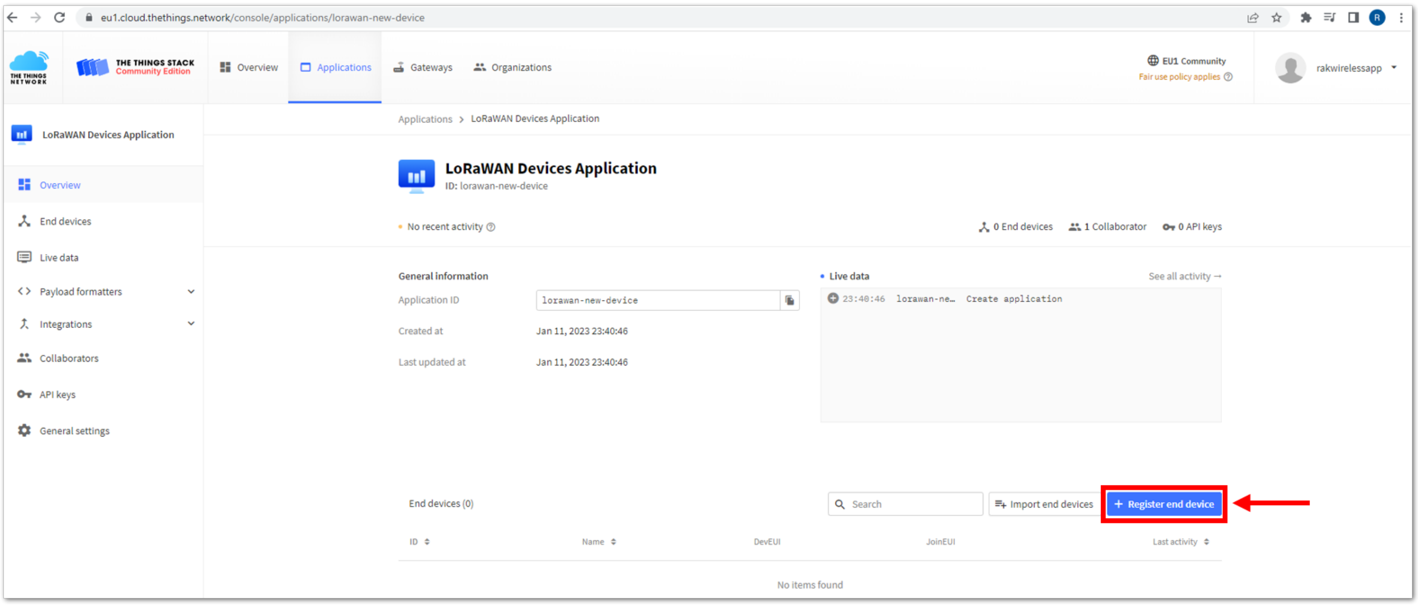
Figure 46: Details of the TTN application

4. If you have no error on the previous step, you should now be on the application console page. The next step is to add end-devices to your TTN application.

LoRaWAN specifications enforce that each end-device has to be personalized and activated. There are two options in registering devices depending on the activation mode selected. Activation can be done either via Over-The-Air-Activation (OTAA) or Activation-By-Personalization (ABP).

TTN OTAA Device Registration

1. Go to your application console to register a device. To start adding an OTAA end-device, click **+ Register end device**, as shown in **Figure 47**.



LoRaWAN Devices Application
ID: lorawan-new-device

General information

Application ID lorawan-new-device

Created at Jan 11, 2023 23:40:46

Last updated at Jan 11, 2023 23:40:46

Live data

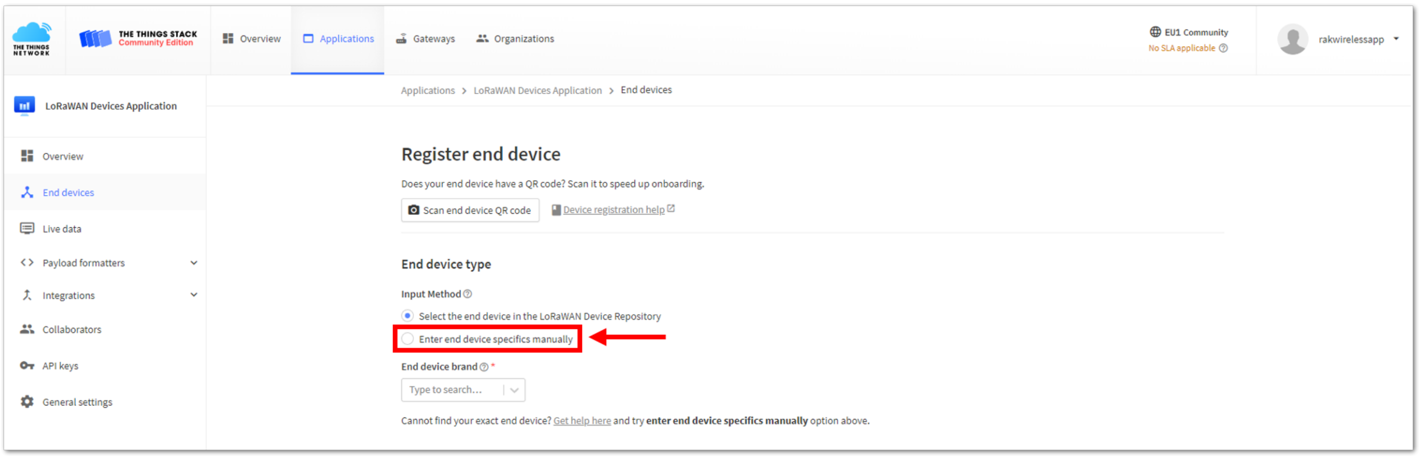
23:40:46 lorawan-ne… Create application

End devices (0)

No items found

Figure 47: Register end device

2. To register the board, click the **Enter end device specifics manually**.



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

End device type

Input Method

● Select the end device in the LoRaWAN Device Repository

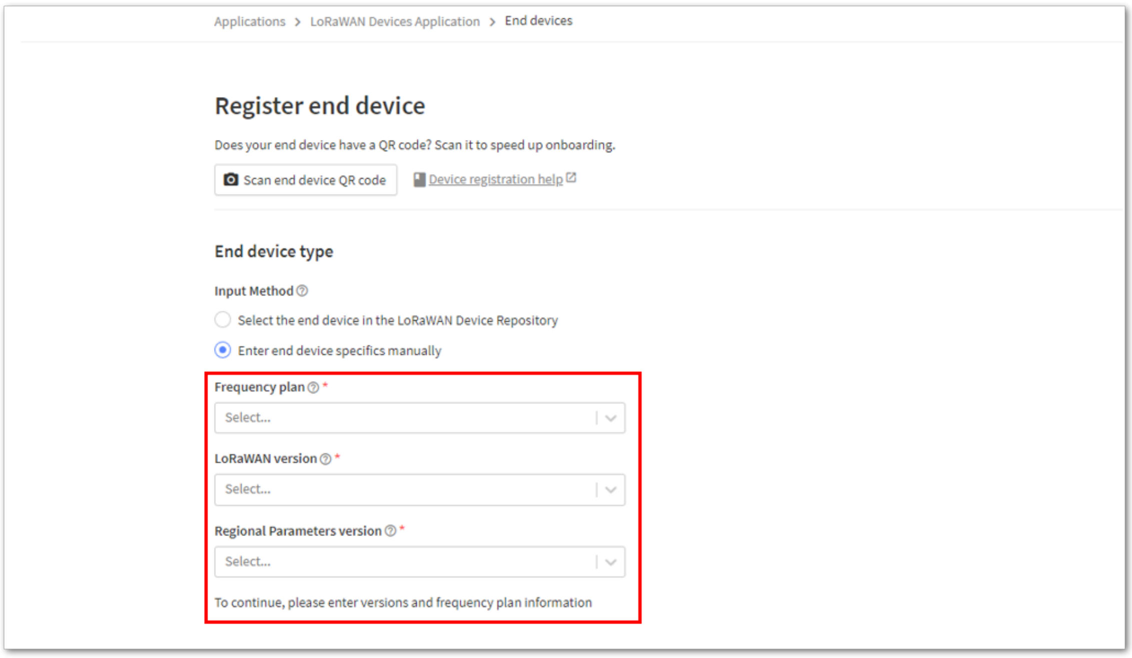
○ Enter end device specifics manually

End device brand *

Cannot find your exact end device? Get help here and try **enter end device specifics manually** option above.

Figure 48: Enter end device specifics manually

3. Next step is to set up **Frequency plan**, compatible **LoRaWAN version**, and **Regional Parameters version** supported. Then provide the **JoinEUI** credentials by entering zeroes into it.



Applications > LoRaWAN Devices Application > End devices

Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

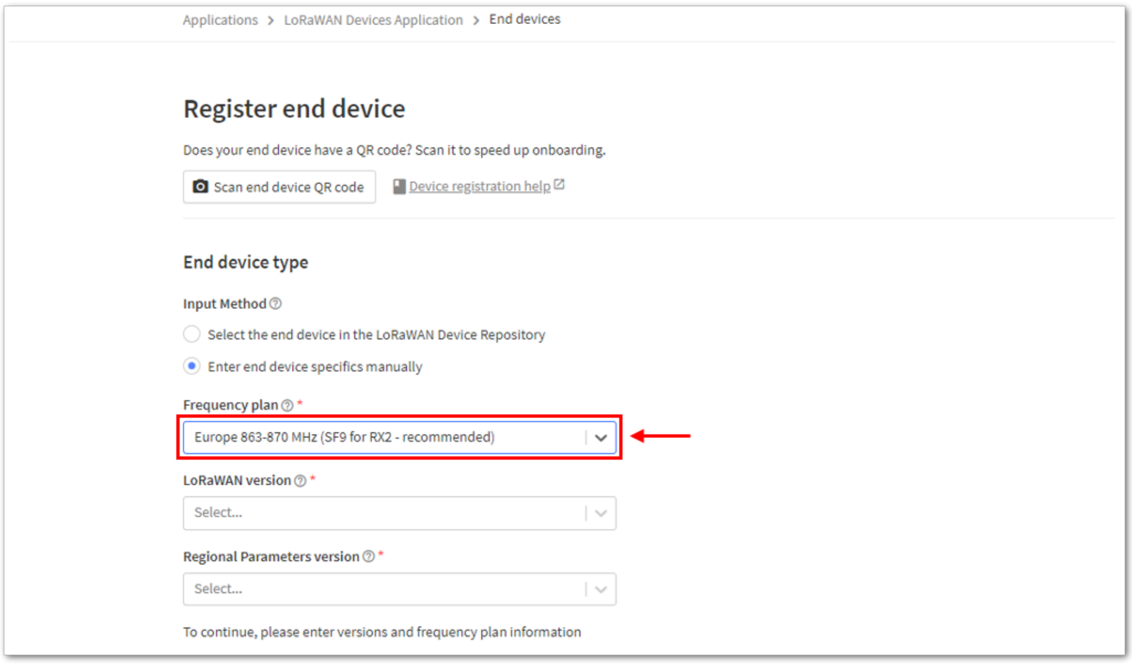
Frequency plan *: Select...

LoRaWAN version *: Select...

Regional Parameters version *: Select...

To continue, please enter versions and frequency plan information

Figure 49: Setting up for your device



Applications > LoRaWAN Devices Application > End devices

Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

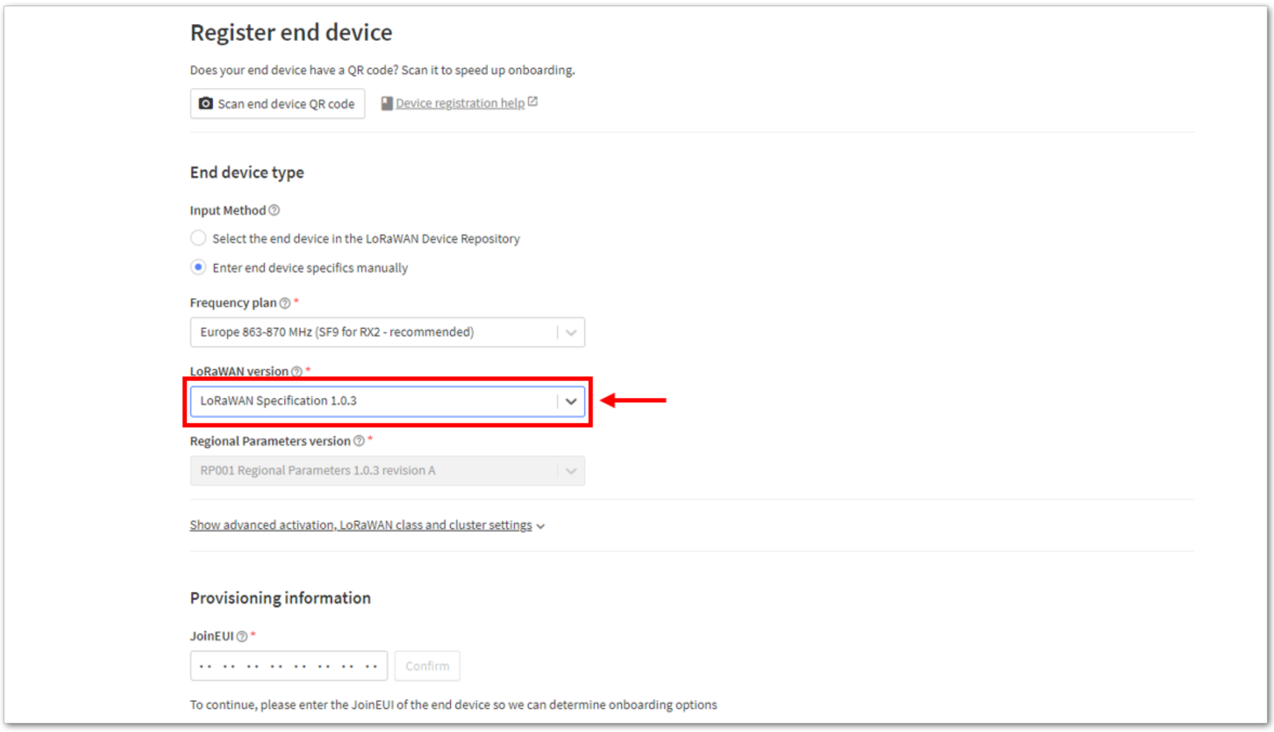
Frequency plan *: Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *: Select...

Regional Parameters version *: Select...

To continue, please enter versions and frequency plan information

Figure 50: Setting up for your device



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *: Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *: LoRaWAN Specification 1.0.3

Regional Parameters version *: RP001 Regional Parameters 1.0.3 revision A

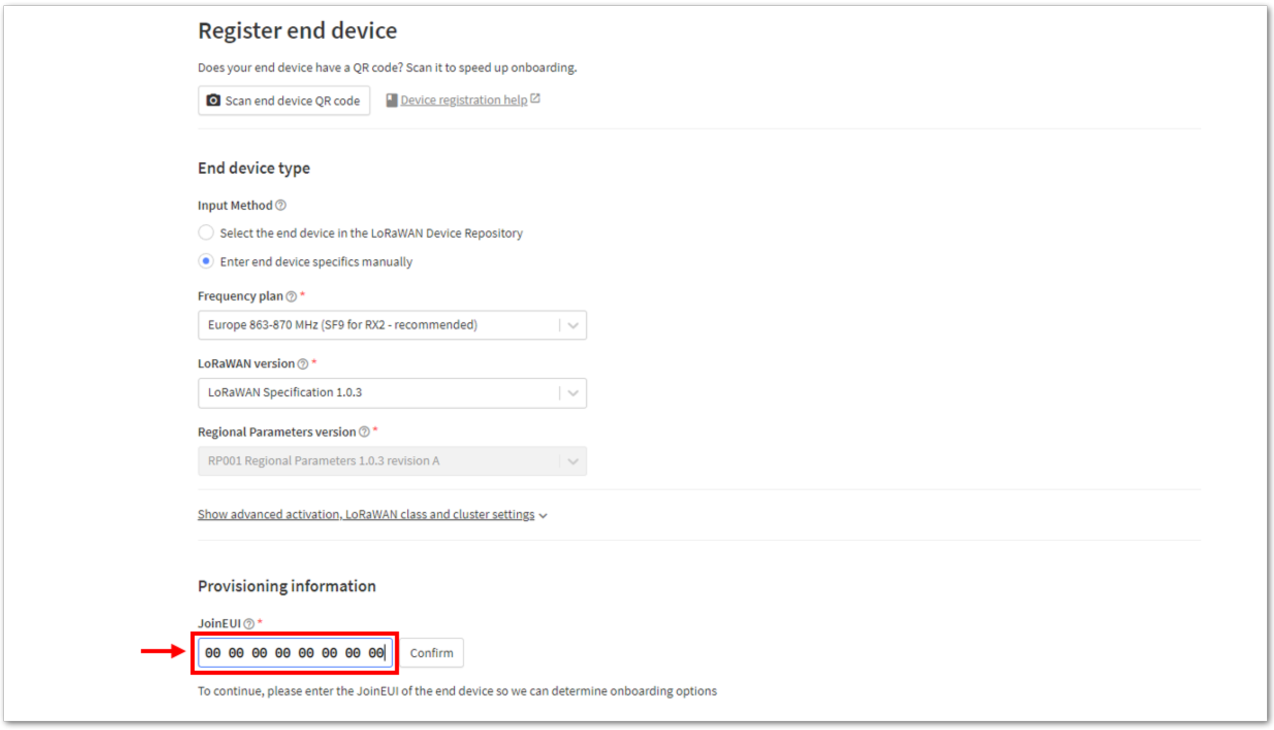
Show advanced activation, LoRaWAN class and cluster settings

Provisioning information

JoinEUI *: ·· ·· ·· ·· ·· ·· ·· ·· | Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 51: Setting up for your device



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *

LoRaWAN Specification 1.0.3

Regional Parameters version *

RP001 Regional Parameters 1.0.3 revision A

Show advanced activation, LoRaWAN class and cluster settings

Provisioning information

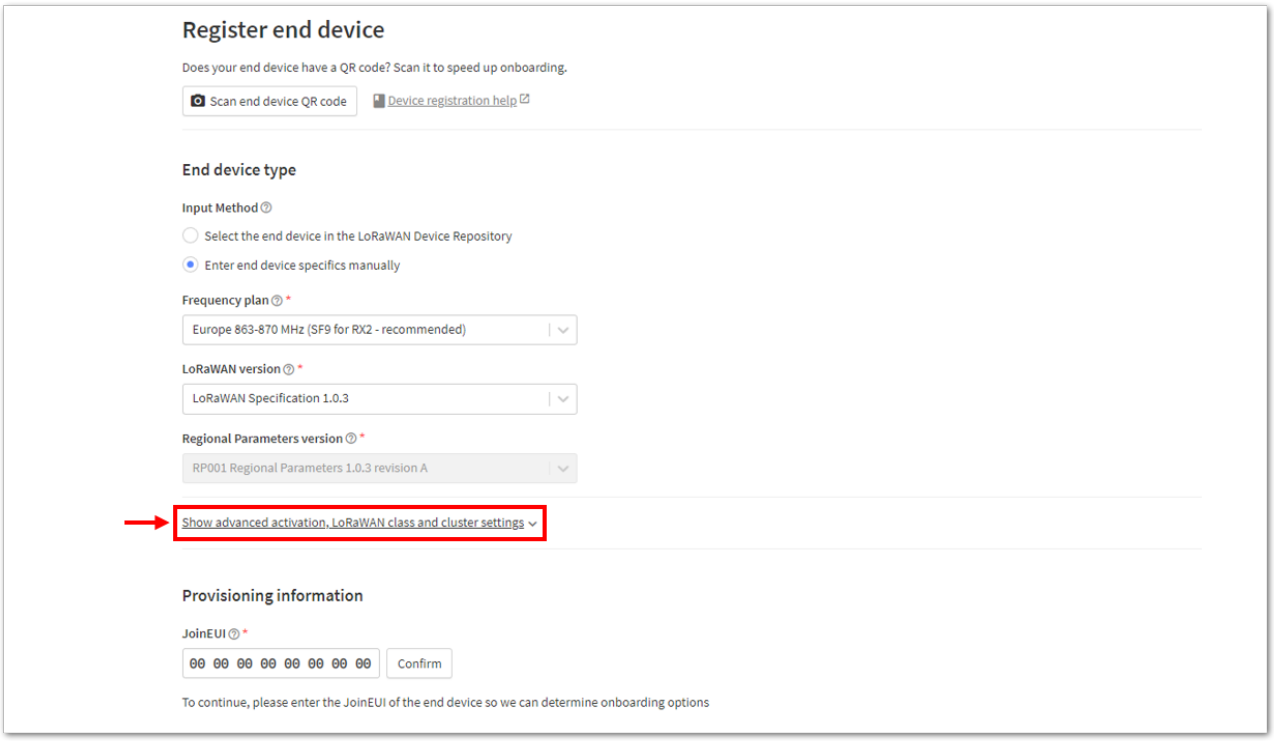
JoinEUI *

00 00 00 00 00 00 00 00 | Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 52: Setting up for your device

4. Then click **Show advanced activation, LoRaWAN class and cluster settings**. Configure the activation mode by selecting **Over the air activation (OTAA)** and Additional LoRaWAN class capabilities to **class A only**. Then click **Confirm**.



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *

LoRaWAN Specification 1.0.3

Regional Parameters version *

RP001 Regional Parameters 1.0.3 revision A

Show advanced activation, LoRaWAN class and cluster settings

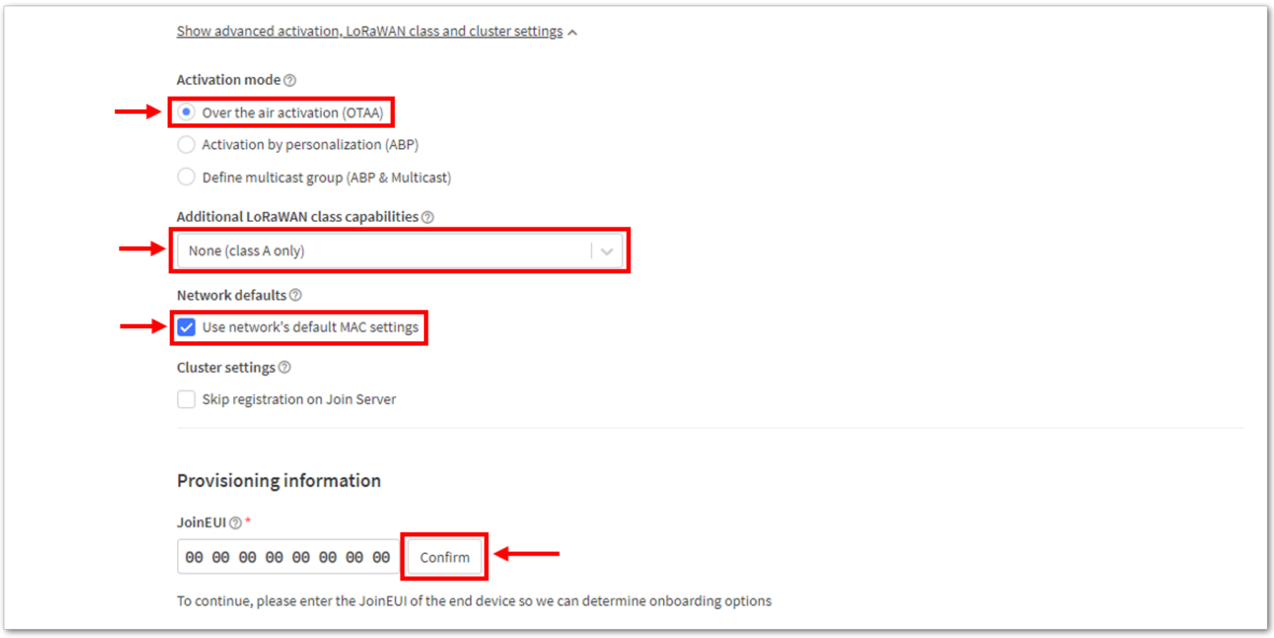
Provisioning information

JoinEUI *

00 00 00 00 00 00 00 00 | Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 53: Setting up for your device



Show advanced activation, LoRaWAN class and cluster settings

Activation mode

● Over the air activation (OTAA)

○ Activation by personalization (ABP)

○ Define multicast group (ABP & Multicast)

Additional LoRaWAN class capabilities

None (class A only)

Network defaults

☑ Use network's default MAC settings

Cluster settings

□ Skip registration on Join Server

Provisioning information

JoinEUI *

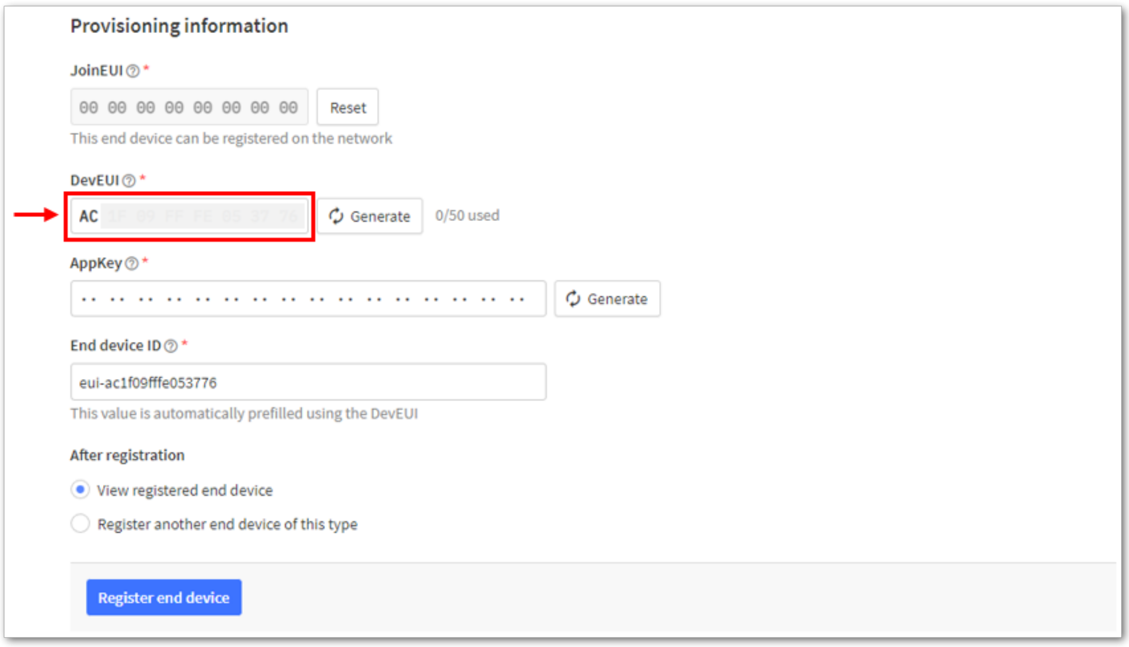
00 00 00 00 00 00 00 00 Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 54: Setting up for your device

5. Once done, provide the DevEUI credentials of your device into the **DevEUI** portion. This will automatically generate the specific end-device ID of your board. Click **Generate** in **AppKey** tab under **Provisioning information** section. Then click **Register end device**.

> 📝 **NOTE:**
>
> - The **AppEUI**, **DevEUI**, and **AppKey** are hidden in this section as these are unique from a specific device. The **DevEUI** credential is unique to every RAK3172 device. Also, you should generate your own **AppEUI** and **AppKey** credentials for your specific device and application.
> - The **AppEUI** is the same as **JoinEUI**.



Provisioning information

JoinEUI *

00 00 00 00 00 00 00 00 Reset

This end device can be registered on the network

DevEUI *

AC [redacted] Generate 0/50 used

AppKey *

•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• Generate

End device ID *

eui-ac1f09fffe053776

This value is automatically prefilled using the DevEUI

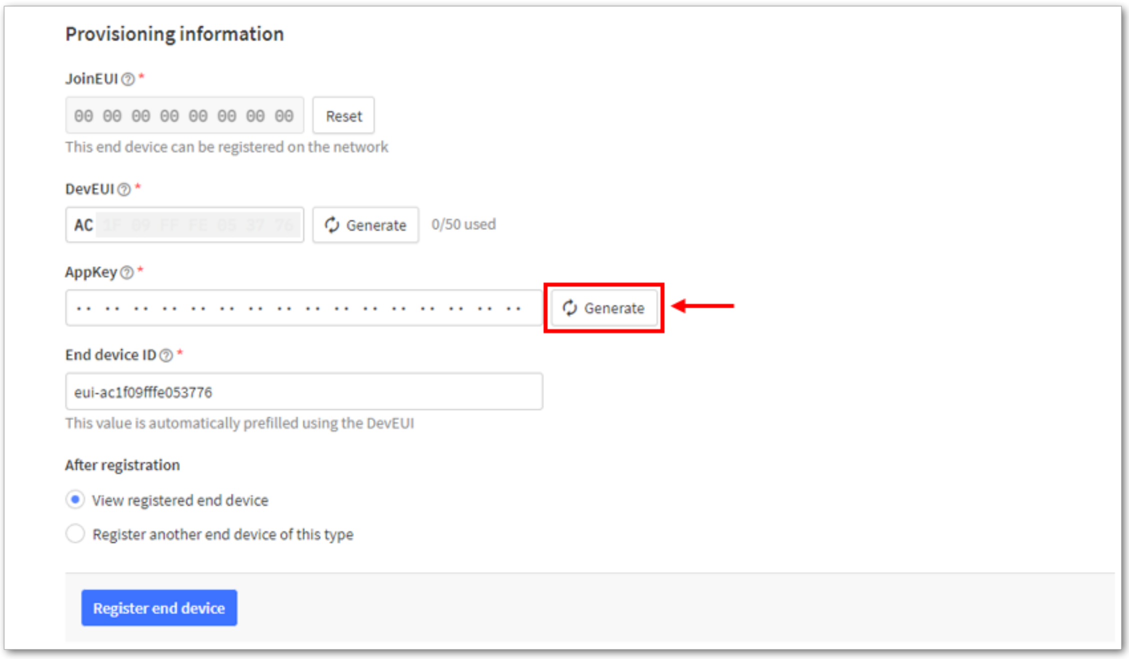
After registration

● View registered end device

○ Register another end device of this type

Register end device

Figure 55: Setting up for your device



Provisioning information

JoinEUI ⓘ *

00 00 00 00 00 00 00 00 | Reset

This end device can be registered on the network

DevEUI ⓘ *

AC [redacted] | Generate | 0/50 used

AppKey ⓘ *

•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• | Generate

End device ID ⓘ *

eui-ac1f09fffe053776

This value is automatically prefilled using the DevEUI

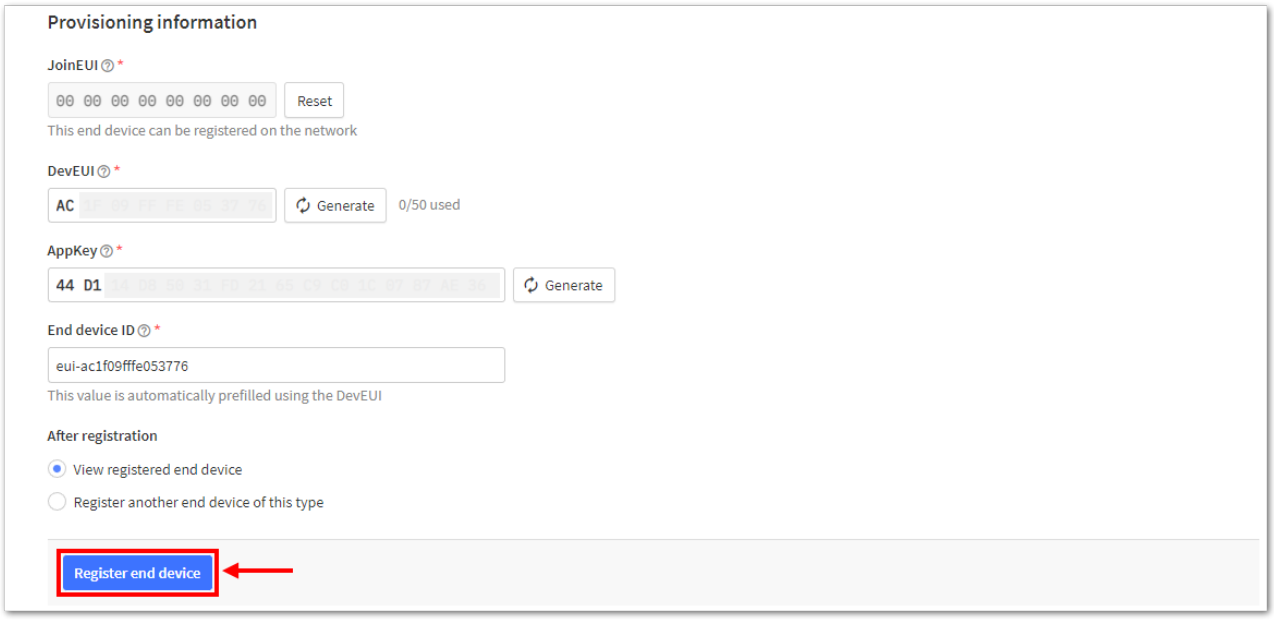
After registration

● View registered end device

○ Register another end device of this type

Register end device

Figure 56: Setting up for your device



Provisioning information

JoinEUI ⓘ *

00 00 00 00 00 00 00 00 | Reset

This end device can be registered on the network

DevEUI ⓘ *

AC [redacted] | Generate | 0/50 used

AppKey ⓘ *

44 D1 [redacted] | Generate

End device ID ⓘ *

eui-ac1f09fffe053776

This value is automatically prefilled using the DevEUI

After registration

● View registered end device

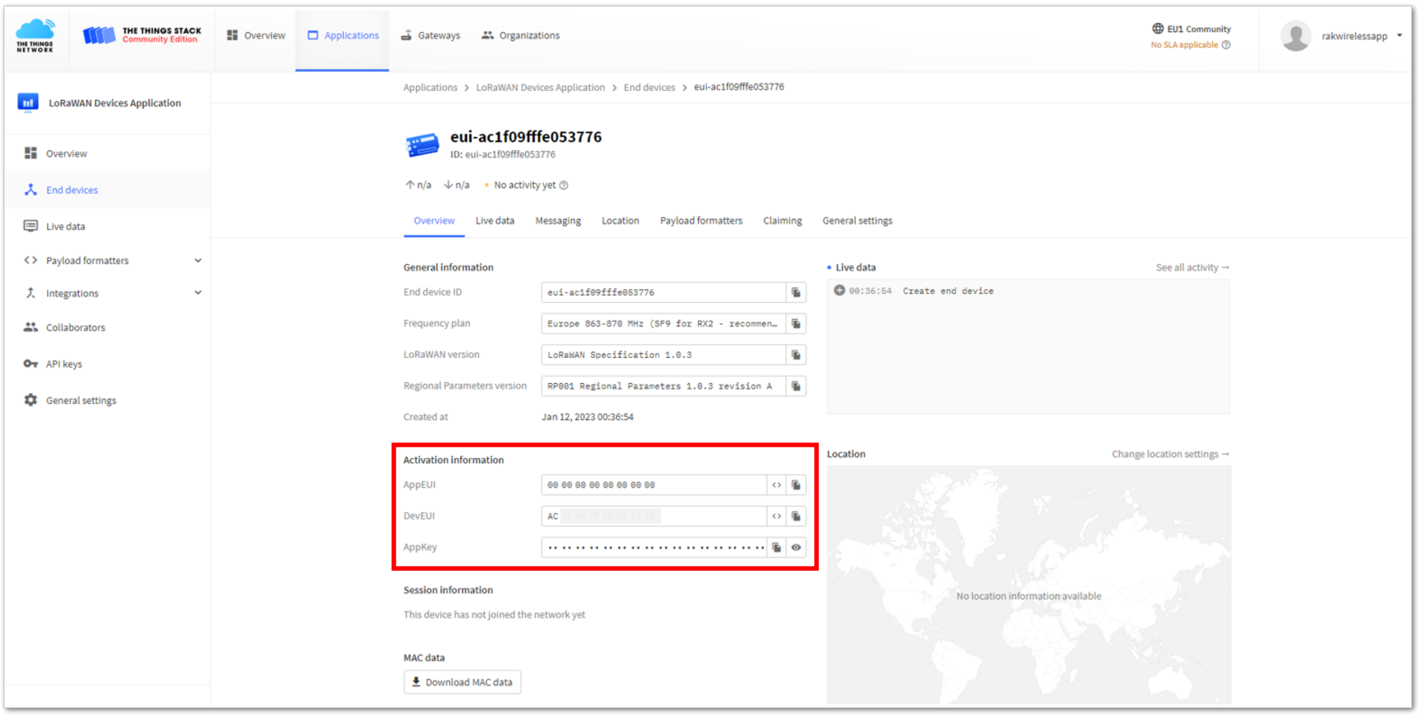
○ Register another end device of this type

Register end device

Figure 57: Register end device

6. You should now be able to see the device on the TTN console after you fully register your device, as shown in **Figure 58**.

> 📝 **NOTE:**
>
> - The **AppEUI**, **DevEUI**, and **AppKey** are the parameters that you will need to activate your LoRaWAN end-device via OTAA. The **AppKey** is hidden by default for security reasons, but you can easily show it by clicking the show button. You can also copy the parameters quickly using the copy button.
> - The three OTAA parameters on the TTN device console are MSB by default.
> - These parameters are always accessible on the device console page, as shown in **Figure 58**.



eui-ac1f09fffe053776
ID: eui-ac1f09fffe053776

General information

End device ID: eui-ac1f09fffe053776
Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version: LoRaWAN Specification 1.0.3
Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A
Created at: Jan 12, 2023 00:36:54

Live data

```
00:36:54  Create end device
```

Activation information

AppEUI: 00 00 00 00 00 00 00 00
DevEUI: AC [redacted]
AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Session information

This device has not joined the network yet

MAC data

Location

Figure 58: OTAA device successfully registered to TTN

OTAA Configuration for TTN

The RAK3172 module can be configured using WisToolBox to do the OTAA configuration. **WisToolBox** is a software tool that supports **RAK3172** module. It automatically detects the RAK3172 module once it is connected to the PC. Below are the options in WisToolBox where the OTAA configuration can be done.

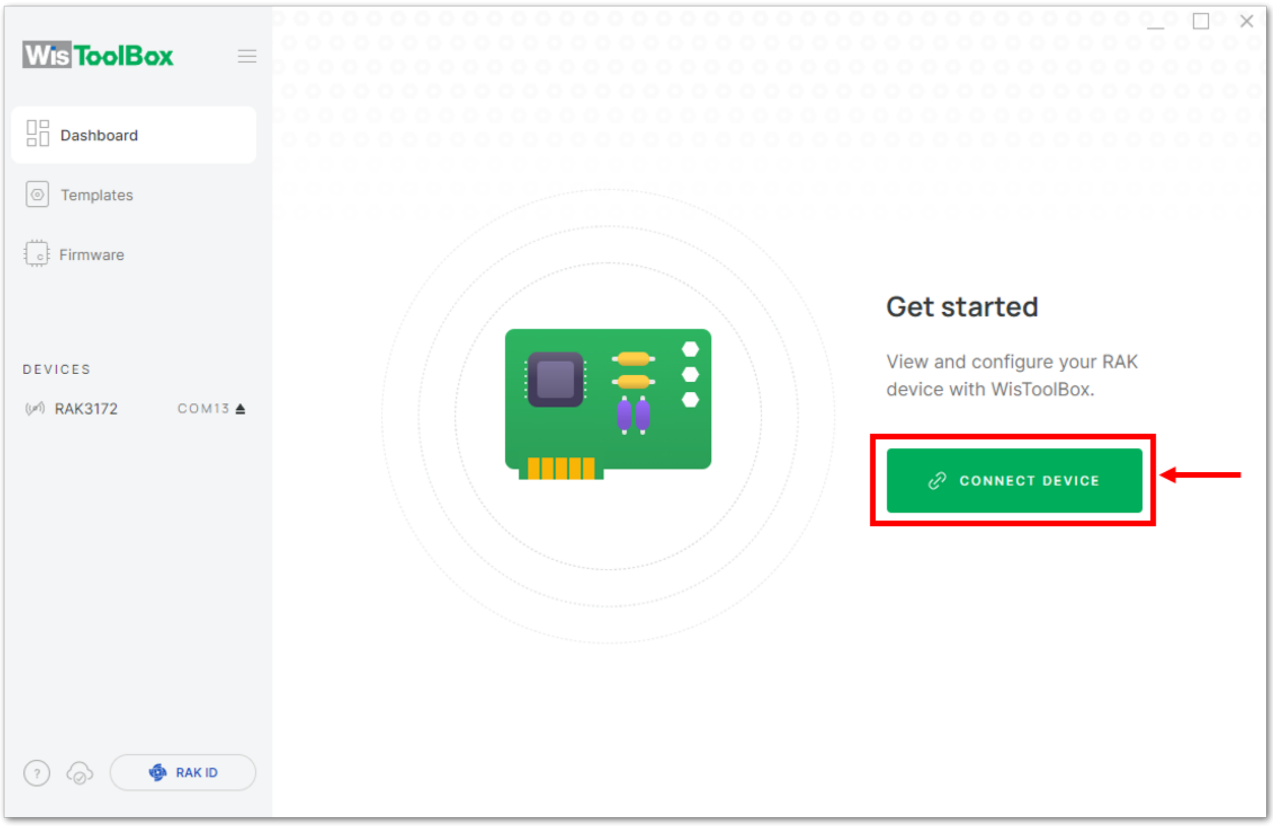
- OTAA Configuration for TTN via WisToolBox UI
- OTAA Configuration for TTN via WisToolBox Console

OTAA Configuration for TTN via WisToolBox UI

The **RAK3172** should have the correct OTAA credentials to connect to TTN. This can be done using **WisToolBox UI**. Below are the steps on setting up your **RAK3172** using **WisToolBox**.

1. Connect your **RAK3172** with your chosen WisBlock base board to the PC via USB cable and open the **WisToolBox** application.

2. Click **CONNECT DEVICE** button to launch the WisToolBox Dashboard.



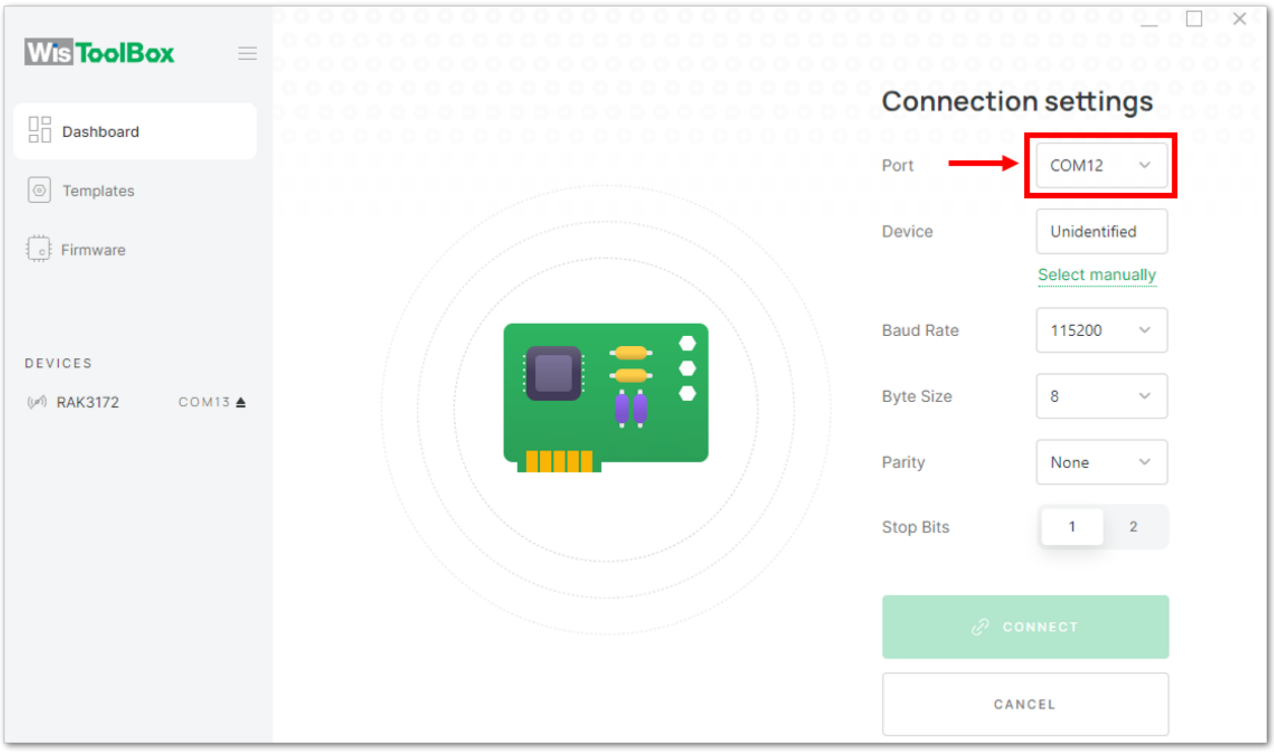
Get started

View and configure your RAK device with WisToolBox.

CONNECT DEVICE

Figure 59: CONNECT DEVICE

3. Then select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT**, as shown in **Figure 60**.



Connection settings

Port: COM12

Device: Unidentified

Select manually

Baud Rate: 115200

Byte Size: 8

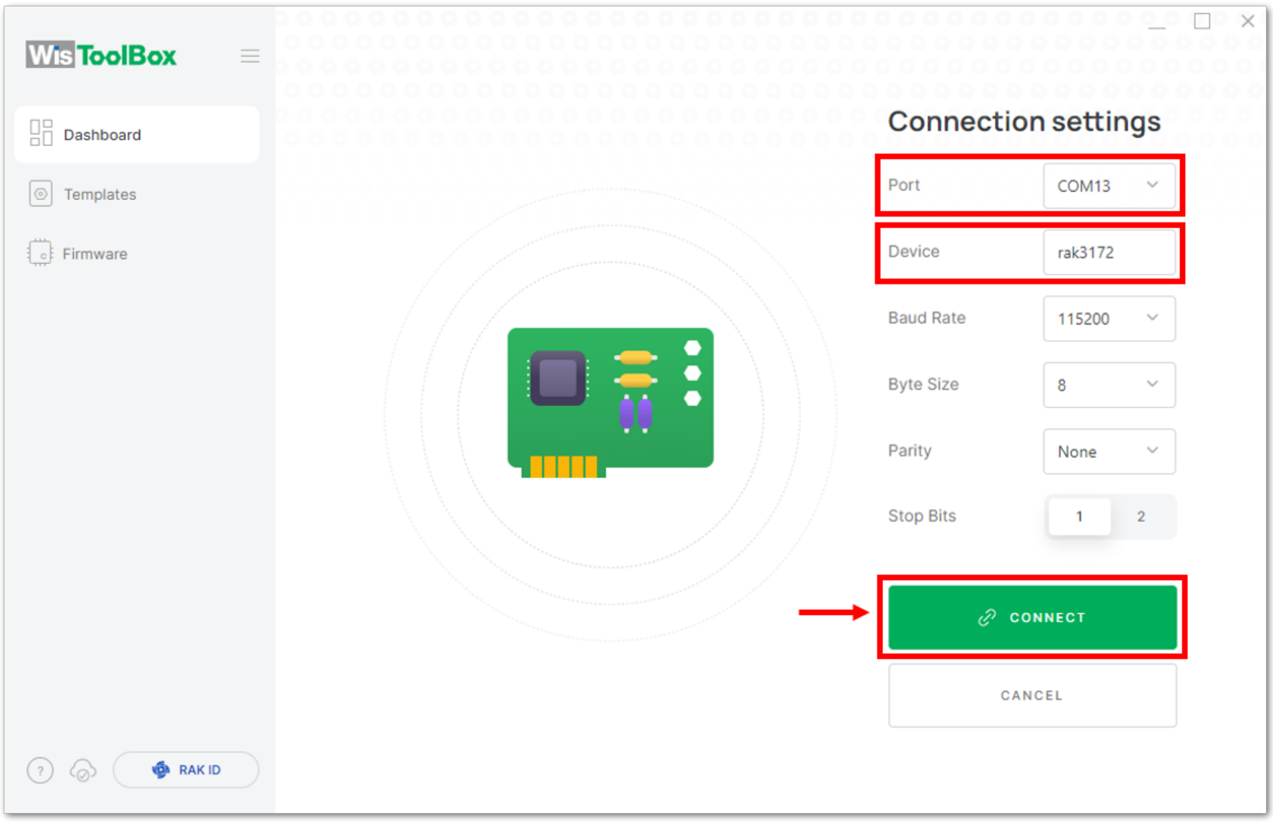
Parity: None

Stop Bits: 1 2

CONNECT

CANCEL

Figure 60: Setting up your device



Connection settings

Port: COM13

Device: rak3172

Baud Rate: 115200

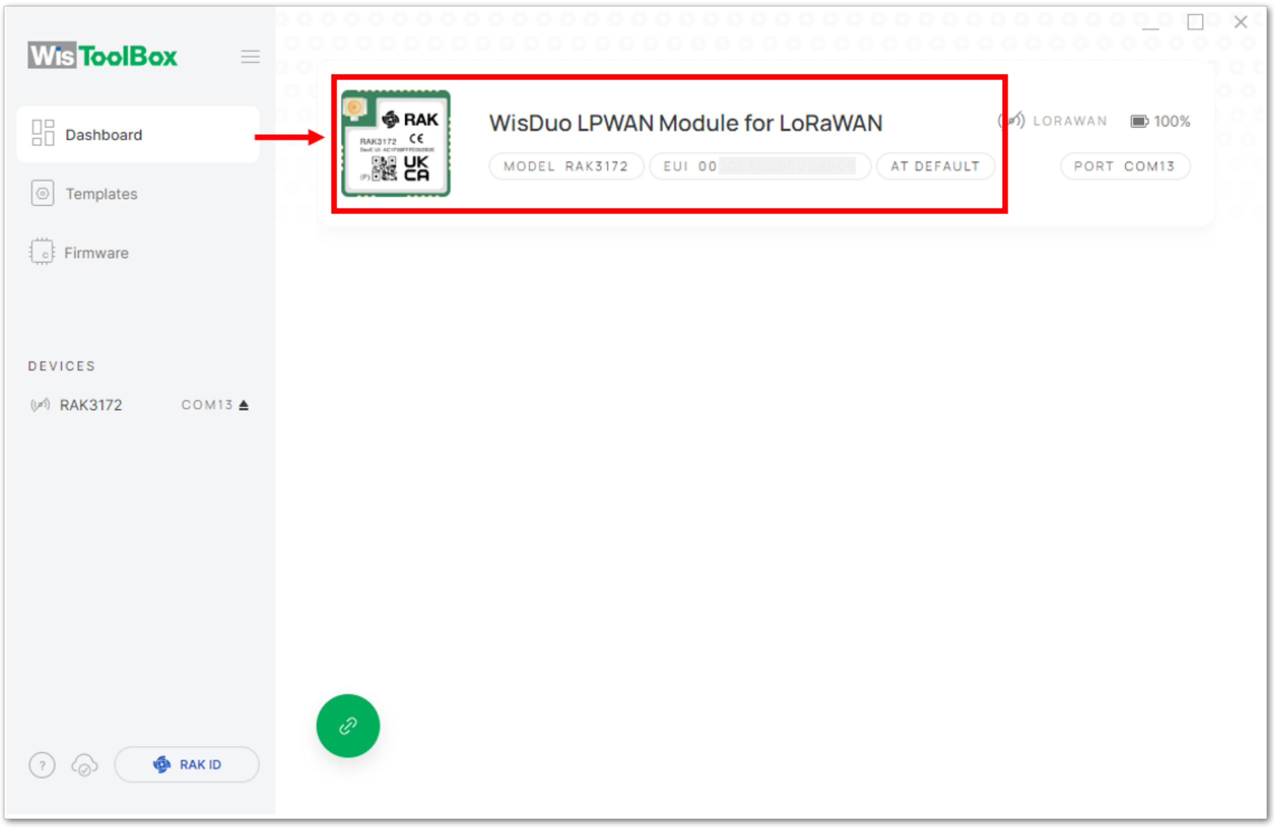
Byte Size: 8

Parity: None

Stop Bits: 1 2

Figure 61: Setting up your device

4. Once done, **RAK3172** will appear in the dashboard then select it.



WisDuo LPWAN Module for LoRaWAN

MODEL RAK3172 | EUI 00[redacted] | AT DEFAULT

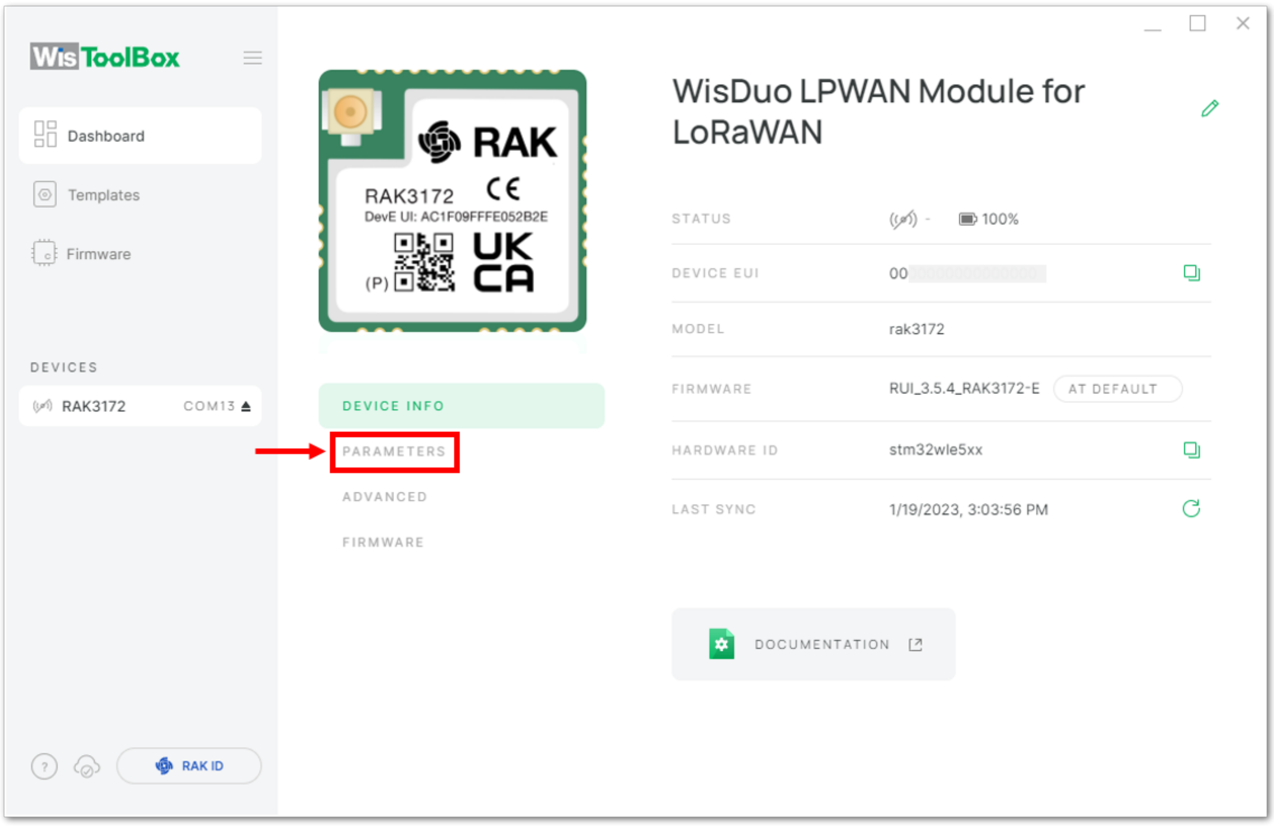
LORAWAN | 100% | PORT COM13

DEVICES

RAK3172 COM13

Figure 62: Device seen from WisToolBox dashboard

5. Then click **PARAMETERS** configure your RAK3172.



WisDuo LPWAN Module for LoRaWAN

STATUS	- 100%
DEVICE EUI	00[redacted]
MODEL	rak3172
FIRMWARE	RUI_3.5.4_RAK3172-E
HARDWARE ID	stm32wle5xx
LAST SYNC	1/19/2023, 3:03:56 PM

Figure 63: Setting up your device

6. Click **Global settings** to set the network mode to **LoRaWAN** and join mode to **OTAA**. Make sure that the active region is using **EU868** for this configuration. If you wish to work on other regional bands, you can choose among active regions based on your location.

- LoRa network mode: **LoRaWAN**
- LoRaWAN join mode: **OTAA**
- LoRaWAN region: **EU868**

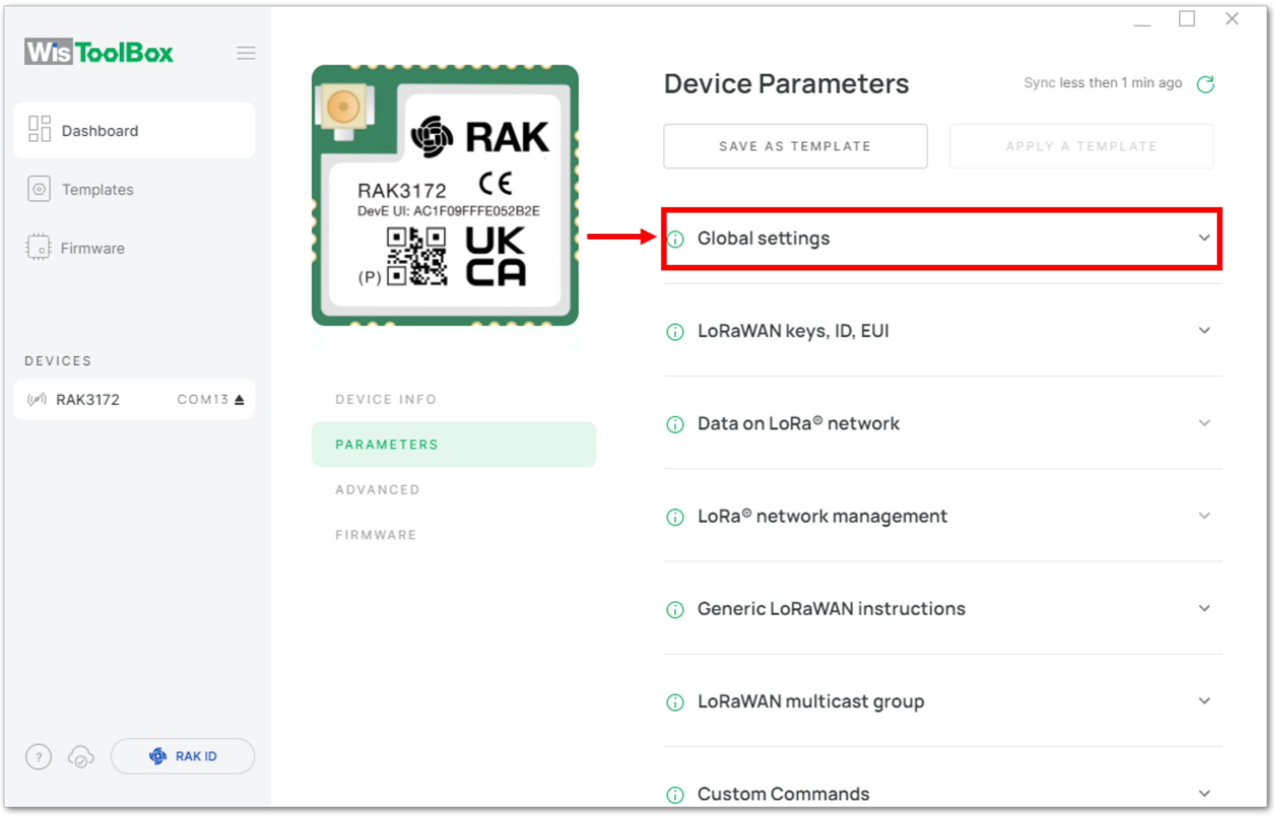
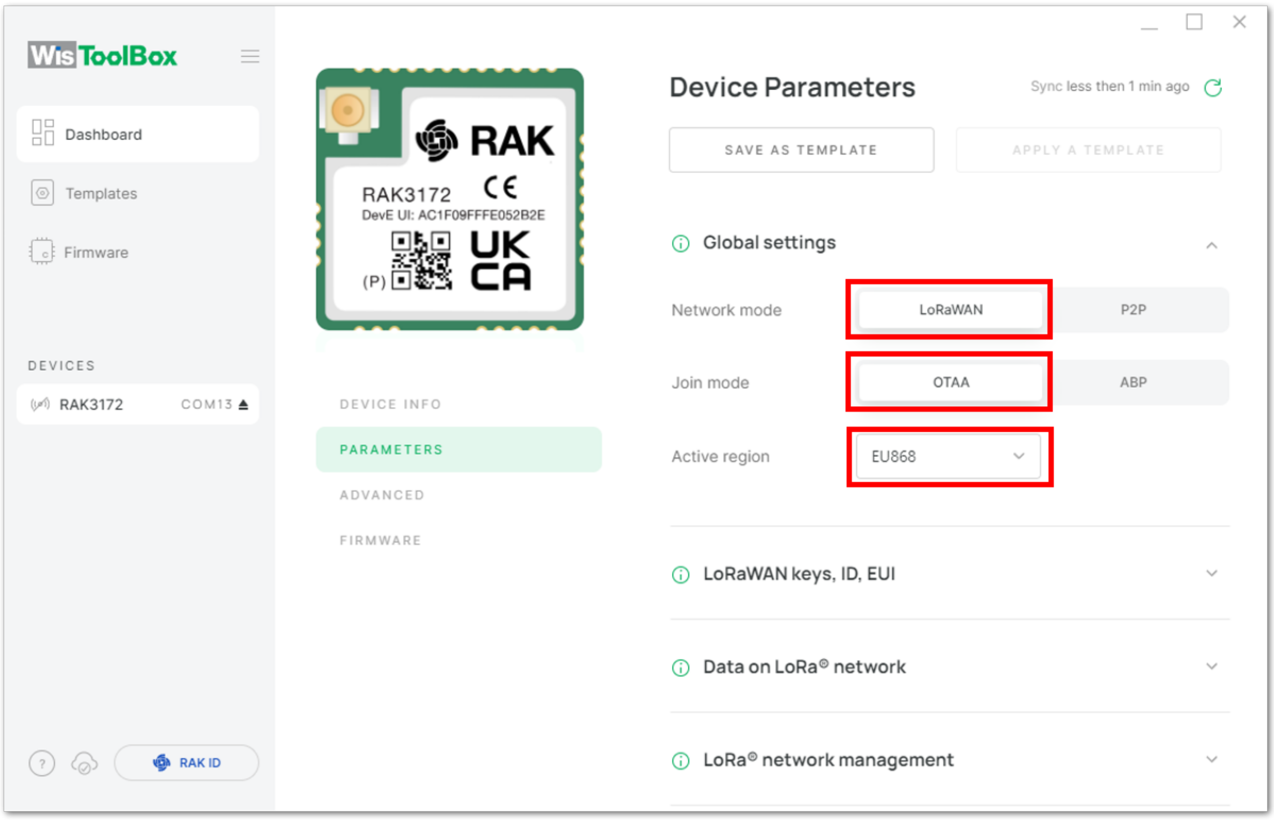


Figure 64: Global settings



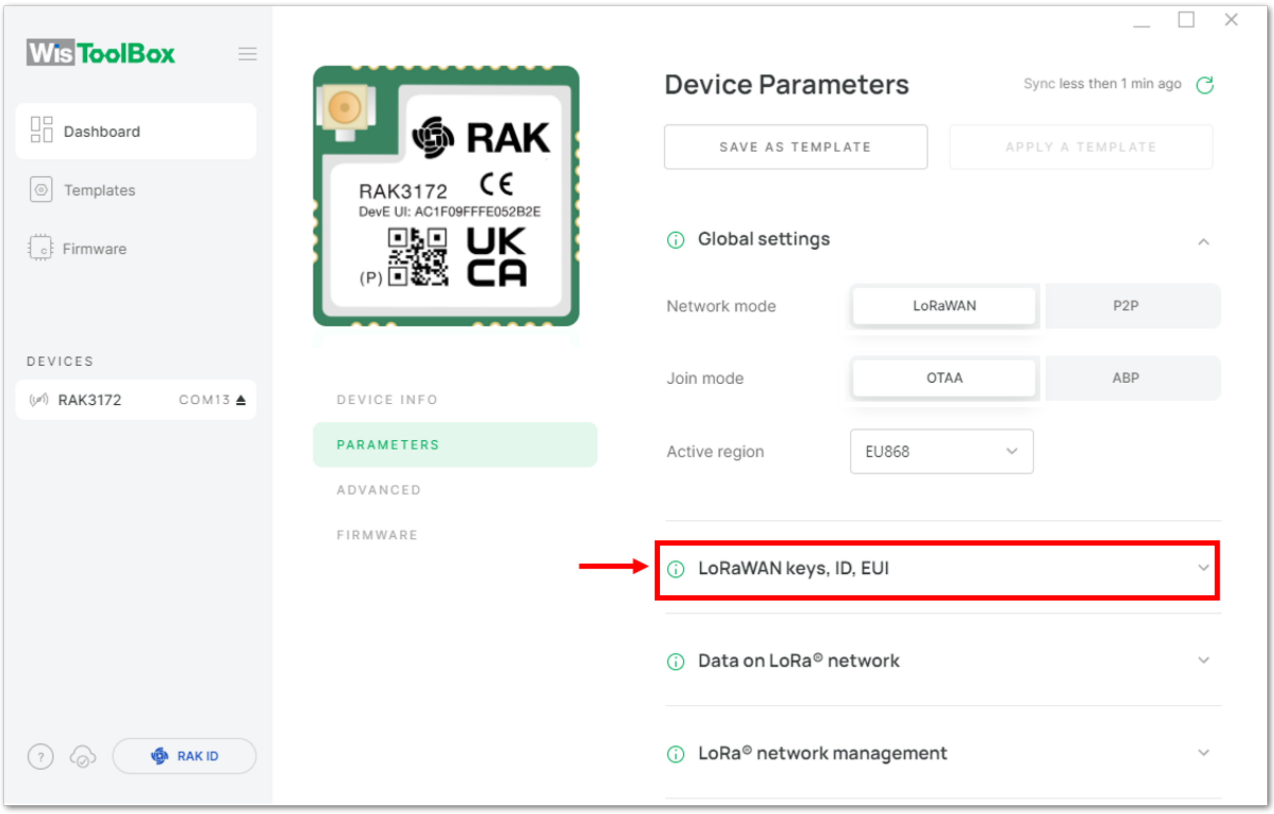
Network mode: LoRaWAN

Join mode: OTAA

Active region: EU868

Figure 65: Global settings

7. Click **LoRaWAN keys, ID, EUI** to configure the **Application EUI (AppEUI)**, **Application key (AppKey)**, and **Device EUI (DevEUI)**.

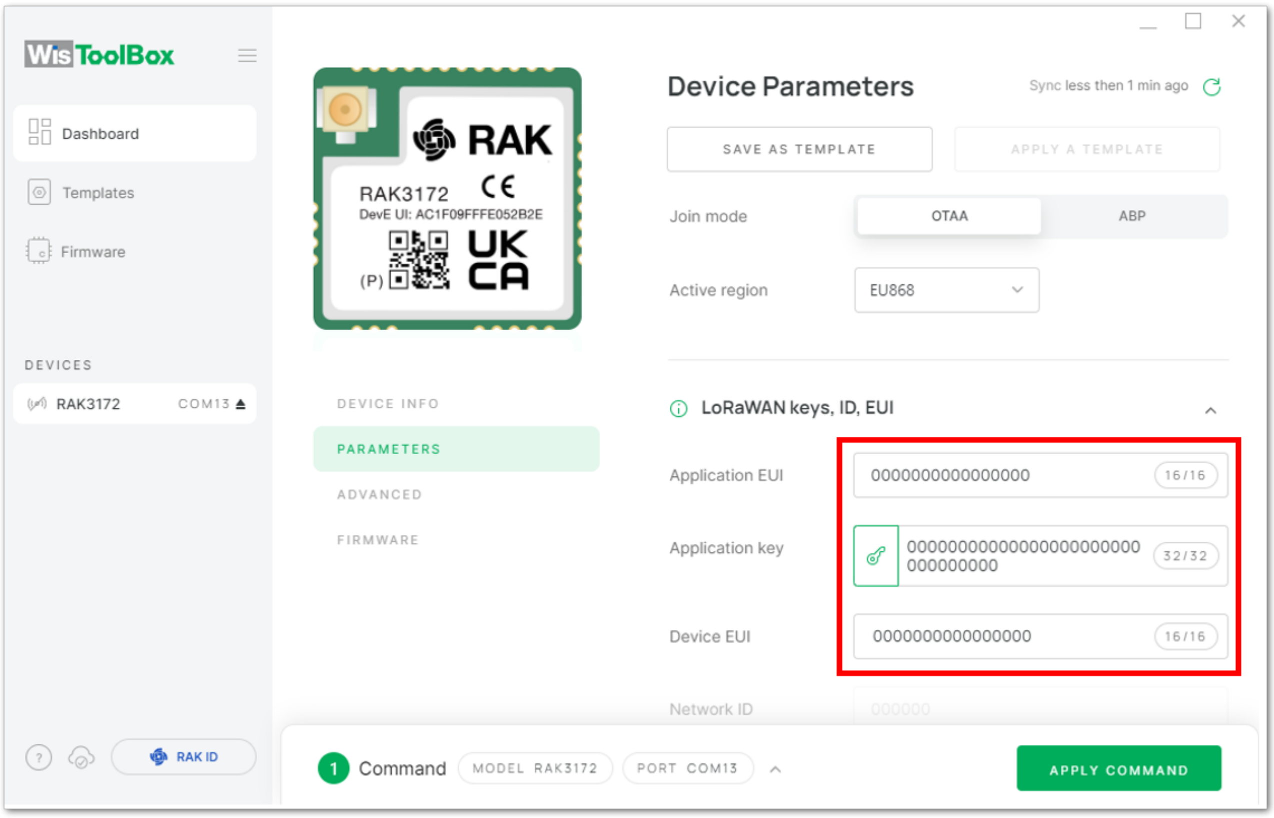


Network mode: LoRaWAN

Join mode: OTAA

Active region: EU868

Figure 66: LoRaWAN keys, ID, EUI



Device Parameters

Sync less then 1 min ago

Join mode: OTAA | ABP

Active region: EU868

LoRaWAN keys, ID, EUI

Application EUI: 0000000000000000 16/16

Application key: 00000000000000000000000000000000 32/32

Device EUI: 0000000000000000 16/16

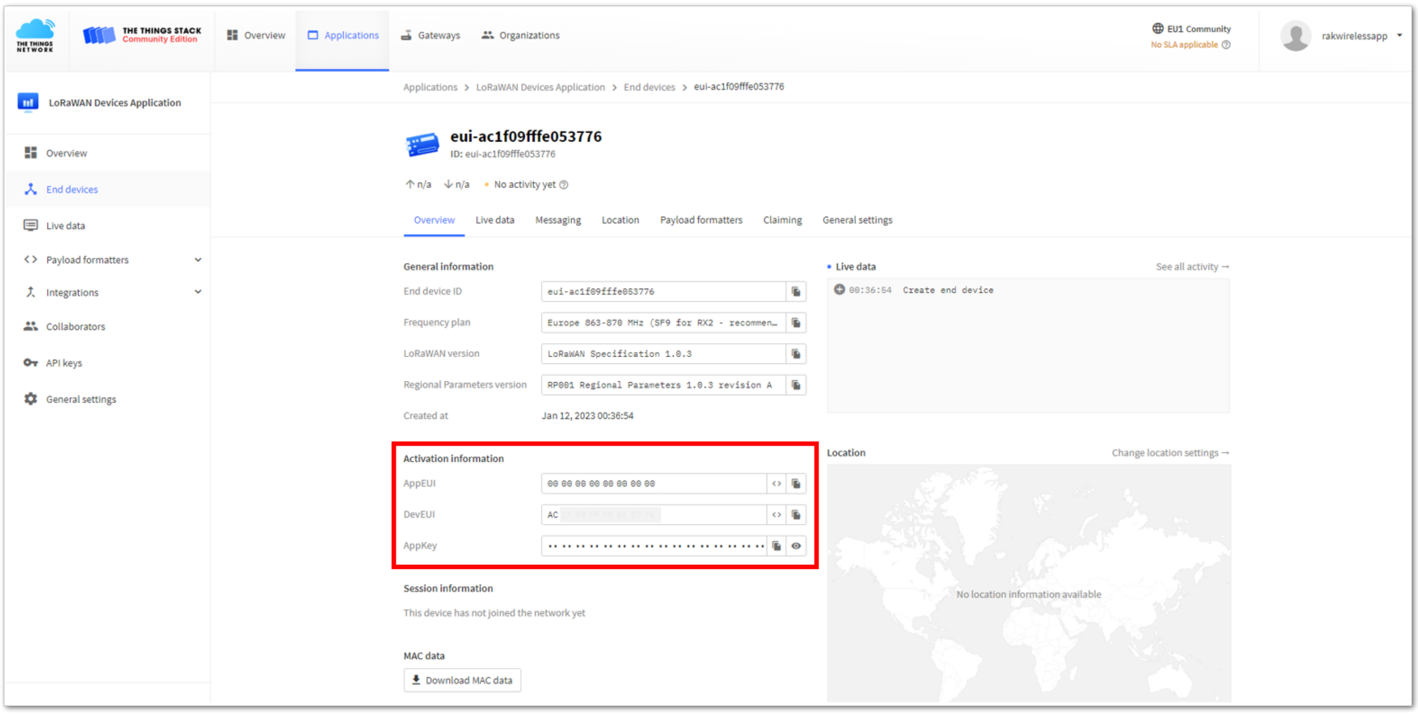
Network ID: 000000

1 Command MODEL RAK3172 PORT COM13

Figure 67: Setting up your device

8. Then go back to the console where your RAK3172 end-device is created previously. Copy the credentials from there since it will be used in the WisToolBox dashboard. Once encoded into the dashboard, click **APPLY COMMAND** to update your device, as shown in **Figure 75**.

> 📝 **NOTE:**
>
> - The **AppEUI**, **DevEUI**, and **AppKey** are hidden in this section as these are unique from a specific device.



eui-ac1f09fffe053776

ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

General information

End device ID: eui-ac1f09fffe053776

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 12, 2023 00:36:54

Live data

00:36:54 Create end device

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: AC [redacted]

AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Session information

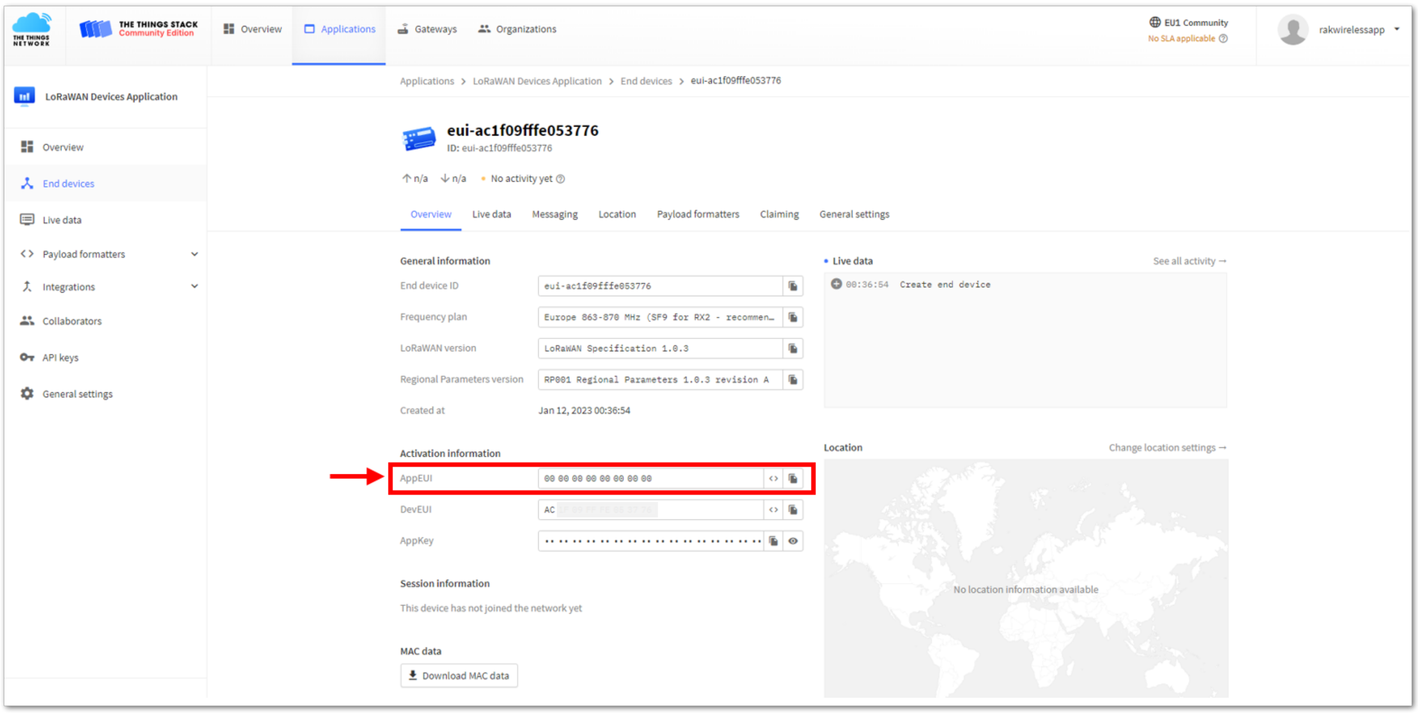
This device has not joined the network yet

MAC data

Location

Figure 68: Your created OTAA device from your console

- **For Application EUI (AppEUI)**



eui-ac1f09fffe053776

ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

Overview | Live data | Messaging | Location | Payload formatters | Claiming | General settings

General information

End device ID: eui-ac1f09fffe053776

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 12, 2023 00:36:54

00:36:54 Create end device

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: AC [redacted]

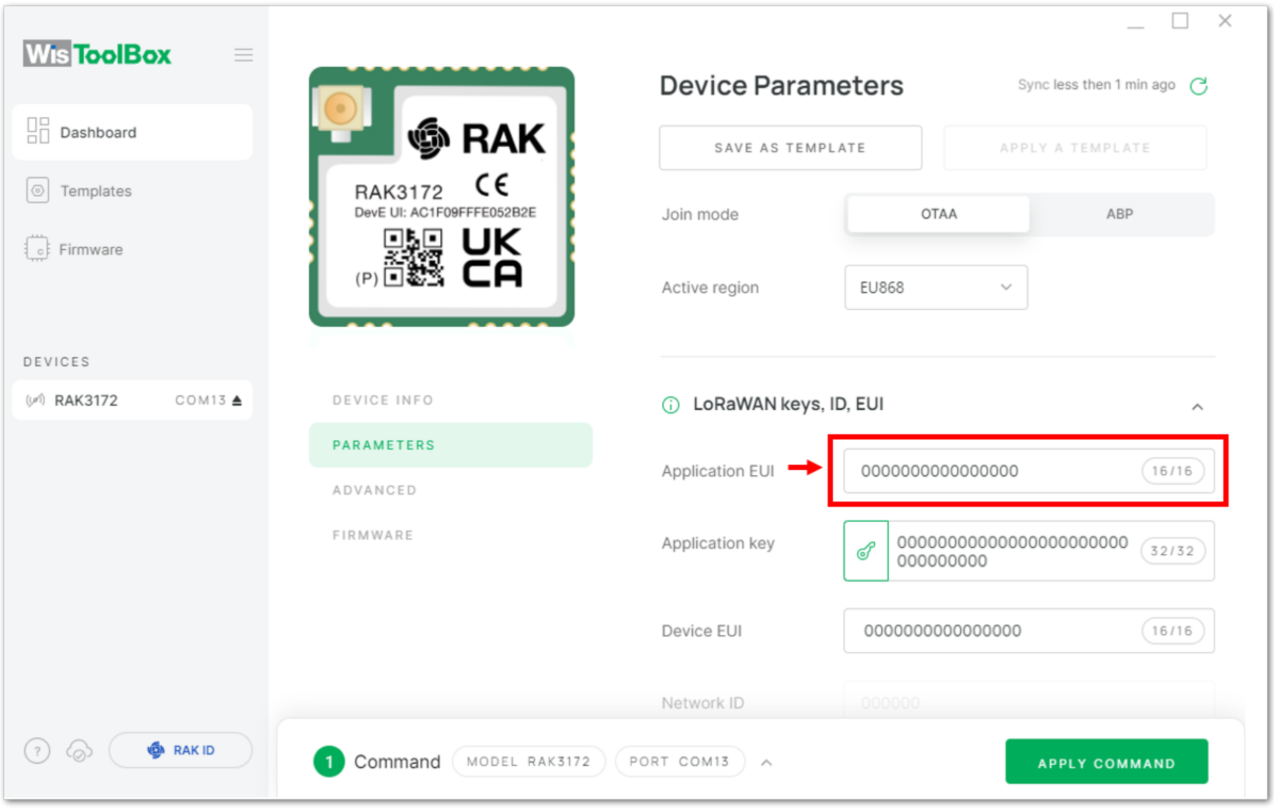
AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Session information

This device has not joined the network yet

MAC data

Figure 69: Copying the AppEUI credential from TTN to WisToolBox



Device Parameters

Sync less then 1 min ago

Join mode: OTAA | ABP

Active region: EU868

LoRaWAN keys, ID, EUI

Application EUI: 0000000000000000 16/16

Application key: 00000000000000000000000000000000 32/32

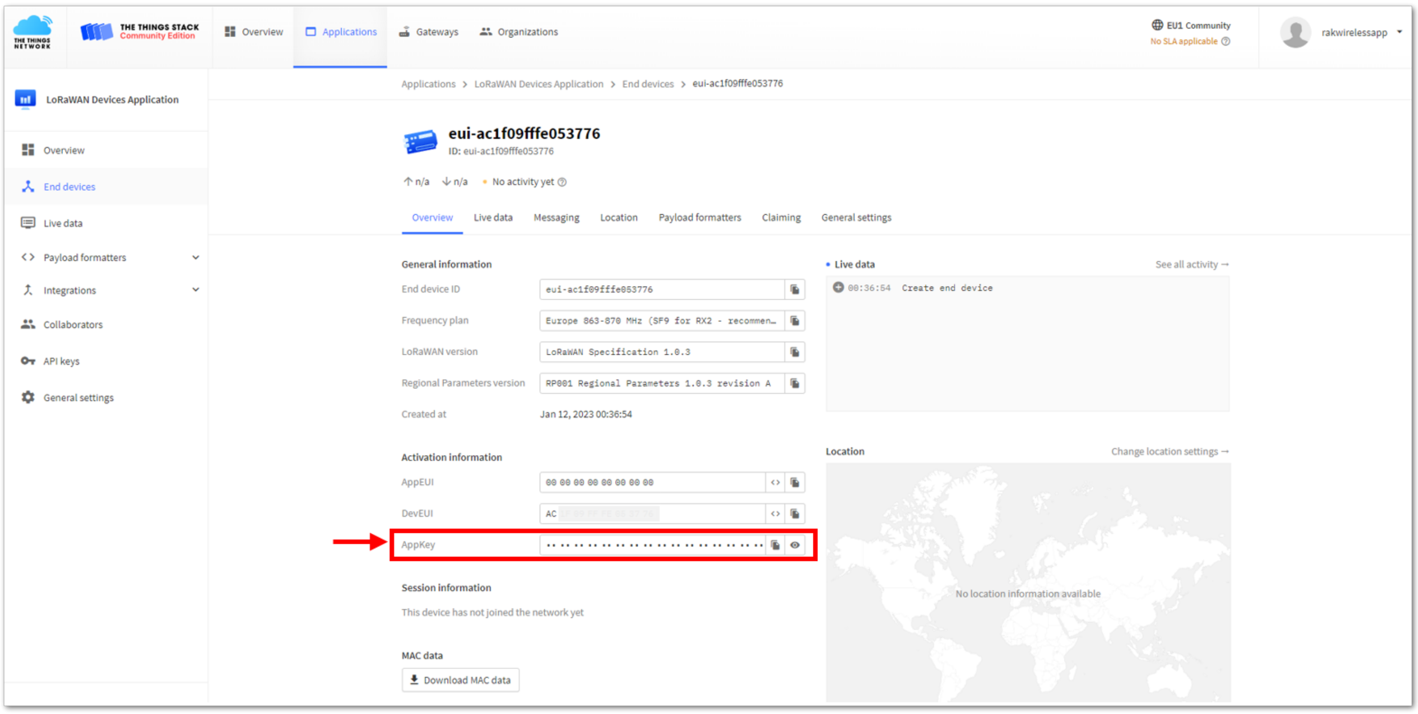
Device EUI: 0000000000000000 16/16

Network ID: 000000

1 Command MODEL RAK3172 PORT COM13

Figure 70: Copying the AppEUI credential from TTN to WisToolBox

- **For Application key (AppKey)**



eui-ac1f09fffe053776
ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

General information

End device ID: eui-ac1f09fffe053776
Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version: LoRaWAN Specification 1.0.3
Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A
Created at: Jan 12, 2023 00:36:54

Live data

```
00:36:54   Create end device
```

Activation information

AppEUI: 00 00 00 00 00 00 00 00
DevEUI: AC[redacted]
AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

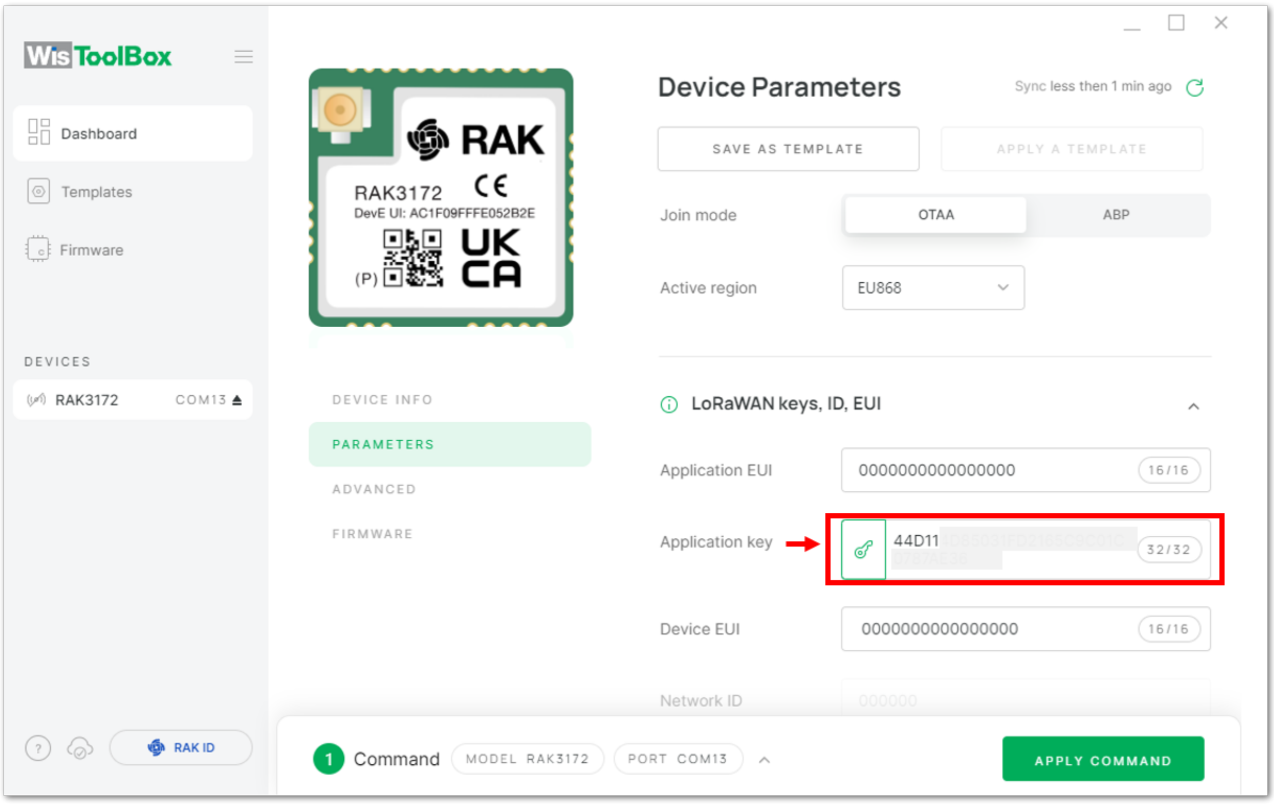
Location

Session information

This device has not joined the network yet

MAC data

Figure 71: Copying the AppKey credential from TTN to WisToolBox



Device Parameters

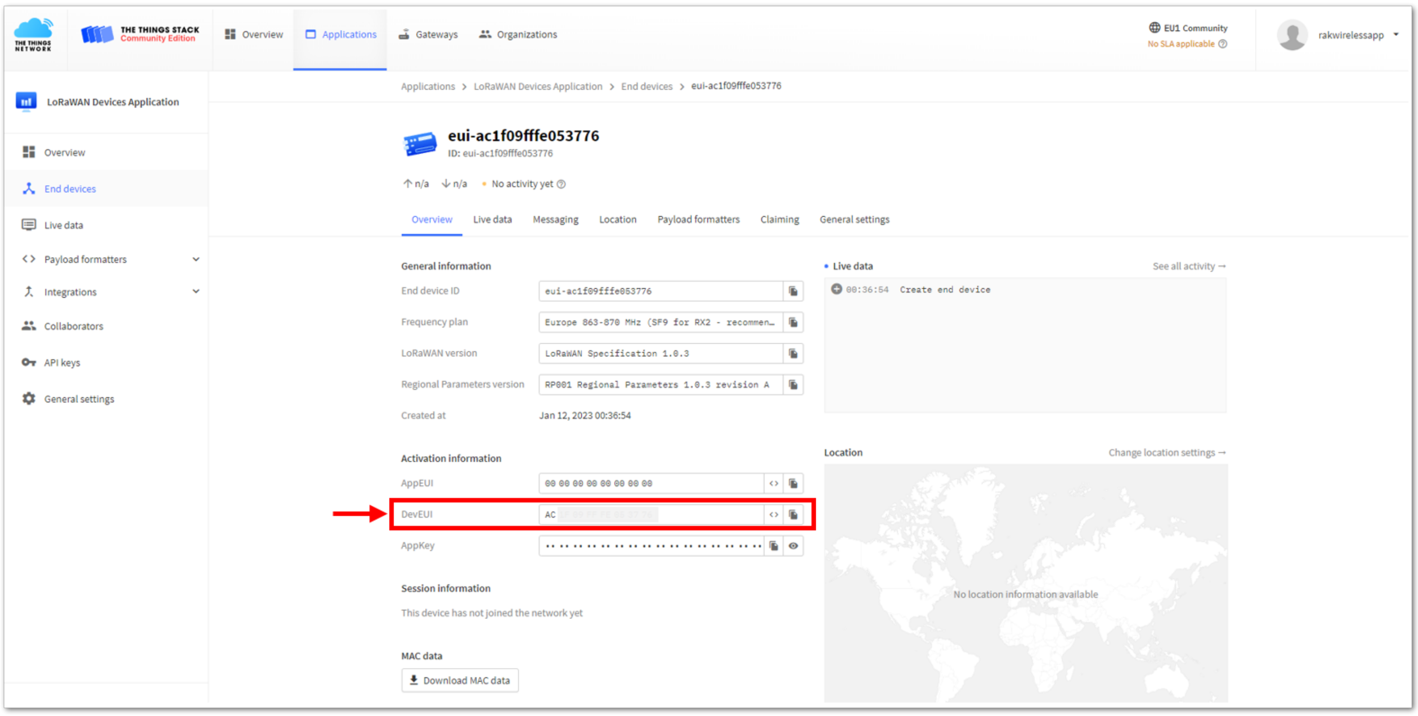
Join mode: OTAA | ABP
Active region: EU868

LoRaWAN keys, ID, EUI

Application EUI: 0000000000000000 16/16
Application key: 44D11[redacted] 32/32
Device EUI: 0000000000000000 16/16
Network ID: 000000

Figure 72: Copying the AppKey credential from TTN to WisToolBox

- **For Device EUI (DevEUI)**



eui-ac1f09fffe053776
ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

General information

End device ID: eui-ac1f09fffe053776

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 12, 2023 00:36:54

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: AC [redacted]

AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Session information

This device has not joined the network yet

MAC data

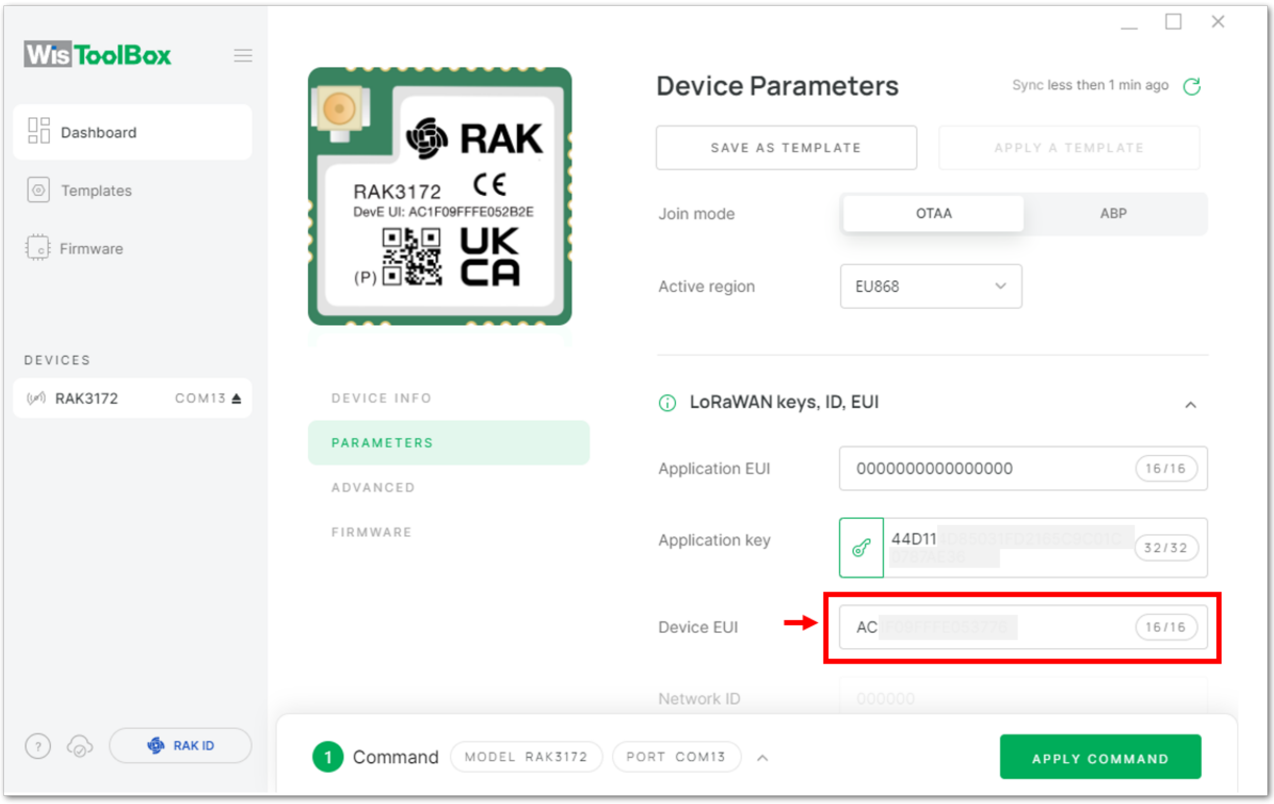
• Live data

00:36:54 Create end device

Location

No location information available

Figure 73: Copying the DevEUI credential from TTN to WisToolBox



Device Parameters

Sync less then 1 min ago

Join mode: OTAA | ABP

Active region: EU868

LoRaWAN keys, ID, EUI

Application EUI: 0000000000000000 16/16

Application key: 44D11 [redacted] 32/32

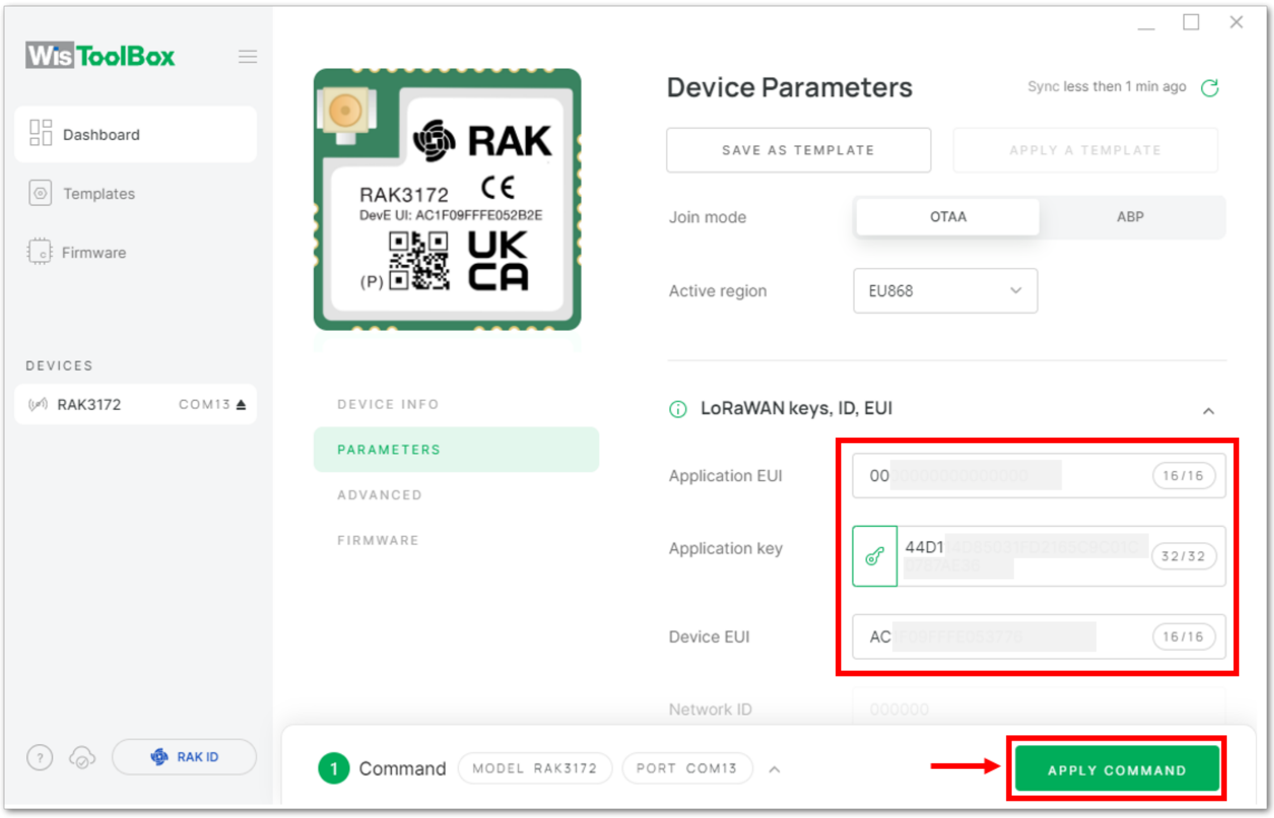
Device EUI: AC [redacted] 16/16

Network ID: 000000

1 Command MODEL RAK3172 PORT COM13

Figure 74: Copying the DevEUI credential from TTN to WisToolBox

- **WisToolBox Dashboard**



Device Parameters

Join mode: OTAA | ABP

Active region: EU868

LoRaWAN keys, ID, EUI

Application EUI: 00[redacted] 16/16

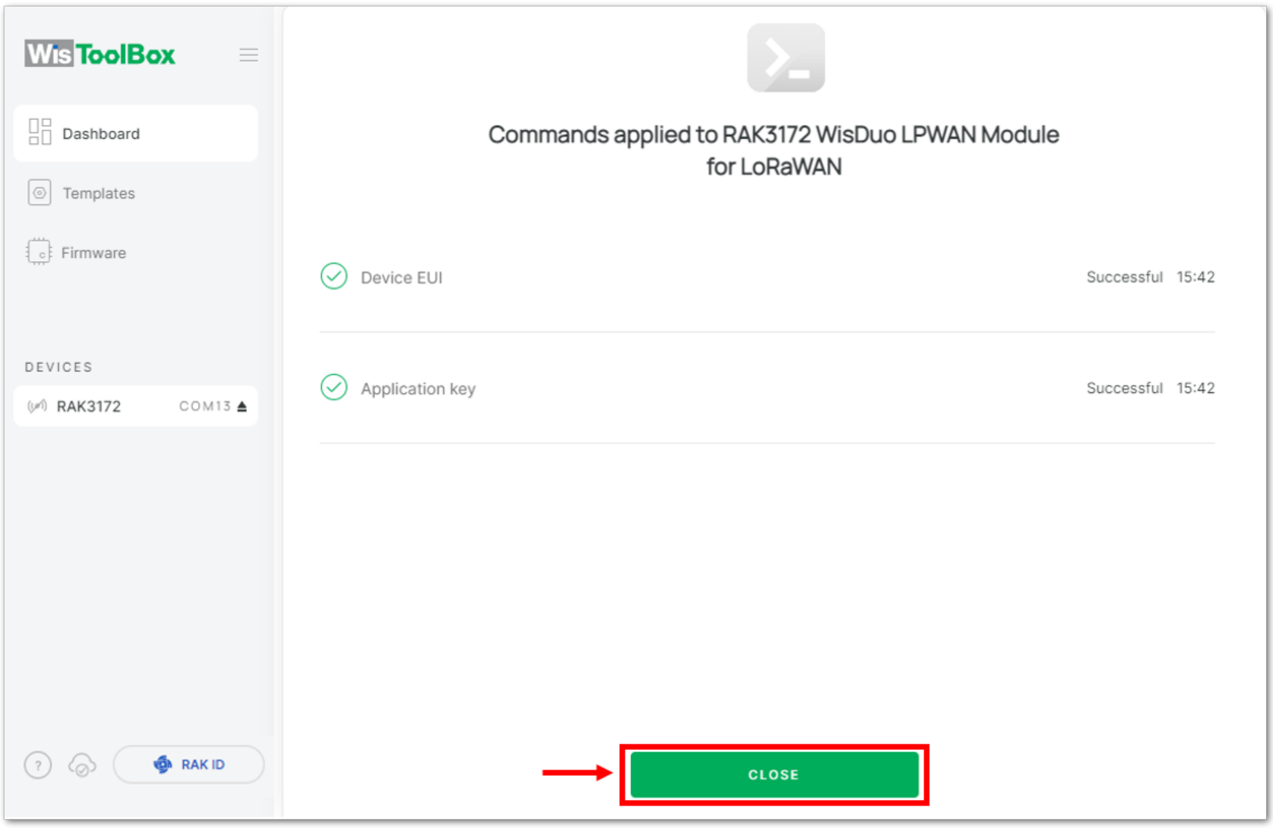
Application key: 44D1[redacted] 32/32

Device EUI: AC[redacted] 16/16

Network ID: 000000

Figure 75: Used credentials from your console in WisToolBox dashboard

9. Once done, you will see the summary of commands that is applied to your device. Then click **CLOSE**.



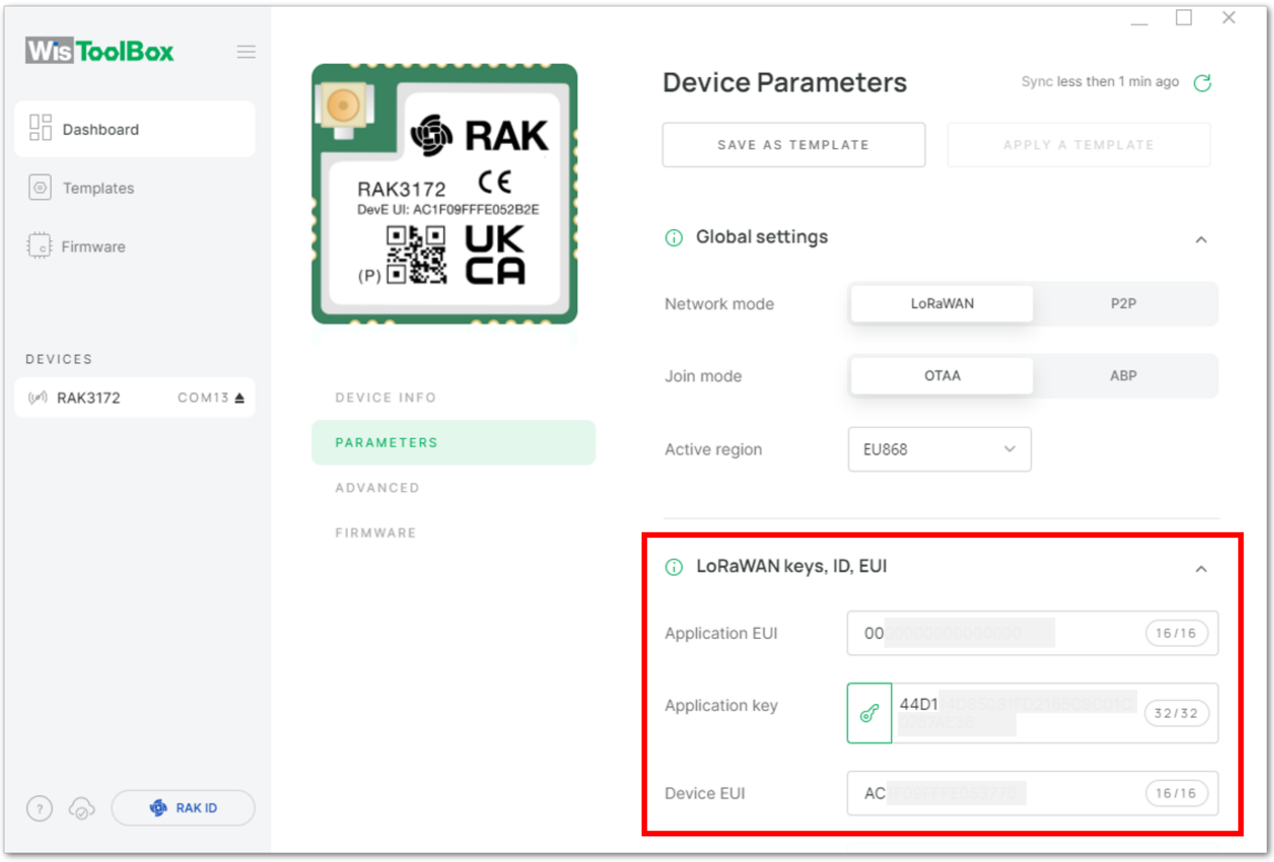
Commands applied to RAK3172 WisDuo LPWAN Module for LoRaWAN

Device EUI Successful 15:42

Application key Successful 15:42

Figure 76: Summary of commands

10. Now, you will see it return to the dashboard with updated credentials of your device.



Device Parameters

Sync less then 1 min ago

Global settings

Network mode: LoRaWAN | P2P

Join mode: OTAA | ABP

Active region: EU868

LoRaWAN keys, ID, EUI

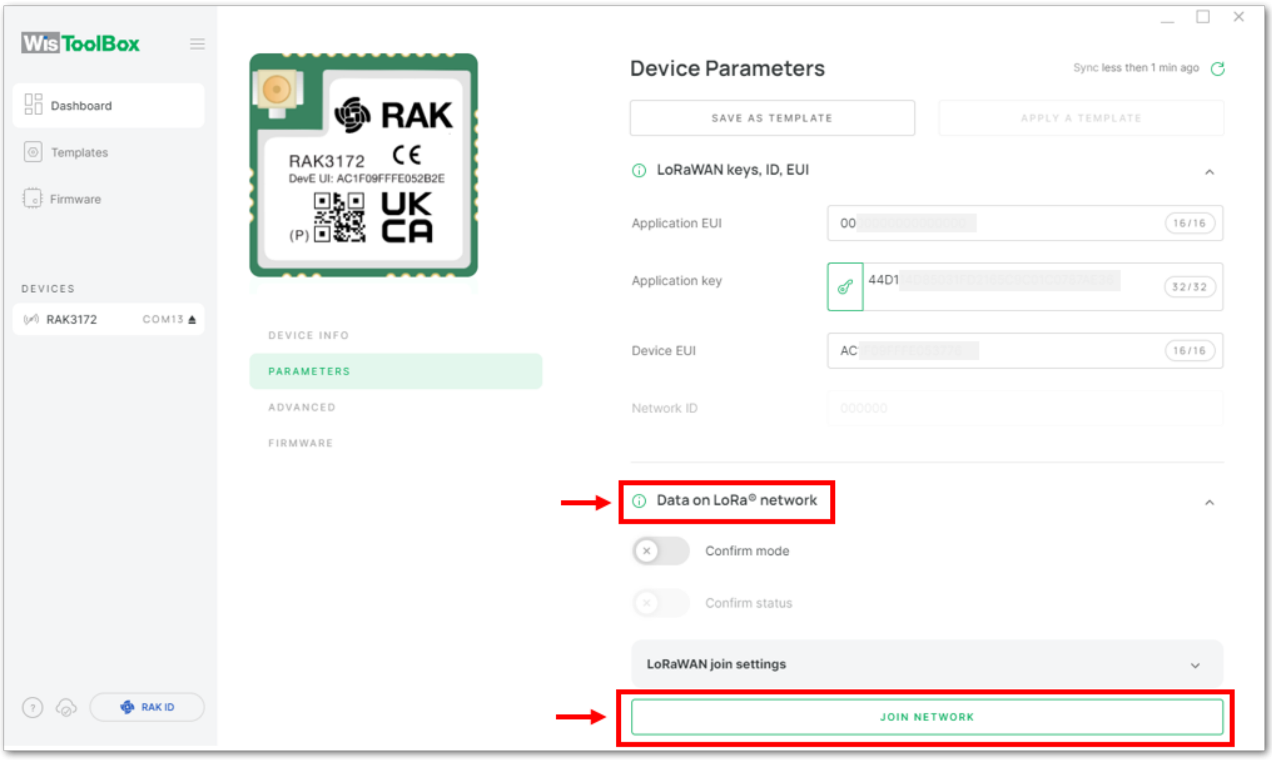
Application EUI: 00[redacted] 16/16

Application key: 44D1[redacted] 32/32

Device EUI: AC[redacted] 16/16

Figure 77: Successfully configured OTAA device via WisToolBox dashboard

11. After your device's credentials are updated, it can now join the network. To do this, you need to go to **Data on LoRa network** under **PARAMETERS**, and then click **JOIN NETWORK** under **LoRaWAN join settings**. After a few seconds, it will notify you that your OTAA device has already joined the TTN server. You can also access your TTN console if your device has successfully joined the TTN.



Device Parameters

Sync less then 1 min ago

LoRaWAN keys, ID, EUI

Application EUI: 00[redacted] 16/16

Application key: 44D1[redacted] 32/32

Device EUI: AC[redacted] 16/16

Network ID: 000000

Data on LoRa® network

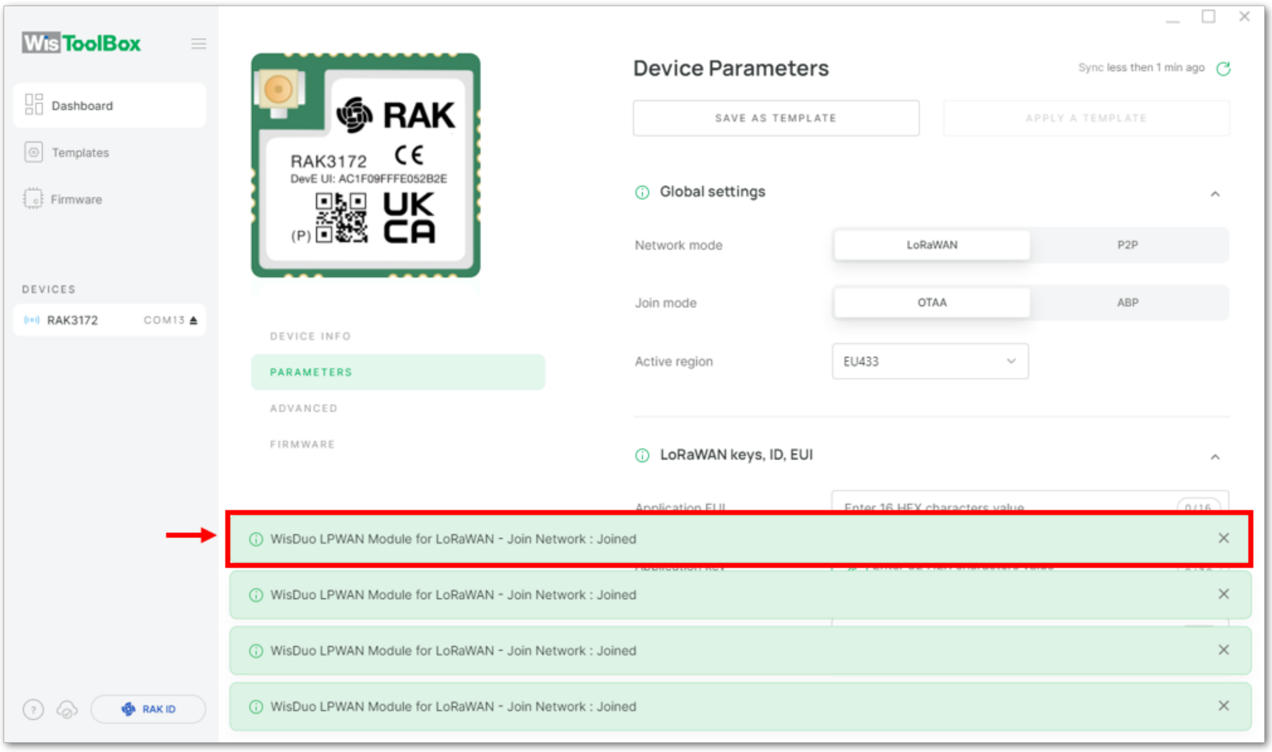
Confirm mode

Confirm status

LoRaWAN join settings

JOIN NETWORK

Figure 78: Joining mode of your OTAA device



Device Parameters

Global settings

Network mode: LoRaWAN | P2P

Join mode: OTAA | ABP

Active region: EU433

LoRaWAN keys, ID, EUI

Application EUI: Enter 16 HEX characters value 0/16

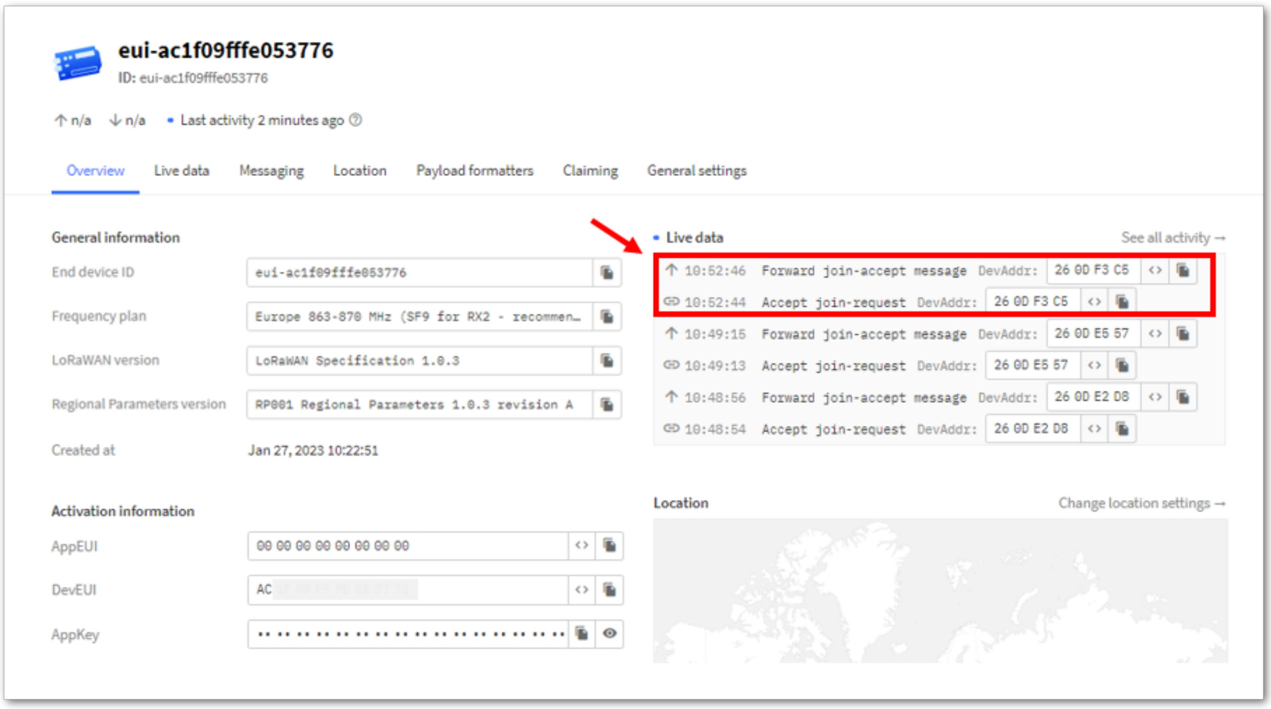
WisDuo LPWAN Module for LoRaWAN - Join Network : Joined

WisDuo LPWAN Module for LoRaWAN - Join Network : Joined

WisDuo LPWAN Module for LoRaWAN - Join Network : Joined

WisDuo LPWAN Module for LoRaWAN - Join Network : Joined

Figure 79: OTAA device successfully joined the TTN server



eui-ac1f09fffe053776

ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • Last activity 2 minutes ago

General information

End device ID: eui-ac1f09fffe053776

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 27, 2023 10:22:51

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: AC [redacted]

AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Live data

```
↑ 10:52:46  Forward join-accept message  DevAddr: 26 0D F3 C5
  10:52:44  Accept join-request  DevAddr: 26 0D F3 C5
↑ 10:49:15  Forward join-accept message  DevAddr: 26 0D E5 57
  10:49:13  Accept join-request  DevAddr: 26 0D E5 57
↑ 10:48:56  Forward join-accept message  DevAddr: 26 0D E2 D8
  10:48:54  Accept join-request  DevAddr: 26 0D E2 D8
```

Location

Figure 80: OTAA device successfully joined the TTN server

OTAA Configuration for TTN via WisToolBox Console

Here's another way of OTAA configuration using **WisToolBox Console**. Below are the steps on setting up your **RAK3172** using **WisToolBox Console**.

1. Connect your **RAK3172** with your chosen WisBlock base board to the PC via USB cable and open the **WisToolBox** application.

2. Click **CONNECT DEVICE** button to launch the WisToolBox Dashboard.

   CONNECT DEVICE

   Figure 81: CONNECT DEVICE

3. Select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT**, as shown in **Figure 83**.

Setting up your device

Figure 82: Setting up your device

Setting up your device

Figure 83: Setting up your device

4. Once done, **RAK3172** will appear in the dashboard, then select it.

Device seen from WisToolBox dashboard

Figure 84: Device seen from WisToolBox dashboard

5. Then click **ADVANCED**.

Setting up your device

Figure 85: Setting up your device

6. Once done, click **OPEN CONSOLE** to do the configuration.

OPEN CONSOLE

Figure 86: OPEN CONSOLE

Opening the Console terminal of WisToolBox

Figure 87: Opening the Console terminal of WisToolBox

Opening the Console terminal of WisToolBox

Figure 88: Opening the Console terminal of WisToolBox

7. To start the configuration, type `ATE` so you can echo the commands you input during your configuration. Then press **Enter**.

It is recommended to start by testing the console and verify that the current configuration is working by sending these two AT commands:

```
AT
```

```
ATE
```

`ATE` is useful for tracking the commands and troubleshooting.

You will receive `OK` when you input the two commands. After setting `ATE`, you can now see all the commands you input together with the replies.

> 📝 **NOTE:**
>
> If there is no `OK` or any reply, check if the device is powered correctly. If you are getting power from a USB port, ensure that you have a good USB cable.

Setting up your Console

Figure 89: Setting up your Console

Setting up your Console

Figure 90: Setting up your Console

Setting up your Console

Figure 91: Setting up your Console

8. Then configure the LoRaWAN join mode to **OTAA**. You can check what parameter you will input by typing **AT+NJM?** and then **Enter** into the console terminal. For **OTAA**, you should input **AT+NJM=1** and then press **Enter**, as shown in **Figure 92**.

   Setting up your Console

   Figure 92: Setting up your Console

   Setting up your Console

   Figure 93: Setting up your Console

   Setting up your Console

   Figure 94: Setting up your Console

9. Once done, set your LoRaWAN region to EU868. You can check what parameter you will input by typing `AT+BAND?` then **Enter** into the console terminal. For **EU868**, you should input **AT+BAND=4** then press **Enter**. If you wish to work on other regional bands, you may check the list of band parameter options below.

Set the frequency/region to EU868.

```
AT+BAND=4
```

📝 **NOTE:**

Depending on the Regional Band you selected, you might need to configure the sub-band of your RAK3172 to match the gateway and LoRaWAN network server. This is especially important for regional bands like US915, AU915, and CN470.

To configure the masking of channels for the sub-bands, you can use the `AT+MASK` command that can be found on the AT Command Manual .

To illustrate, you can use sub-band 2 by sending the command `AT+MASK=0002`.

List of band parameter options

Code	Regional Band
0	EU433
1	CN470
2	RU864
3	IN865
4	EU868
5	US915
6	AU915
7	KR920
8	AS923-1
9	AS923-2
10	AS923-3
11	AS923-4

Setting up your Console

Figure 95: Setting up your Console

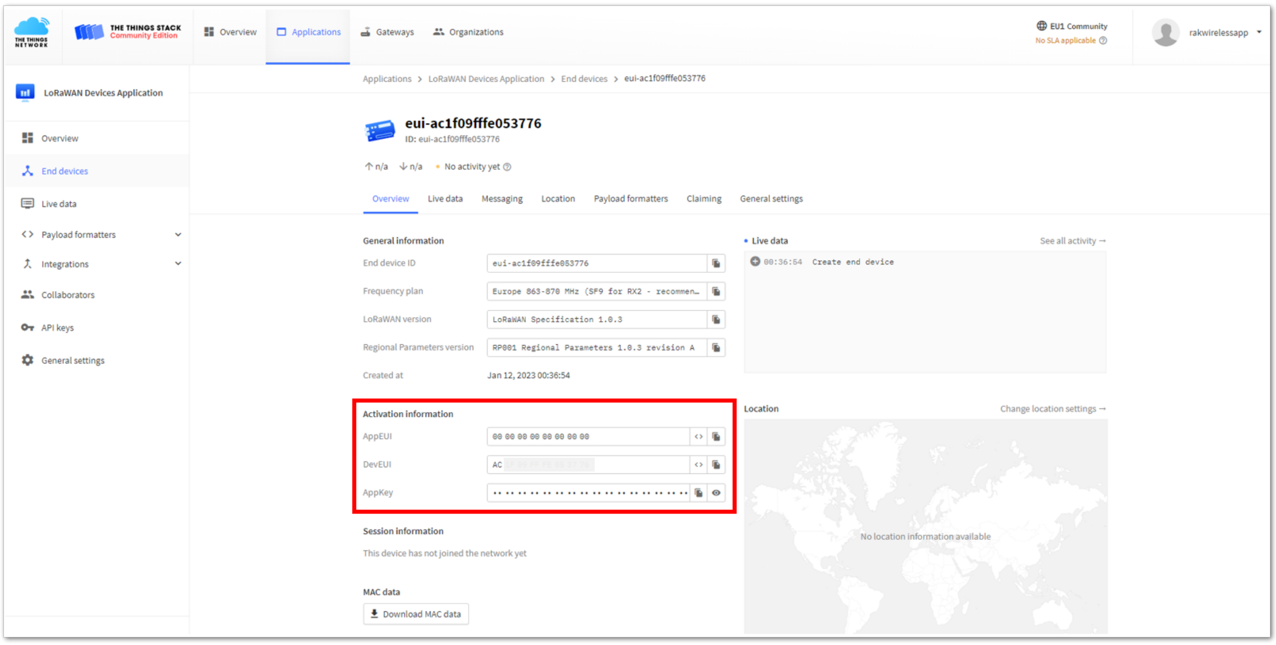
Setting up your Console

Figure 96: Setting up your Console

Setting up your Console

Figure 97: Setting up your Console

10. Then next to this will be updating the OTAA credentials of your device. First on this list will be the **Application EUI (AppEUI)**. Go back to your console where your RAK3172 End device was created to copy the AppEUI credential, then paste it to the WisToolBox Console and press **Enter**.



eui-ac1f09fffe053776

ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

General information

End device ID: eui-ac1f09fffe053776

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 12, 2023 00:36:54

Live data

00:36:54 Create end device

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: AC [redacted]

AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Location

Session information

This device has not joined the network yet

MAC data

Figure 98: Your created OTAA device from your TTN console

Setting up your Console

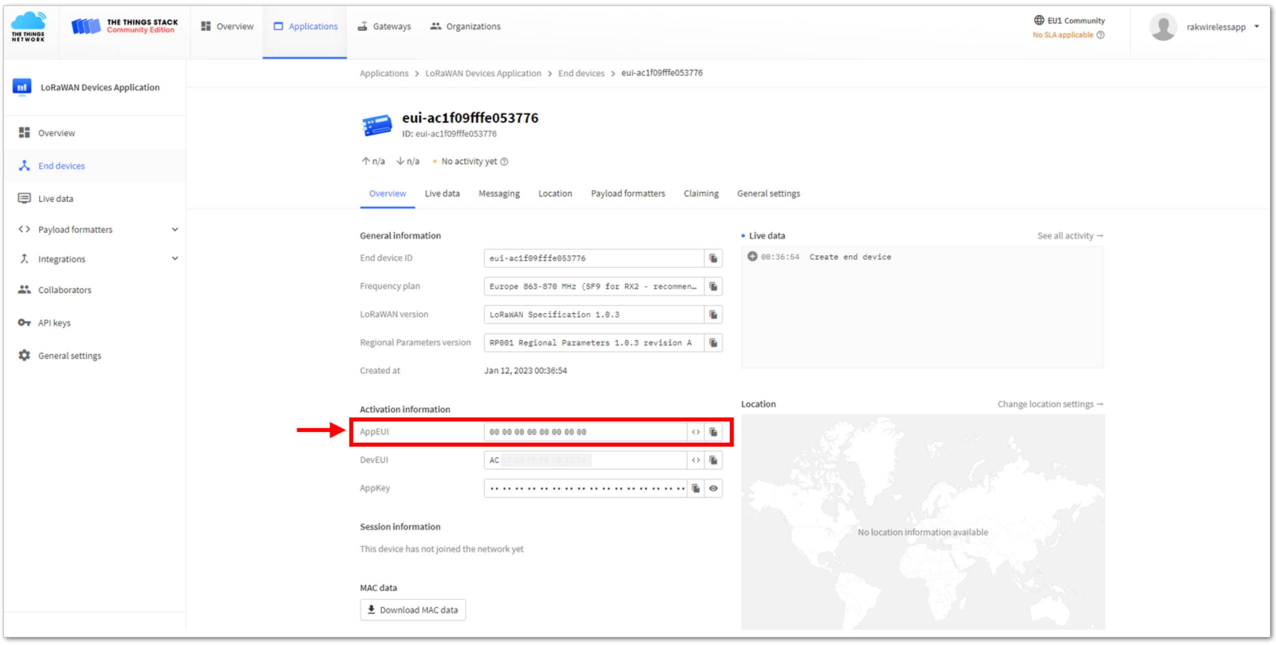
Figure 99: Setting up your Console

Setting up your Console

Figure 100: Setting up your Console

Setting up your Console

Figure 101: Setting up your Console



eui-ac1f09fffe053776

ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

General information

End device ID: eui-ac1f09fffe053776

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 12, 2023 00:36:54

Live data

00:36:54 Create end device

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: AC [redacted]

AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Location

Session information

This device has not joined the network yet

MAC data

Figure 102: Copying the AppEUI credential from TTN to WisToolBox

Setting up your Console

Figure 103: Setting up your Console

11. Once done, do the same procedure to **Application key (AppKey)** and **Device EUI (DevEUI)**.

- **For Application key (AppKey)**

Setting up your Console

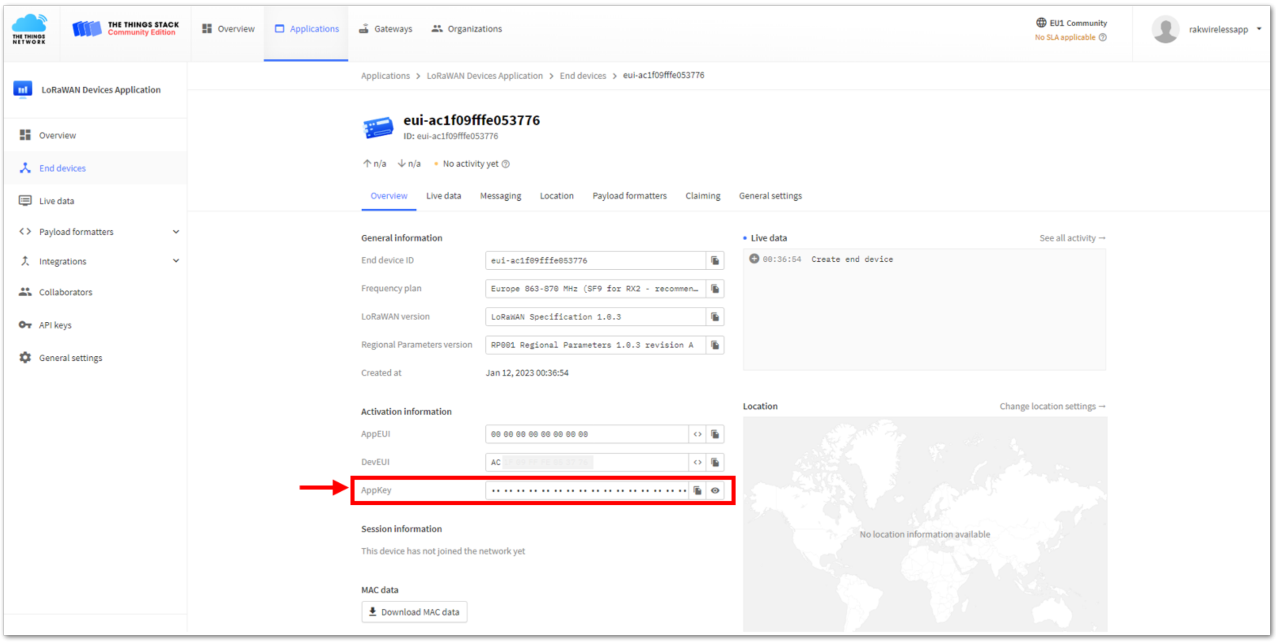
Figure 104: Setting up your Console

Setting up your Console

Figure 105: Setting up your Console

Setting up your Console

Figure 106: Setting up your Console



THE THINGS STACK Community Edition | Overview | Applications | Gateways | Organizations | EU1 Community No SLA applicable | rakwirelessapp

Applications > LoRaWAN Devices Application > End devices > eui-ac1f09fffe053776

LoRaWAN Devices Application: Overview, End devices, Live data, Payload formatters, Integrations, Collaborators, API keys, General settings

eui-ac1f09fffe053776
ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

Overview | Live data | Messaging | Location | Payload formatters | Claiming | General settings

General information

End device ID	eui-ac1f09fffe053776
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 12, 2023 00:36:54

Live data — See all activity →

00:36:54 Create end device

Activation information

AppEUI	00 00 00 00 00 00 00 00
DevEUI	AC [redacted]
AppKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Location — Change location settings →

No location information available

Session information

This device has not joined the network yet

MAC data

Download MAC data

Figure 107: Copying the AppKey credential from TTN to WisToolBox

Setting up your Console

Figure 108: Setting up your Console

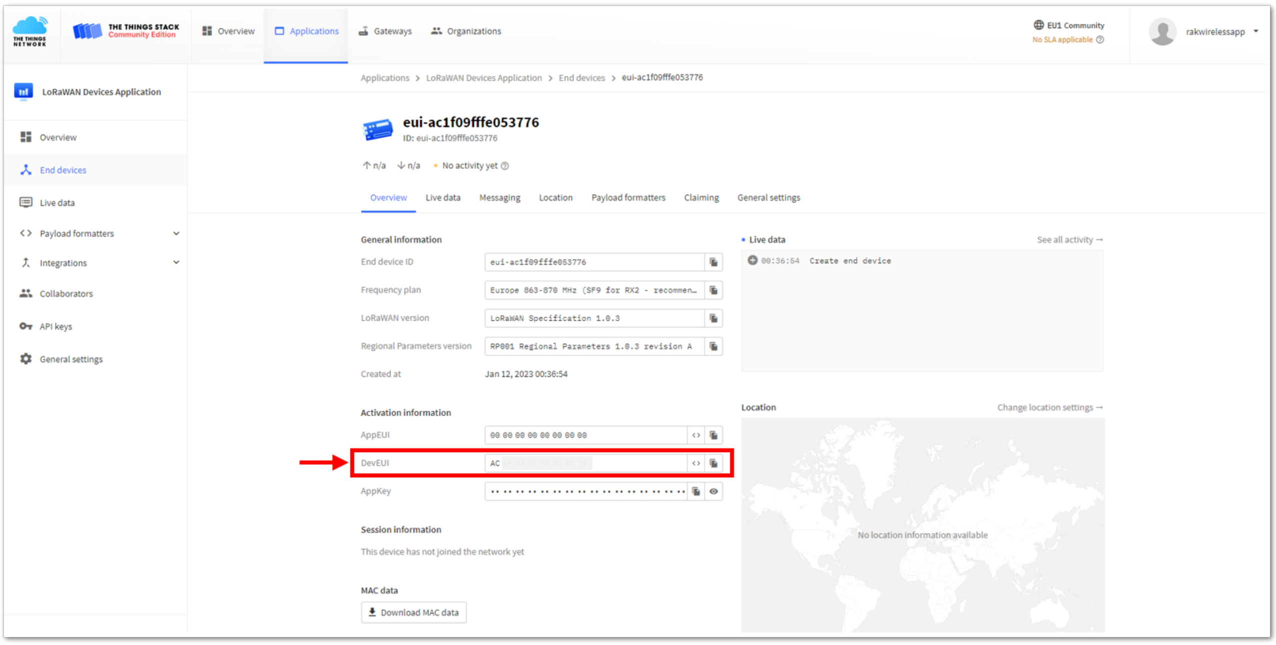
- **For Device EUI (DevEUI)**

Setting up your Console

Figure 109: Setting up your Console

Setting up your Console

Figure 110: Setting up your Console



THE THINGS STACK Community Edition | Overview | Applications | Gateways | Organizations | EU1 Community No SLA applicable | rakwirelessapp

Applications > LoRaWAN Devices Application > End devices > eui-ac1f09fffe053776

LoRaWAN Devices Application: Overview, End devices, Live data, Payload formatters, Integrations, Collaborators, API keys, General settings

eui-ac1f09fffe053776
ID: eui-ac1f09fffe053776

↑ n/a ↓ n/a • No activity yet

Overview | Live data | Messaging | Location | Payload formatters | Claiming | General settings

General information

End device ID	eui-ac1f09fffe053776
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 12, 2023 00:36:54

Live data — See all activity →

00:36:54 Create end device

Activation information

AppEUI	00 00 00 00 00 00 00 00
DevEUI	AC [redacted]
AppKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Location — Change location settings →

No location information available

Session information

This device has not joined the network yet

MAC data

Download MAC data

Figure 111: Copying the DevEUI credential from TTN to WisToolBox

Setting up your Console

Figure 112: Setting up your Console

12. Once done, click **Dashboard** to check the updated credentials of your OTAA device. Click **PARAMETERS** to open the **Global Settings** and **LoRaWAN keys, ID, EUI**, and check whether these portions are updated.

Setting up your Console

Figure 113: Setting up your Console

Setting up your Console

Figure 114: Setting up your Console

PARAMETERS

Figure 115: PARAMETERS

Global settings and LoRaWAN keys, ID, EUI

Figure 116: Global settings and LoRaWAN keys, ID, EUI

Global settings and LoRaWAN keys, ID, EUI details

Figure 117: Global settings and LoRaWAN keys, ID, EUI details

13. Now you have a configured OTAA device using WisToolBox Console. You can now join the network using the WisToolBox console.

14. To do this, you need to go back to the WisToolBox console and type **AT+JOIN**. Edit it to **AT+JOIN=1** and press **Enter** to join the network.

> 📝 **NOTE:**
>
> `AT+JOIN` command parameters are optional. You can configure the settings for auto-join, reattempt interval, and the number of join attempts if your application needs it. If not configured, it will use the default parameter values.
>
> `AT+JOIN` and `AT+JOIN=1` also share the common functionality of trying to join the network.

Join command format: `AT+JOIN=w:x:y:z`

Parameter	Description
w	Join command - 1: joining, 0: stop joining.
x	Auto-join config - 1: auto-join on power-up, 0: no auto-join
y	Reattempt interval in seconds (7-255) - 8 is the default.
z	Number of join attempts (0-255) - 0 is default.

After 5 or 6 seconds, if the request is successfully received by a LoRa gateway, you should see `+EVT:JOINED` status reply, as shown in the figure below:

> 📝 **NOTE:**
>
> If the OTAA device failed to join, you need to check if your device is within reach of a working LoRaWAN gateway that is configured to connect to TTN. It is also important to check that all your OTAA parameters (DEVEUI, APPEUI, and APPKEY) are correct using the `AT+DEVEUI=?` , `AT+APPEUI=?` , and `AT+APPKEY=?` commands. Lastly, ensure that the antenna of your device is properly connected.
>
> After checking all the things above, try to join again.

Joining mode using WisToolBox Console

Figure 118: Joining mode using WisToolBox Console

Joining mode using WisToolBox Console

Figure 119: Joining mode using WisToolBox Console

Joining mode using WisToolBox Console

Figure 120: Joining mode using WisToolBox Console

Joining mode using WisToolBox Console

Figure 121: Joining mode using WisToolBox Console

OTAA device successfully joined the network

Figure 122: OTAA device successfully joined the network

OTAA device successfully joined the network

Figure 123: OTAA device successfully joined the network

15. With the end-device properly joined to the TTN, you can now try to send some payload after a successful join. Send command format: **`AT+SEND=<port>:<payload>`**

```
AT+SEND=2:12345678
```

OTAA device sending payload to the network

Figure 124: OTAA device sending payload to the network

OTAA device sending payload to the network

Figure 125: OTAA device sending payload to the network

OTAA device sending payload to the network

Figure 126: OTAA device sending payload to the network

OTAA device sending payload to the network

Figure 127: OTAA device sending payload to the network

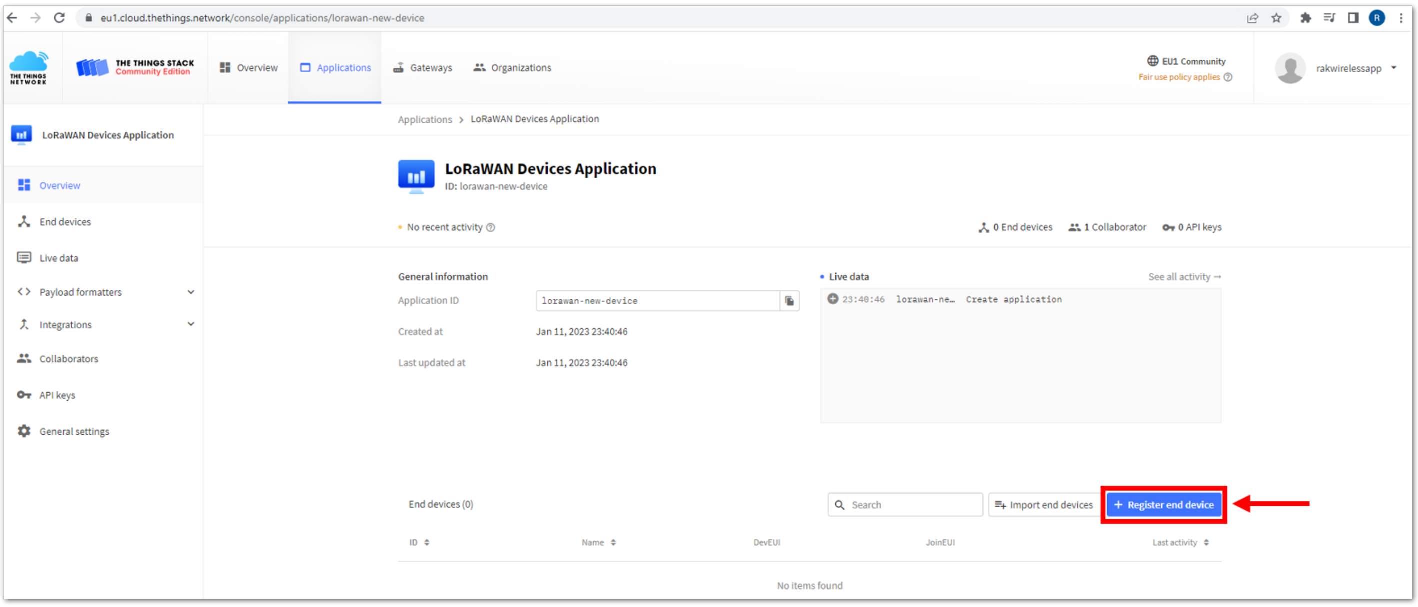
16. You can see the data sent by the RAK3172 module on the TTN device console *Live data* section. Also, the *Last seen* info should be a few seconds or minutes ago.

OTAA Test Sample Data Sent Viewed in TTN

Figure 128: OTAA Test Sample Data Sent Viewed in TTN

TTN ABP Device Registration

1. To register an ABP device, go to your application console and select the application to which you want your device to be added. Then click **+ Register end device**, as shown in **Figure 129**.



LoRaWAN Devices Application
ID: lorawan-new-device

General information

Application ID: lorawan-new-device

Created at: Jan 11, 2023 23:40:46

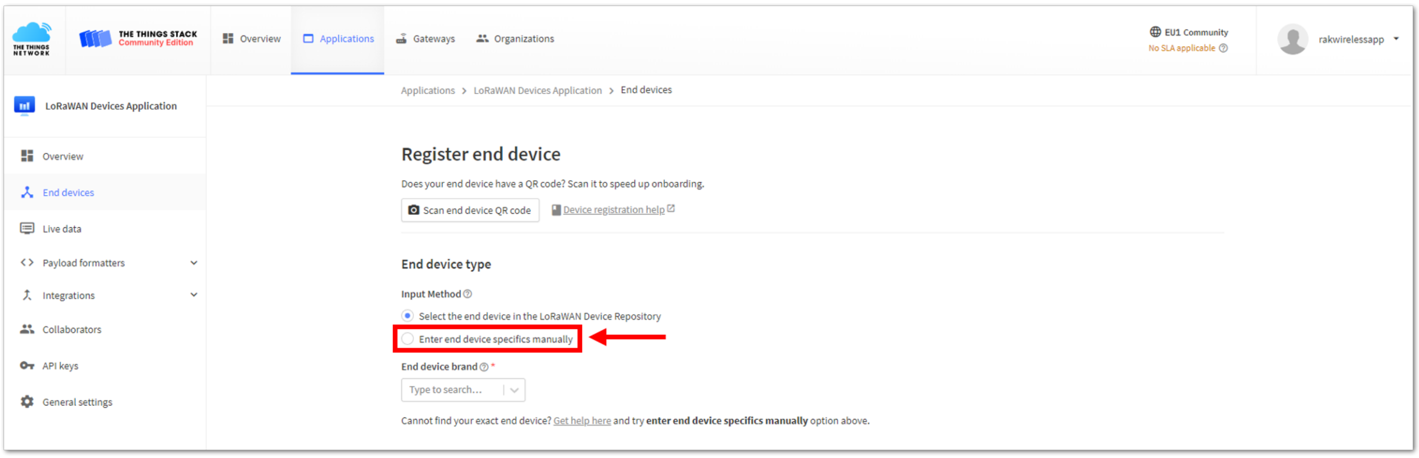
Last updated at: Jan 11, 2023 23:40:46

```
23:40:46  lorawan-ne…  Create application
```

End devices (0)

Figure 129: Adding ABP Device

2. To register the board, click the **Enter end device specifics manually**.



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

End device type

Input Method

● Select the end device in the LoRaWAN Device Repository

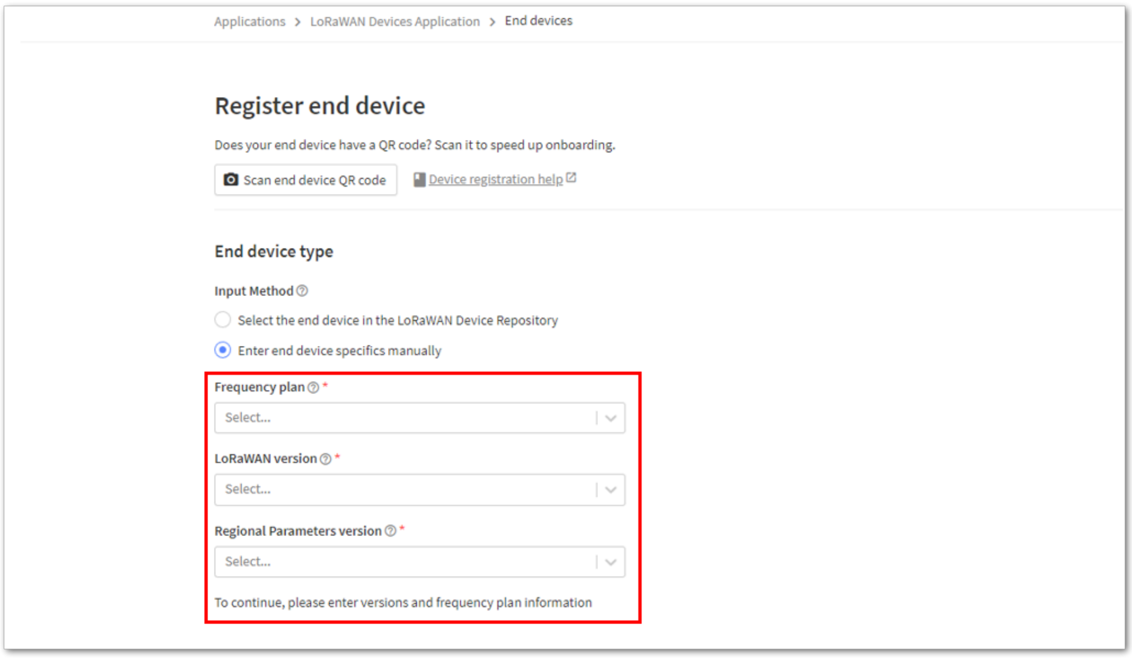
○ Enter end device specifics manually

End device brand *

Cannot find your exact end device? Get help here and try **enter end device specifics manually** option above.

Figure 130: Enter end device specifics manually

3. Next step is to set up **Frequency plan**, compatible **LoRaWAN version**, and **Regional Parameters version** supported.



Applications > LoRaWAN Devices Application > End devices

Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *

Select...

LoRaWAN version *

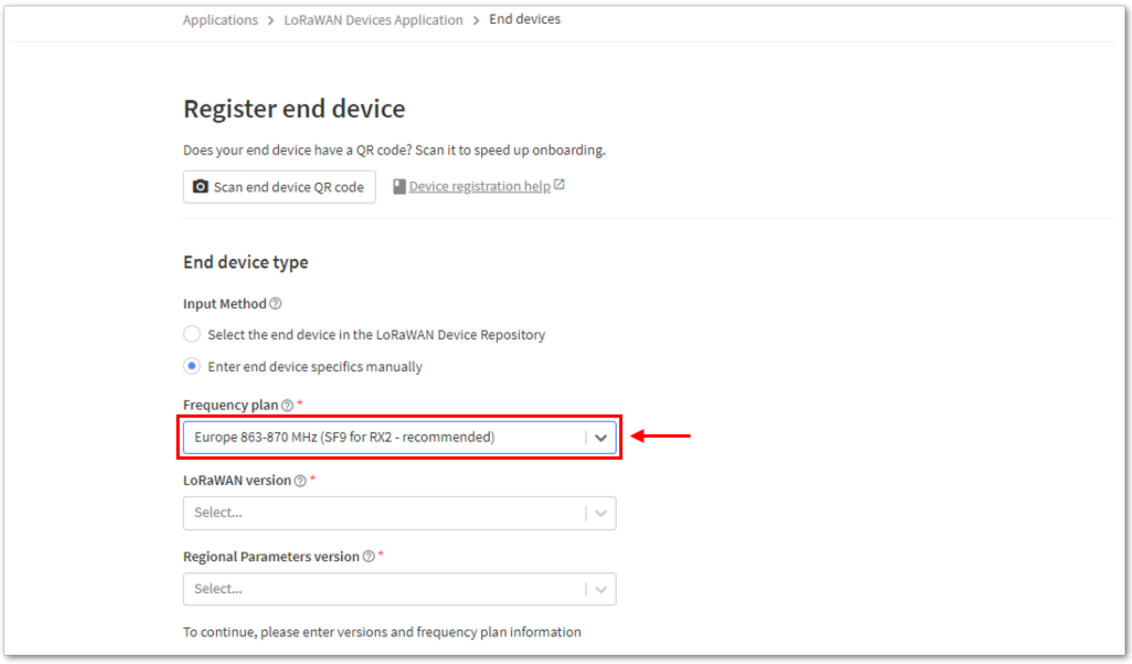
Select...

Regional Parameters version *

Select...

To continue, please enter versions and frequency plan information

Figure 131: Setting up for your device



Applications > LoRaWAN Devices Application > End devices

Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *

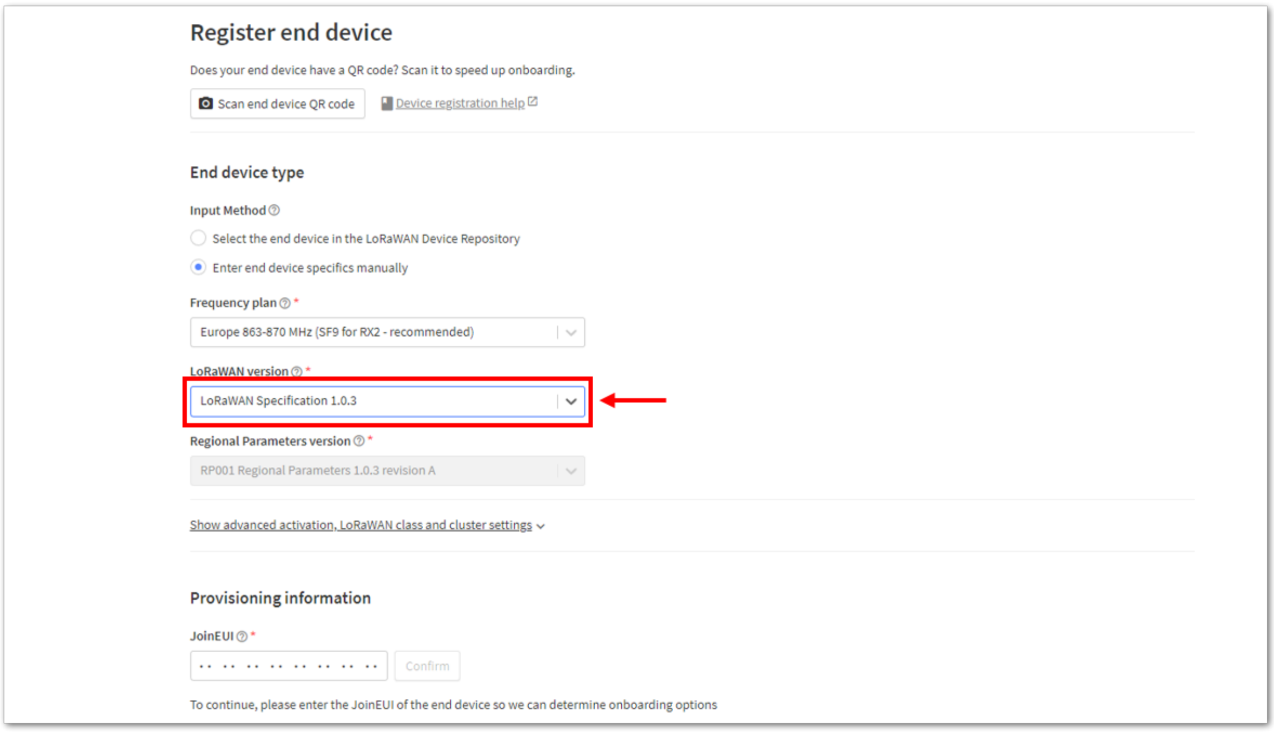
Select...

Regional Parameters version *

Select...

To continue, please enter versions and frequency plan information

Figure 132: Setting up for your device



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *

Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *

LoRaWAN Specification 1.0.3

Regional Parameters version *

RP001 Regional Parameters 1.0.3 revision A

Show advanced activation, LoRaWAN class and cluster settings

Provisioning information

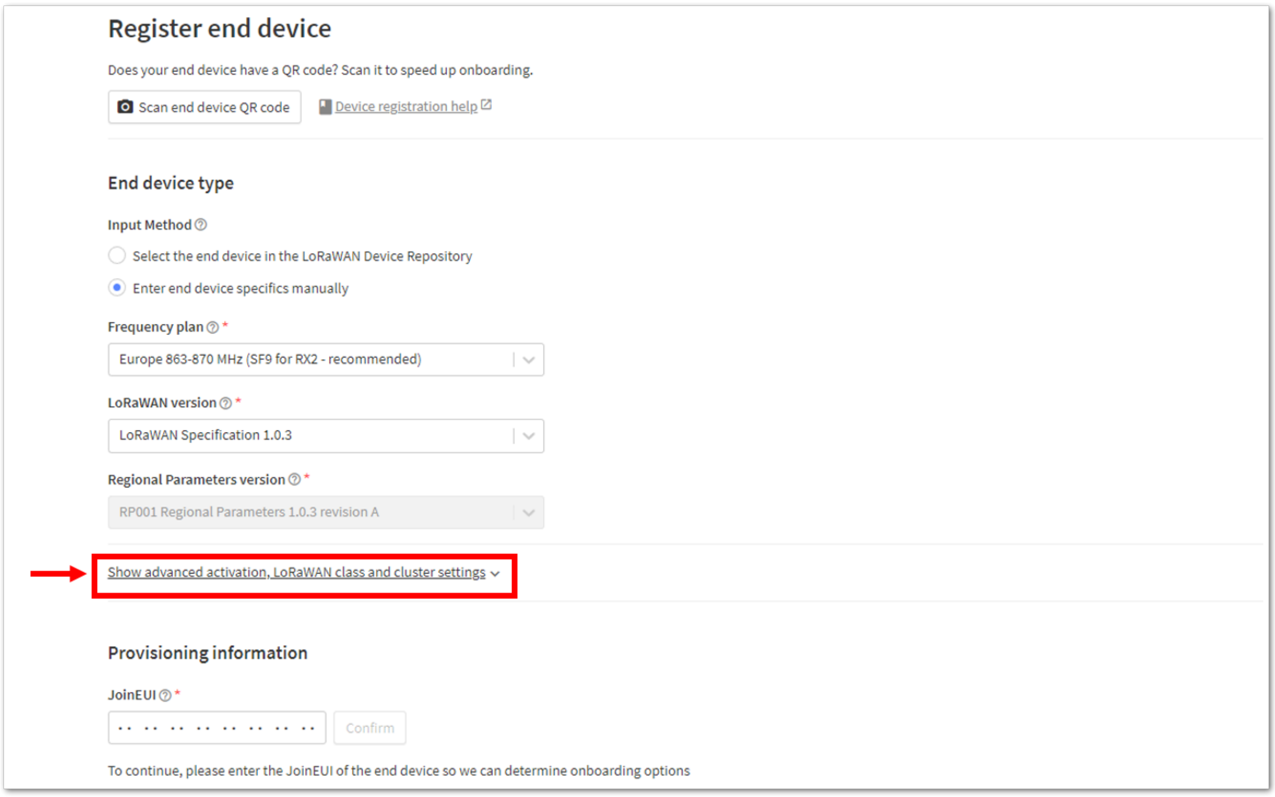
JoinEUI *

.. .. | Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 133: Setting up for your device

4. Then click **Show advanced activation, LoRaWAN class and cluster settings**. Configure the activation mode by selecting **Activation by personalization (ABP)** and Additional LoRaWAN class capabilities to **class A only**.



Register end device

Does your end device have a QR code? Scan it to speed up onboarding.

Scan end device QR code | Device registration help

End device type

Input Method

○ Select the end device in the LoRaWAN Device Repository

● Enter end device specifics manually

Frequency plan *: Europe 863-870 MHz (SF9 for RX2 - recommended)

LoRaWAN version *: LoRaWAN Specification 1.0.3

Regional Parameters version *: RP001 Regional Parameters 1.0.3 revision A

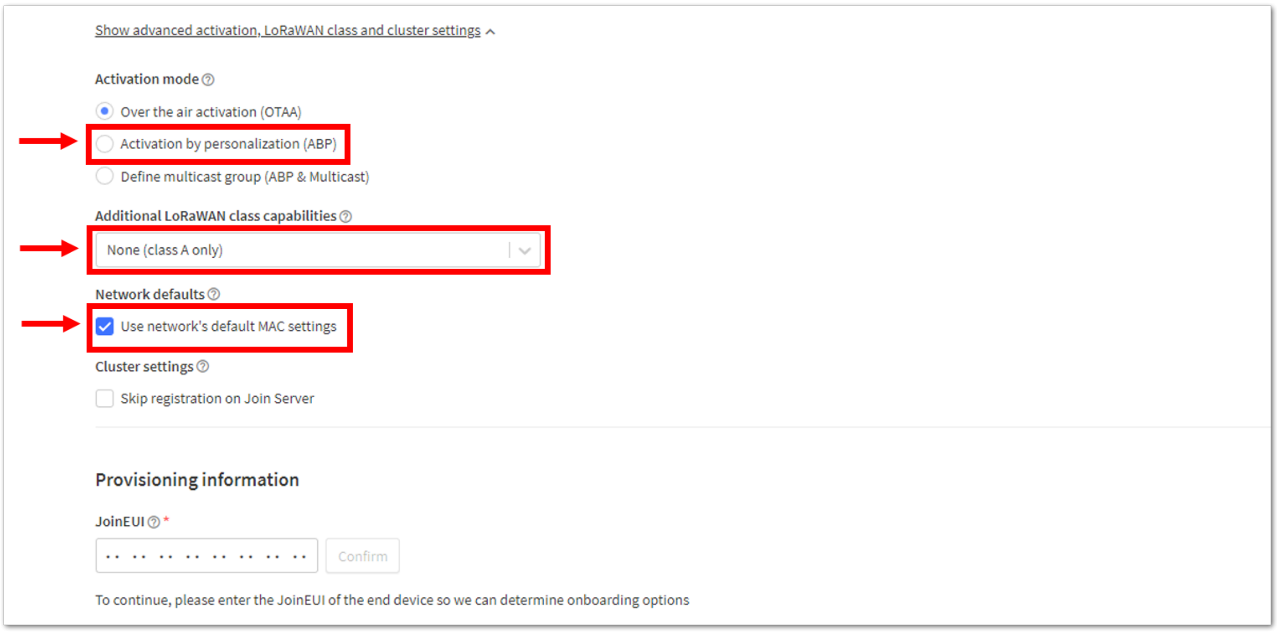
Show advanced activation, LoRaWAN class and cluster settings

Provisioning information

JoinEUI *: .. .. Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 134: Setting up for your device



Show advanced activation, LoRaWAN class and cluster settings

Activation mode

● Over the air activation (OTAA)

○ Activation by personalization (ABP)

○ Define multicast group (ABP & Multicast)

Additional LoRaWAN class capabilities: None (class A only)

Network defaults

☑ Use network's default MAC settings

Cluster settings

□ Skip registration on Join Server

Provisioning information

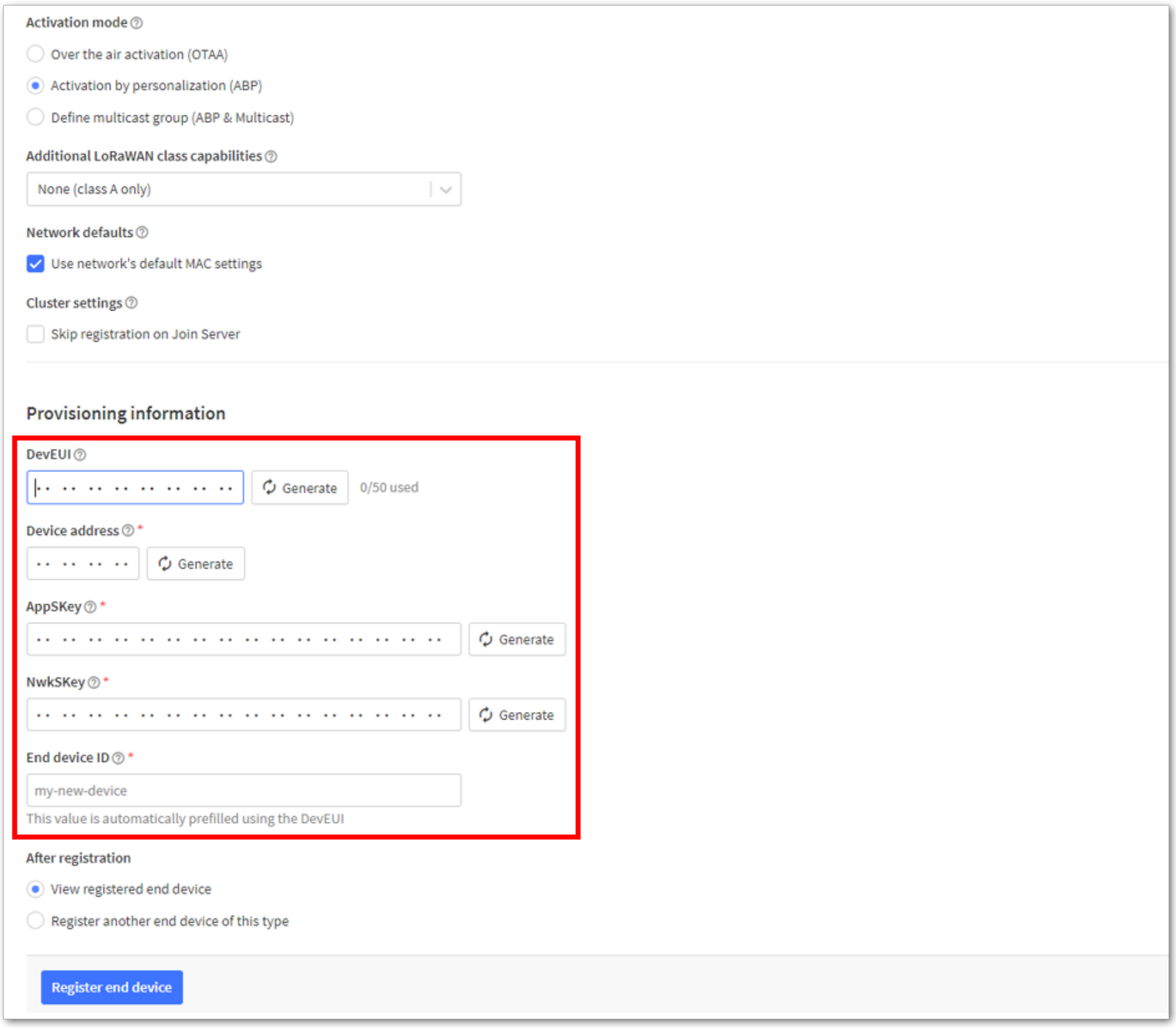
JoinEUI *: .. .. Confirm

To continue, please enter the JoinEUI of the end device so we can determine onboarding options

Figure 135: Setting up for your device

5. Once done, provide the DevEUI credentials of your device into the **DevEUI** portion. This will automatically generate the specific End device ID of your board. Then click **Generate** under **Device address**, **AppSKey**, and **NwkSKey** under the Provisioning information section. Then click **Register end device**.

> 📝 **NOTE:**
>
> - The **DevEUI**, **Device address**, **AppSKey**, and **NwkSKey** are hidden in this section as these are unique from a specific device. The **DevEUI** credential is unique to every RAK3172 device. Also, you should generate your own **Device address**, **AppSKey**, and **NwkSKey** credentials for your specific device and application.



Activation mode ⓘ

○ Over the air activation (OTAA)

● Activation by personalization (ABP)

○ Define multicast group (ABP & Multicast)

Additional LoRaWAN class capabilities ⓘ

None (class A only)

Network defaults ⓘ

☑ Use network's default MAC settings

Cluster settings ⓘ

☐ Skip registration on Join Server

Provisioning information

DevEUI ⓘ

Device address ⓘ *

AppSKey ⓘ *

NwkSKey ⓘ *

End device ID ⓘ *

my-new-device

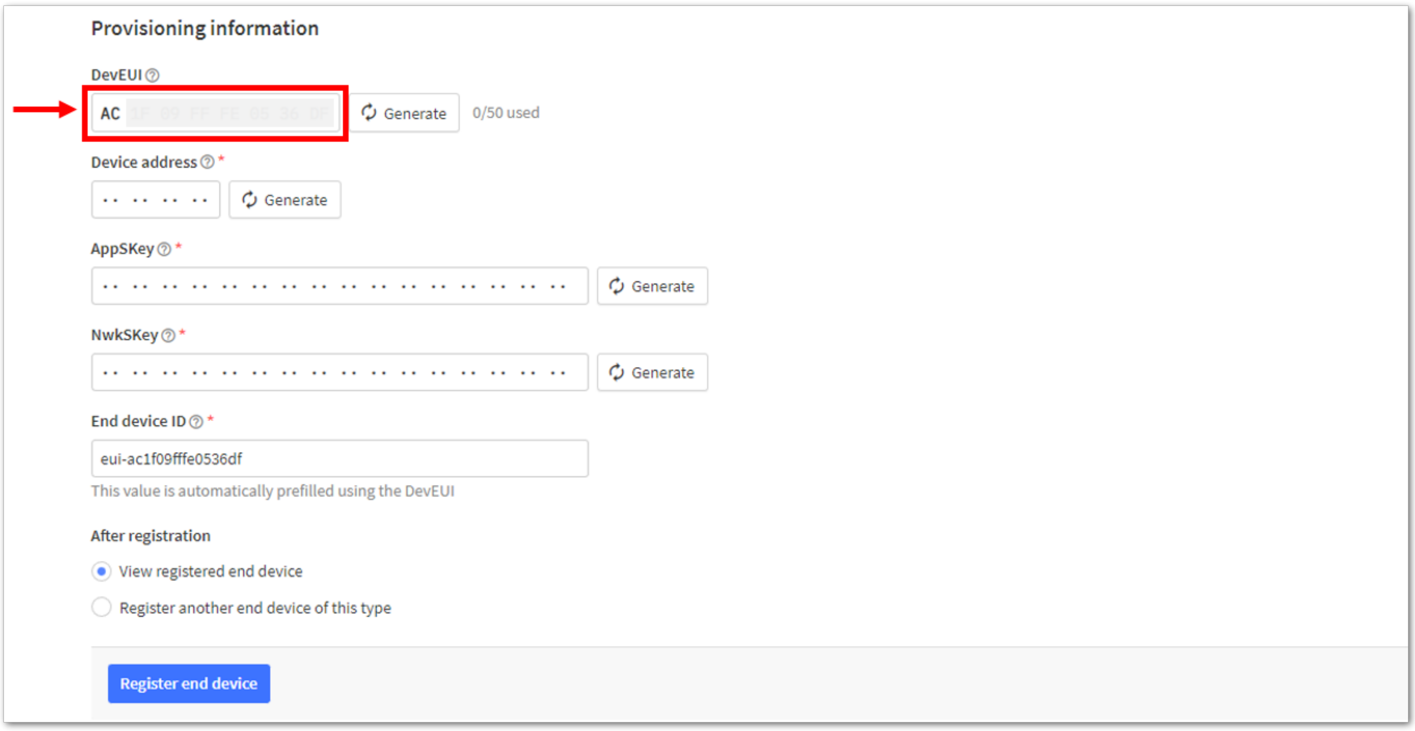
This value is automatically prefilled using the DevEUI

After registration

● View registered end device

○ Register another end device of this type

Figure 136: Setting up for your device



Provisioning information

DevEUI ⓘ

AC [redacted]

Device address ⓘ *

AppSKey ⓘ *

NwkSKey ⓘ *

End device ID ⓘ *

eui-ac1f09fffe0536df

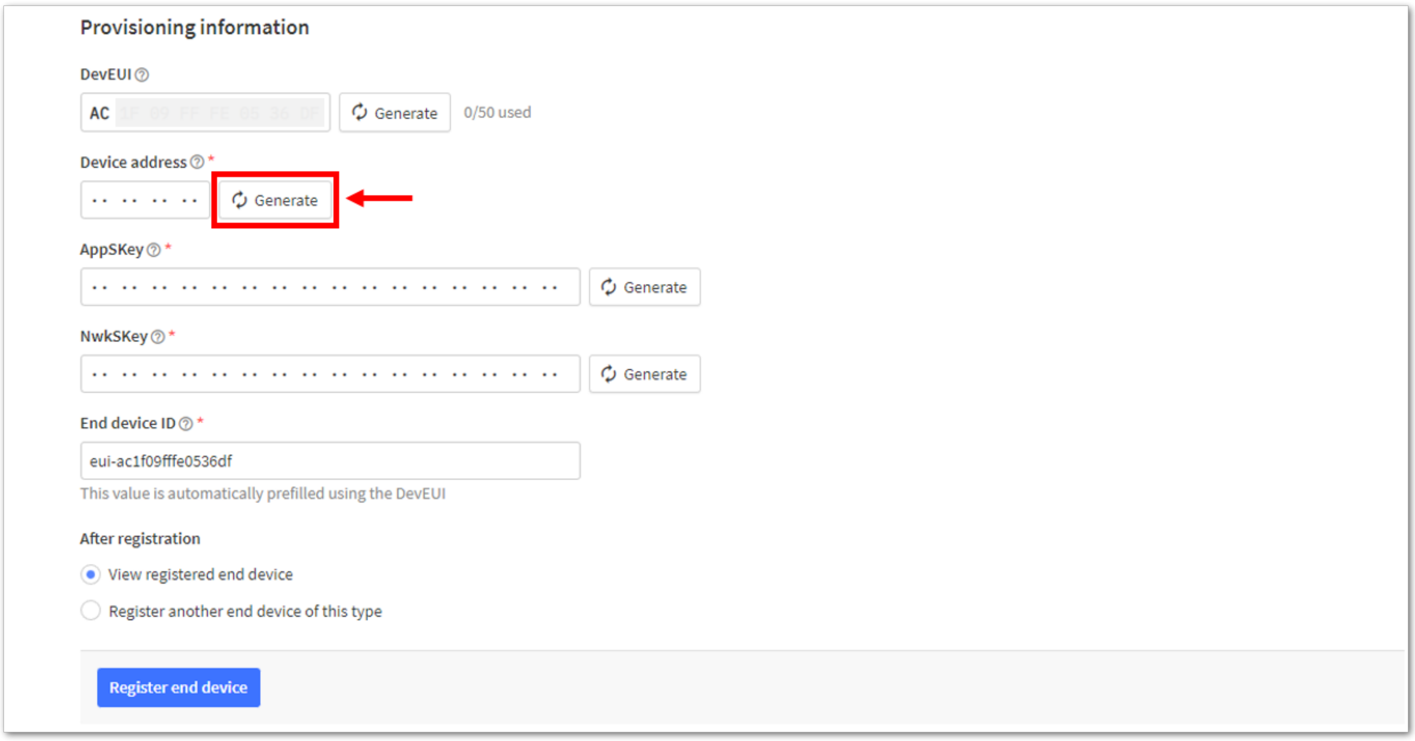
This value is automatically prefilled using the DevEUI

After registration

● View registered end device

○ Register another end device of this type

Figure 137: Setting up for your device



Provisioning information

DevEUI ⓘ
AC 1F 09 FF FE 05 36 DF | Generate | 0/50 used

Device address ⓘ *
•• •• •• •• | Generate

AppSKey ⓘ *
•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• | Generate

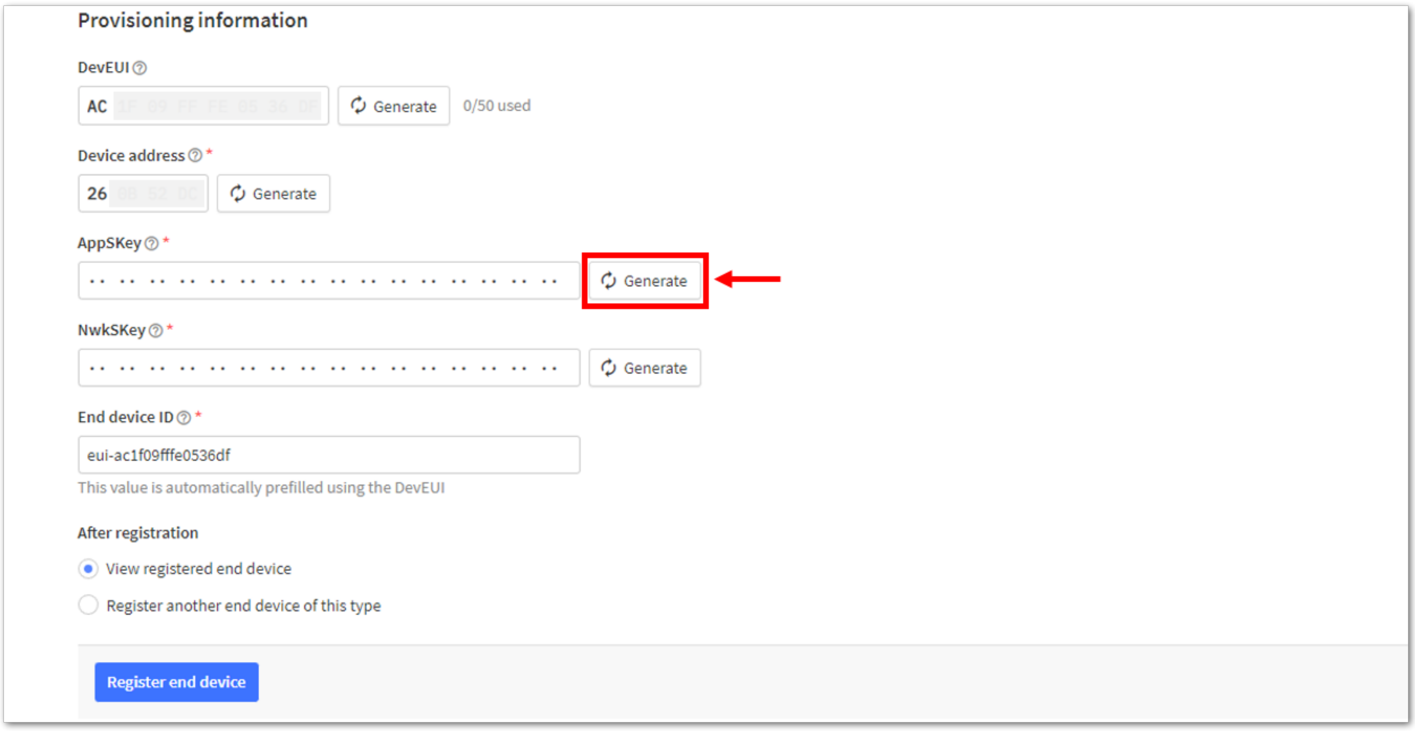
NwkSKey ⓘ *
•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• | Generate

End device ID ⓘ *
eui-ac1f09fffe0536df
This value is automatically prefilled using the DevEUI

After registration
● View registered end device
○ Register another end device of this type

Register end device

Figure 138: Setting up for your device



Provisioning information

DevEUI ⓘ
AC 1F 09 FF FE 05 36 DF | Generate | 0/50 used

Device address ⓘ *
26 0B 52 DC | Generate

AppSKey ⓘ *
•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• | Generate

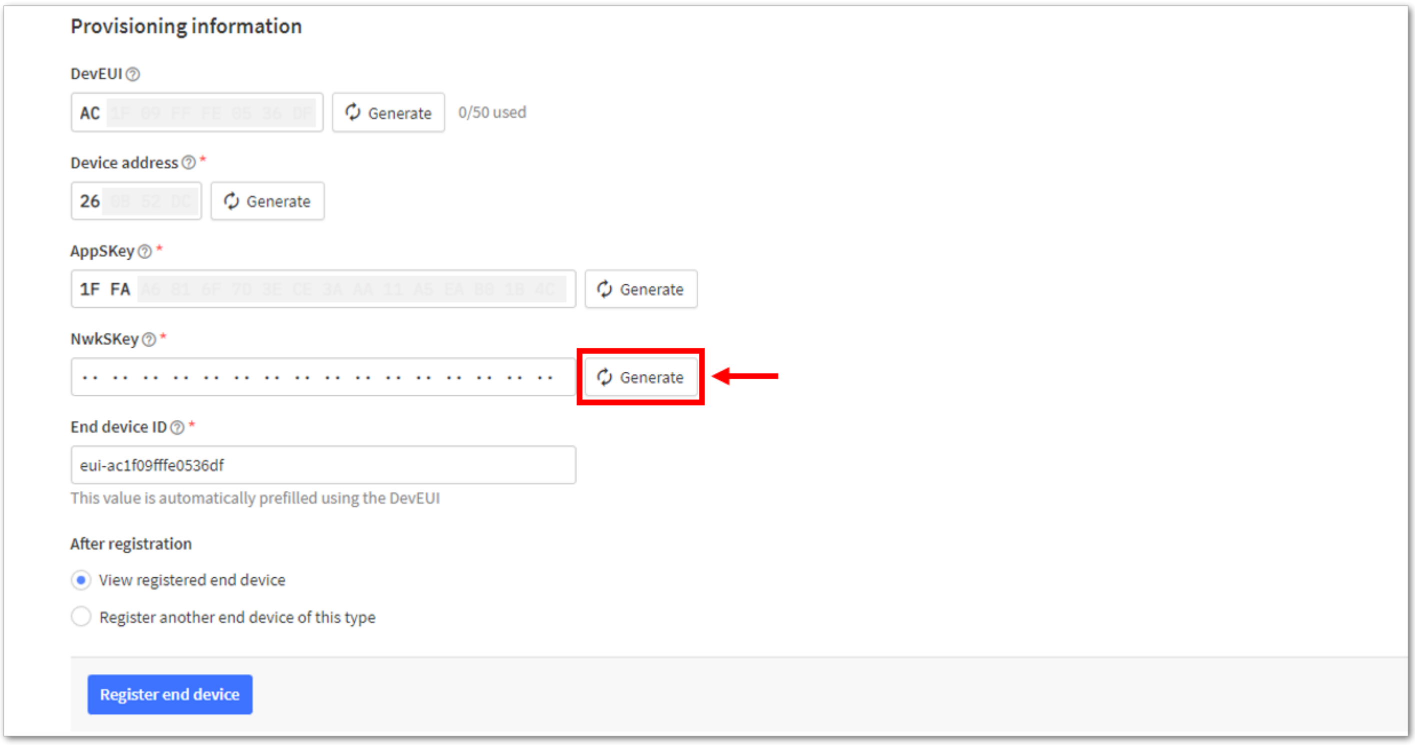
NwkSKey ⓘ *
•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• | Generate

End device ID ⓘ *
eui-ac1f09fffe0536df
This value is automatically prefilled using the DevEUI

After registration
● View registered end device
○ Register another end device of this type

Register end device

Figure 139: Setting up for your device



Provisioning information

DevEUI ⓘ

AC [redacted] | Generate | 0/50 used

Device address ⓘ *

26 [redacted] | Generate

AppSKey ⓘ *

1F FA [redacted] | Generate

NwkSKey ⓘ *

•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• | Generate

End device ID ⓘ *

eui-ac1f09fffe0536df

This value is automatically prefilled using the DevEUI

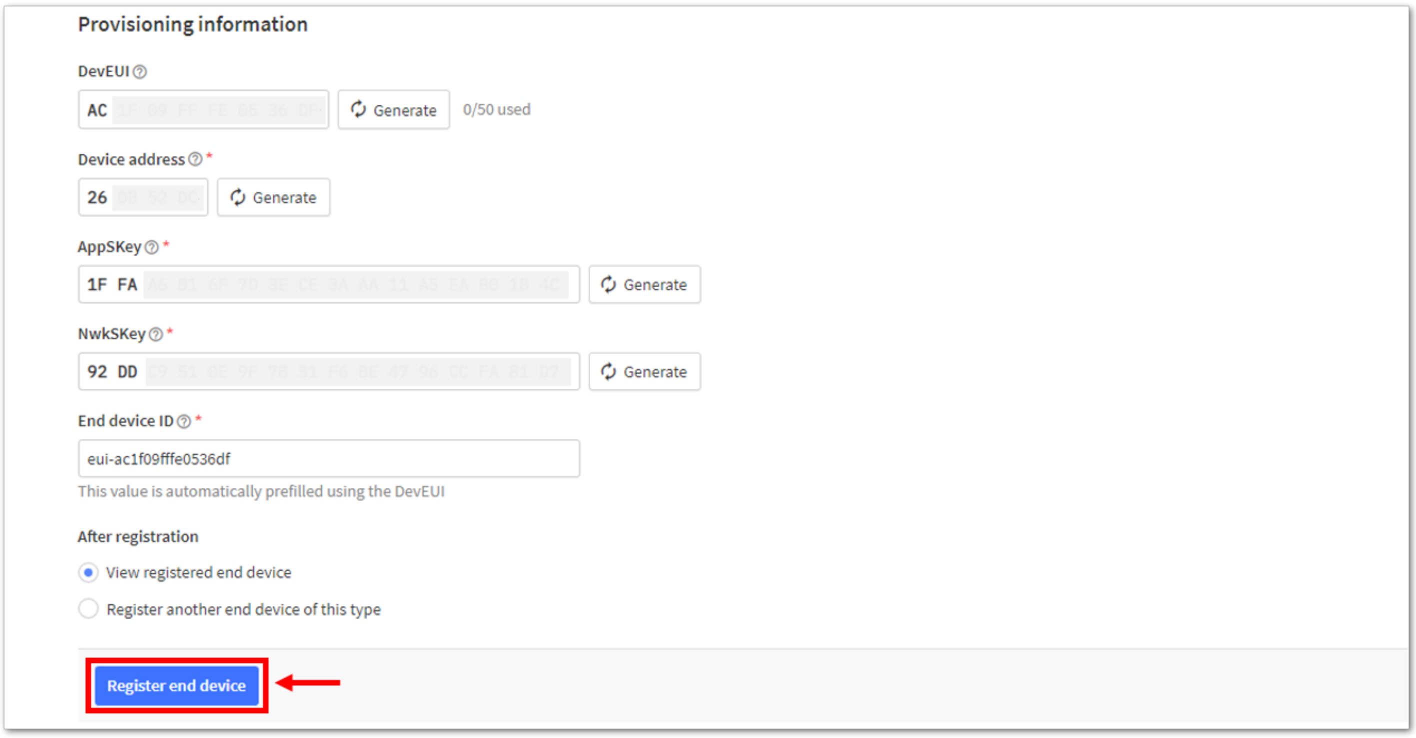
After registration

● View registered end device

○ Register another end device of this type

Register end device

Figure 140: Setting up for your device



Provisioning information

DevEUI ⓘ

AC [redacted] | Generate | 0/50 used

Device address ⓘ *

26 [redacted] | Generate

AppSKey ⓘ *

1F FA [redacted] | Generate

NwkSKey ⓘ *

92 DD [redacted] | Generate

End device ID ⓘ *

eui-ac1f09fffe0536df

This value is automatically prefilled using the DevEUI

After registration

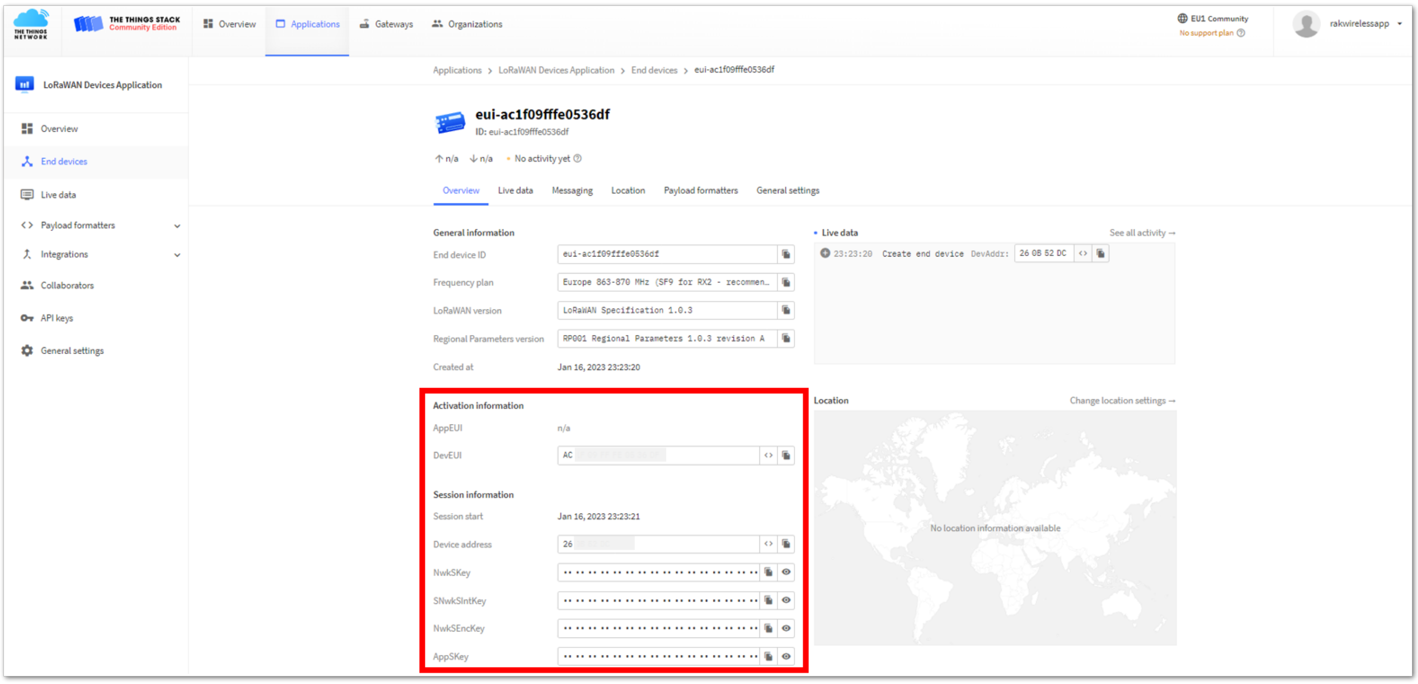
● View registered end device

○ Register another end device of this type

Register end device

Figure 141: Register end device

6. You should now be able to see the device on the TTN console after you fully register your device, as shown in **Figure 142**.



eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df

↑ n/a ↓ n/a • No activity yet

General information

End device ID	eui-ac1f09fffe0536df
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 16, 2023 23:23:20

Live data

```
23:23:20  Create end device  DevAddr:  26 0B 52 DC
```

Activation information

AppEUI	n/a
DevEUI	AC [redacted]

Session information

Session start	Jan 16, 2023 23:23:21
Device address	26 [redacted]
NwkSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Location

Figure 142: ABP device successfully registered to TTN

ABP Configuration for TTN

The RAK3172 module can be configured using WisToolBox to do the ABP configuration. **WisToolBox** is a software tool that supports **RAK3172** module. It automatically detects the RAK3172 module once it is connected to the PC. Below are the options in WisToolBox where the ABP configuration can be done.

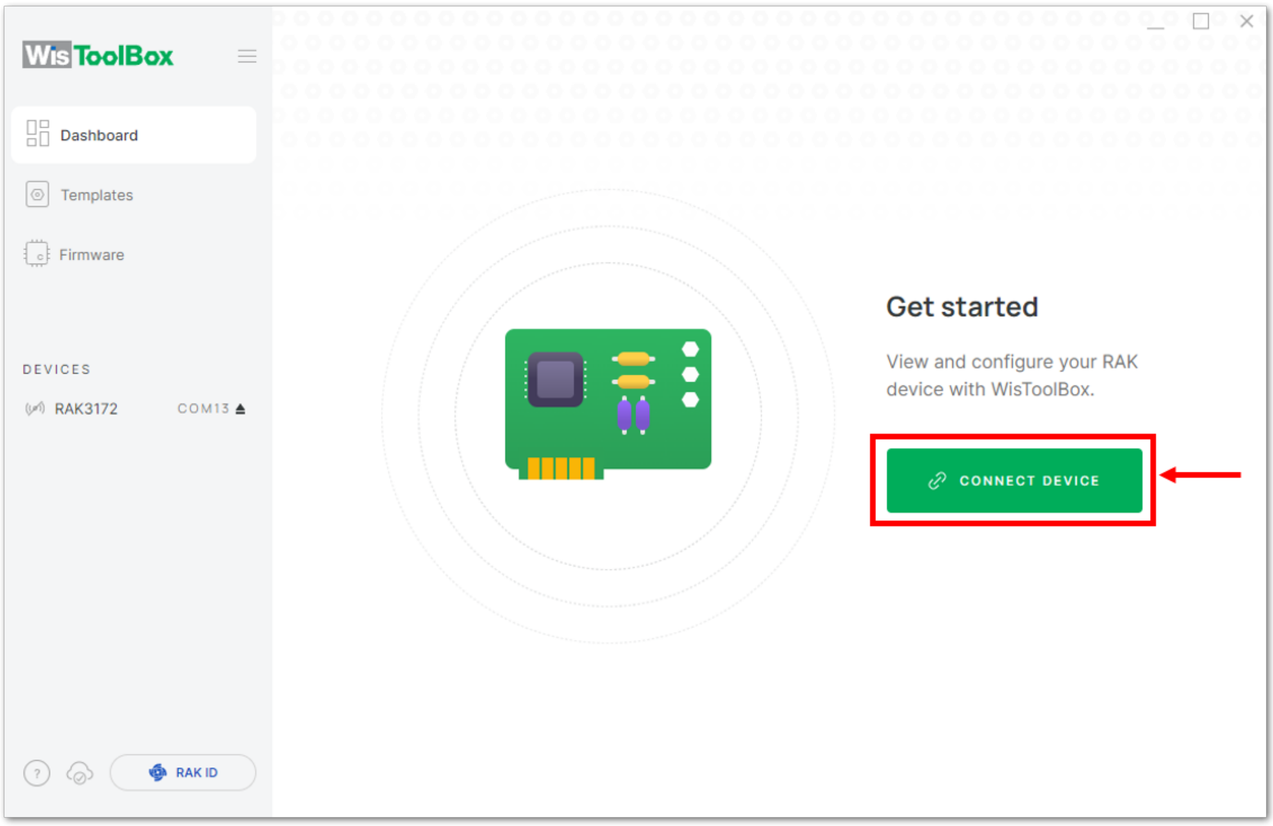
- ABP Configuration for TTN via WisToolBox UI
- ABP Configuration for TTN via WisToolBox Console

ABP Configuration for TTN via WisToolBox UI

The **RAK3172** should have the correct ABP credentials to connect to TTN. This can be done using **WisToolBox**. Below are the steps on setting up your **RAK3172** using **WisToolBox**.

1. Connect your **RAK3172** with your chosen WisBlock base board to the PC via USB cable and open the **WisToolBox** application.

2. Click the **CONNECT DEVICE** button to launch the WisToolBox Dashboard.

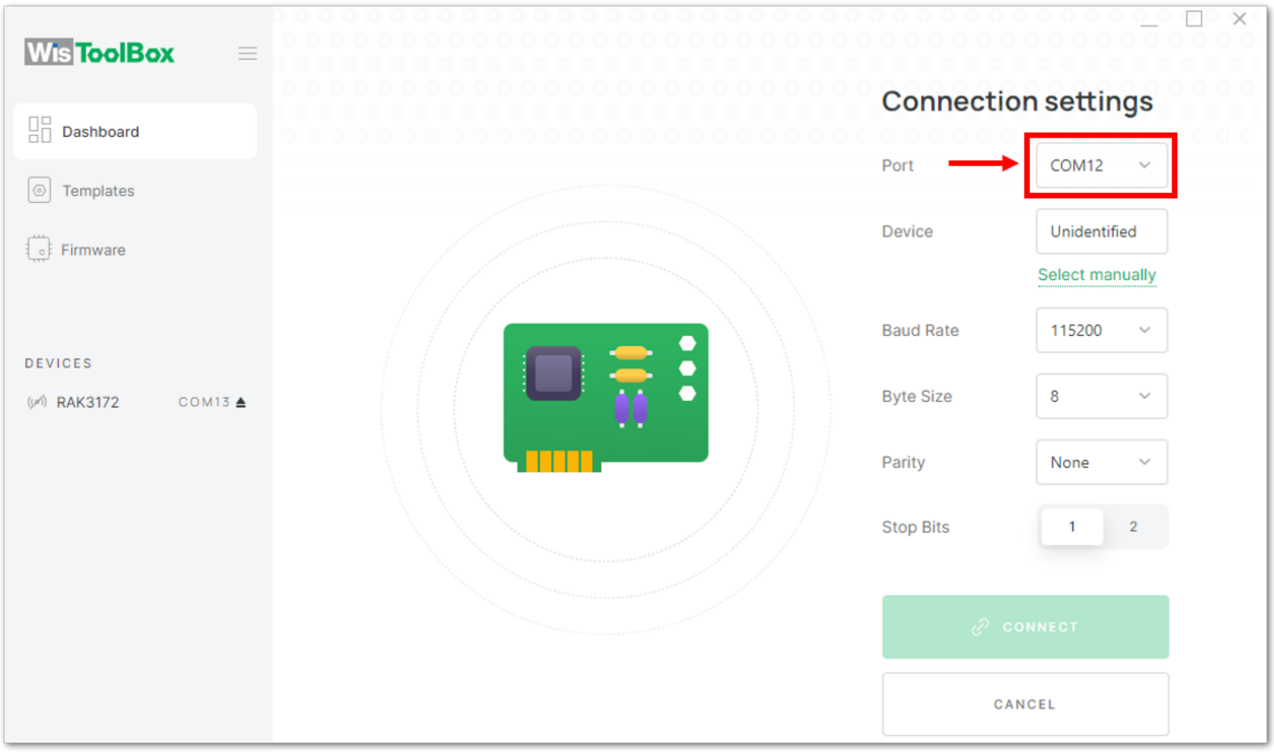


Get started

View and configure your RAK device with WisToolBox.

Figure 143: CONNECT DEVICE

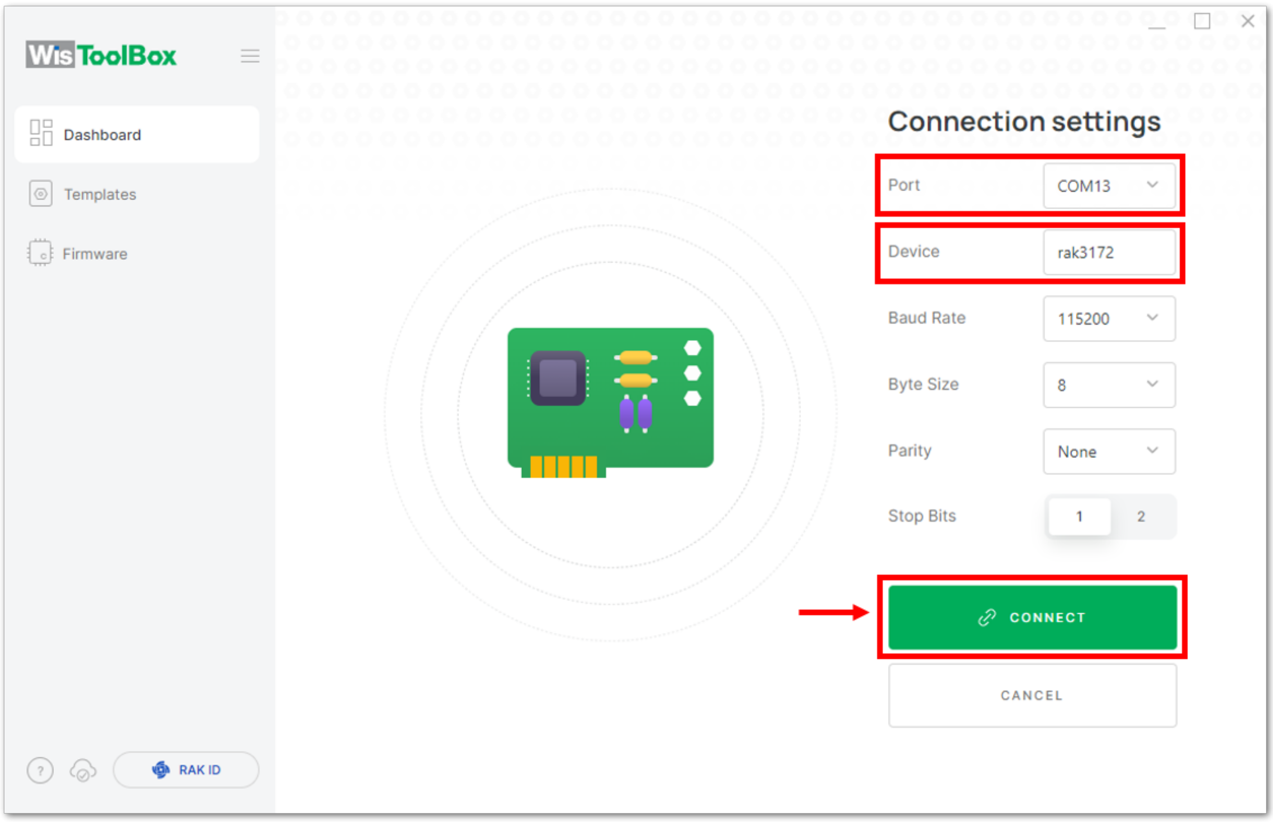
3. Select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT**, as shown in **Figure 145**.



Connection settings

Port	COM12
Device	Unidentified
Baud Rate	115200
Byte Size	8
Parity	None
Stop Bits	1 2

Figure 144: Setting up your device

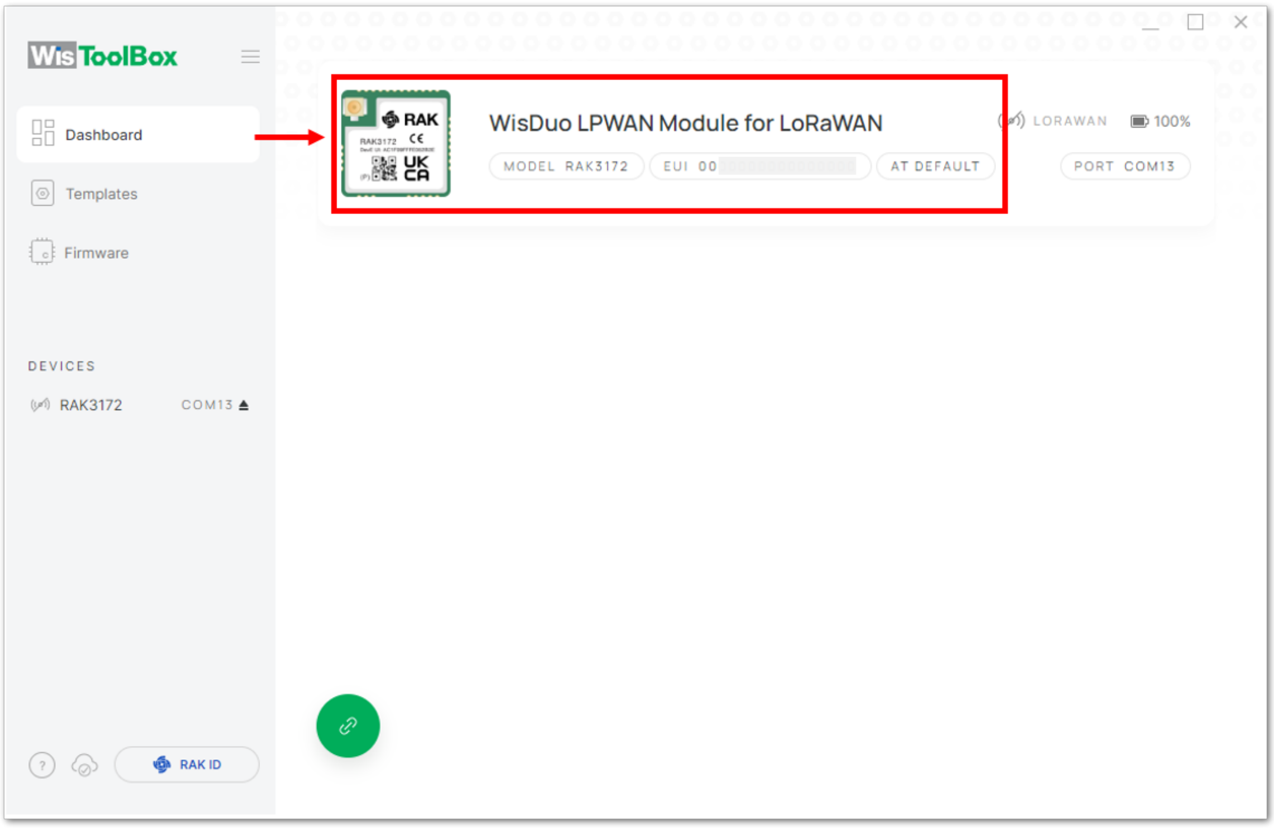


Connection settings

- Port: COM13
- Device: rak3172
- Baud Rate: 115200
- Byte Size: 8
- Parity: None
- Stop Bits: 1 2

Figure 145: Setting up your device

4. Once done, **RAK3172** will appear in the dashboard, and then select it.



WisDuo LPWAN Module for LoRaWAN

MODEL RAK3172 | EUI 00[redacted] | AT DEFAULT | LORAWAN | 100% | PORT COM13

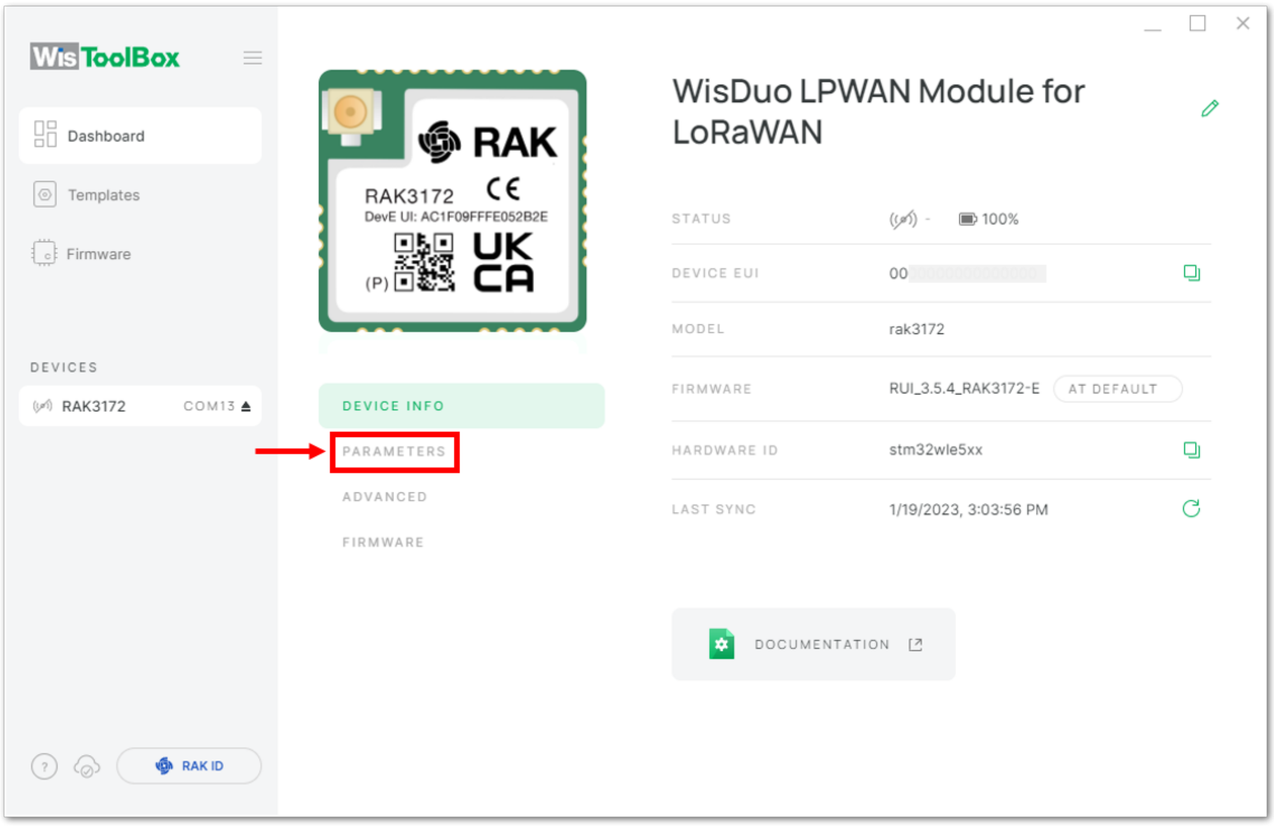
DEVICES

RAK3172 COM13

Figure 146: Device seen from WisToolBox dashboard

5. Then click **PARAMETERS** to do the configuration in your RAK3172.

> 📝 **NOTE:**
>
> - The **AppSKey**, **Device address**, and **NwkSKey** are hidden in this section as these are unique from a specific device.



WisDuo LPWAN Module for LoRaWAN

STATUS	- 100%
DEVICE EUI	00[redacted]
MODEL	rak3172
FIRMWARE	RUI_3.5.4_RAK3172-E AT DEFAULT
HARDWARE ID	stm32wle5xx
LAST SYNC	1/19/2023, 3:03:56 PM

Figure 147: Setting up your device

6. Click **Global settings** to set the network mode into **LoRaWAN** and join mode to **ABP**. Make sure that the active region is using **EU868** for this configuration. If you wish to work on other regional bands, you can choose among active regions based on your location.

- LoRa network mode: **LoRaWAN**
- LoRaWAN join mode: **ABP**
- LoRaWAN region: **EU868**

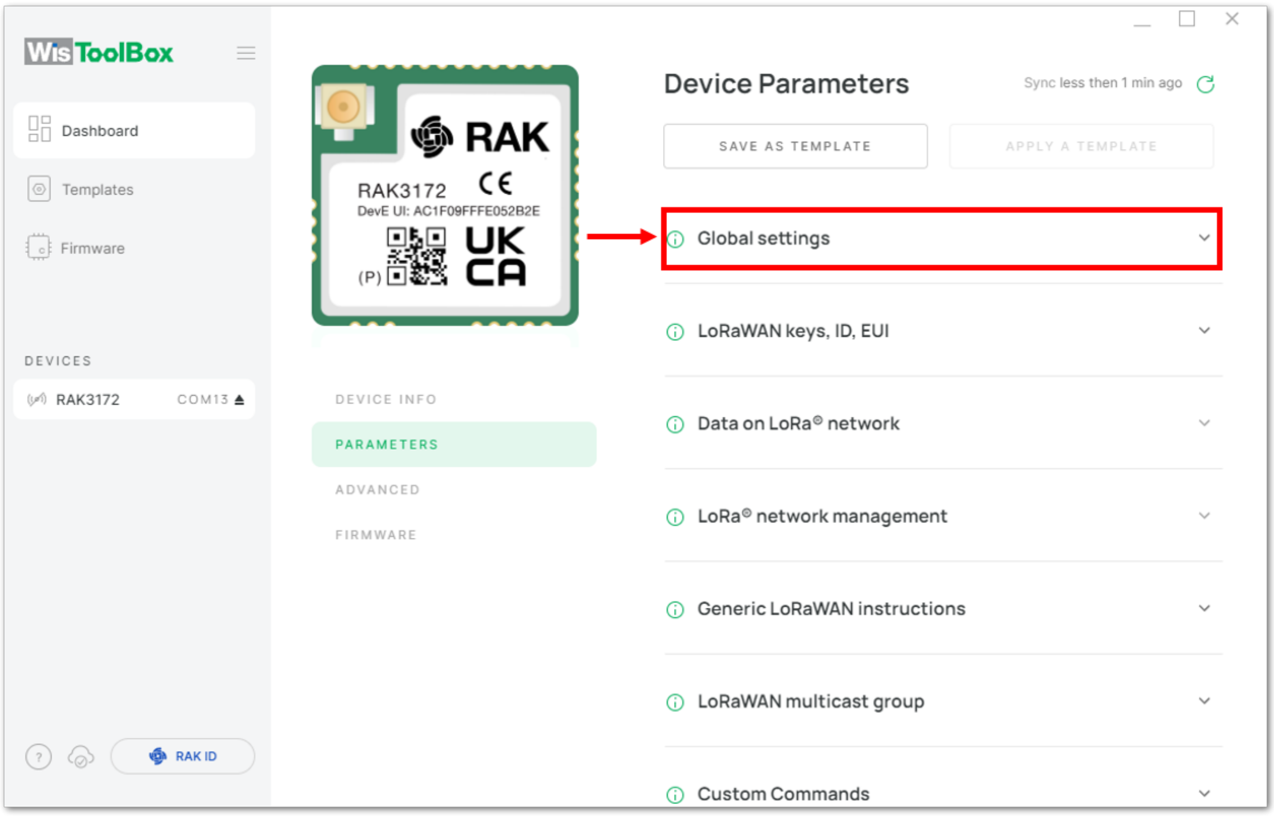
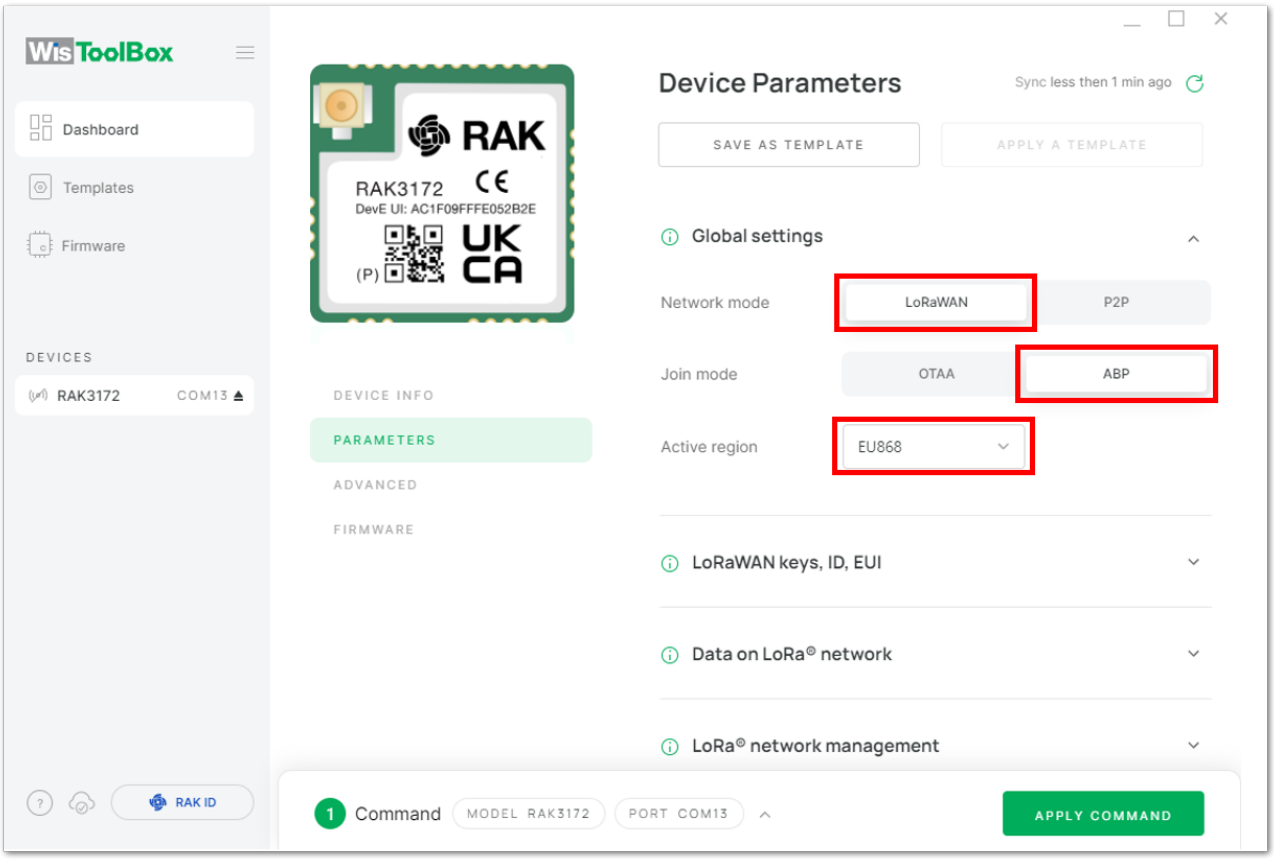


Figure 148: Global settings



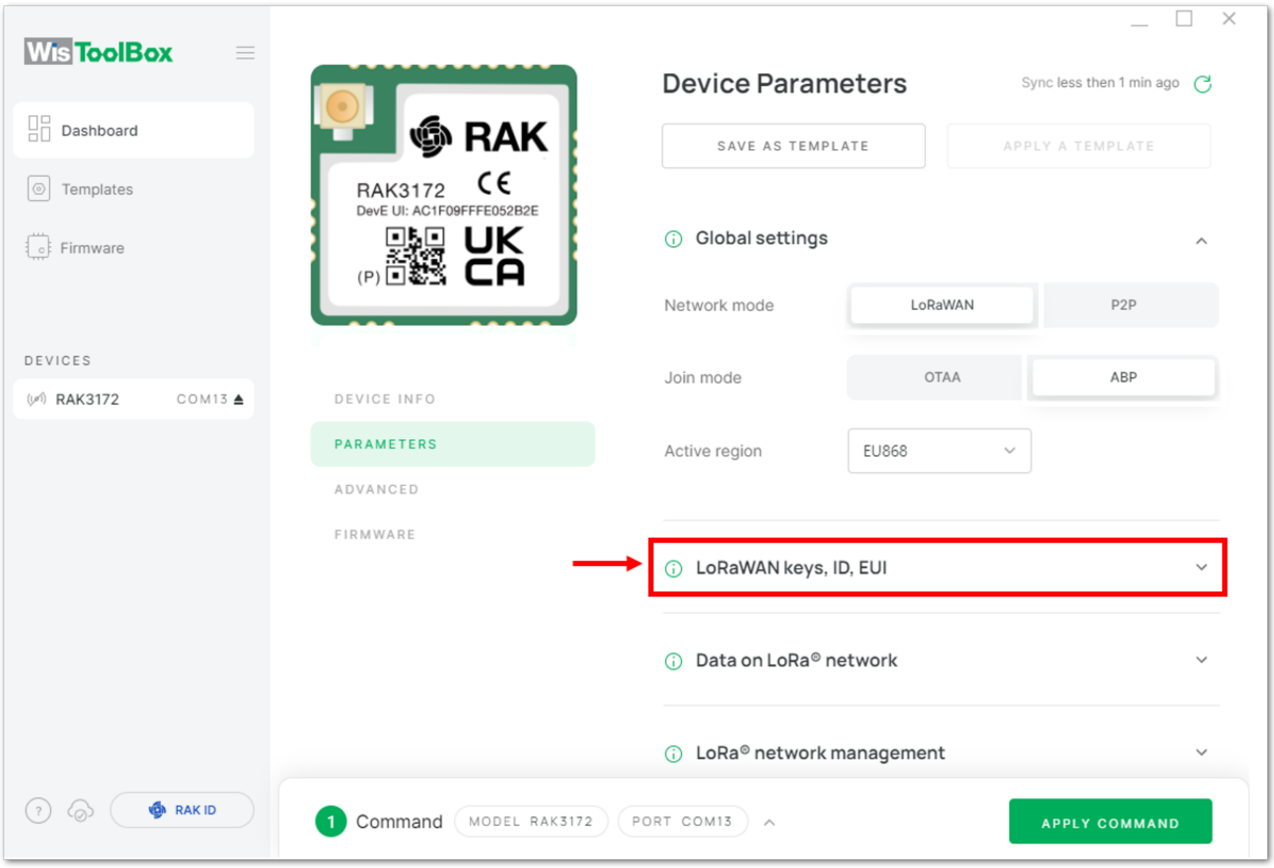
Network mode: LoRaWAN

Join mode: ABP

Active region: EU868

Figure 149: Global settings

7. Then click **LoRaWAN keys, ID, EUI** to configure the **Application session key (AppSKey)**, **Device address** and **Network session key (NwkSKey)**.

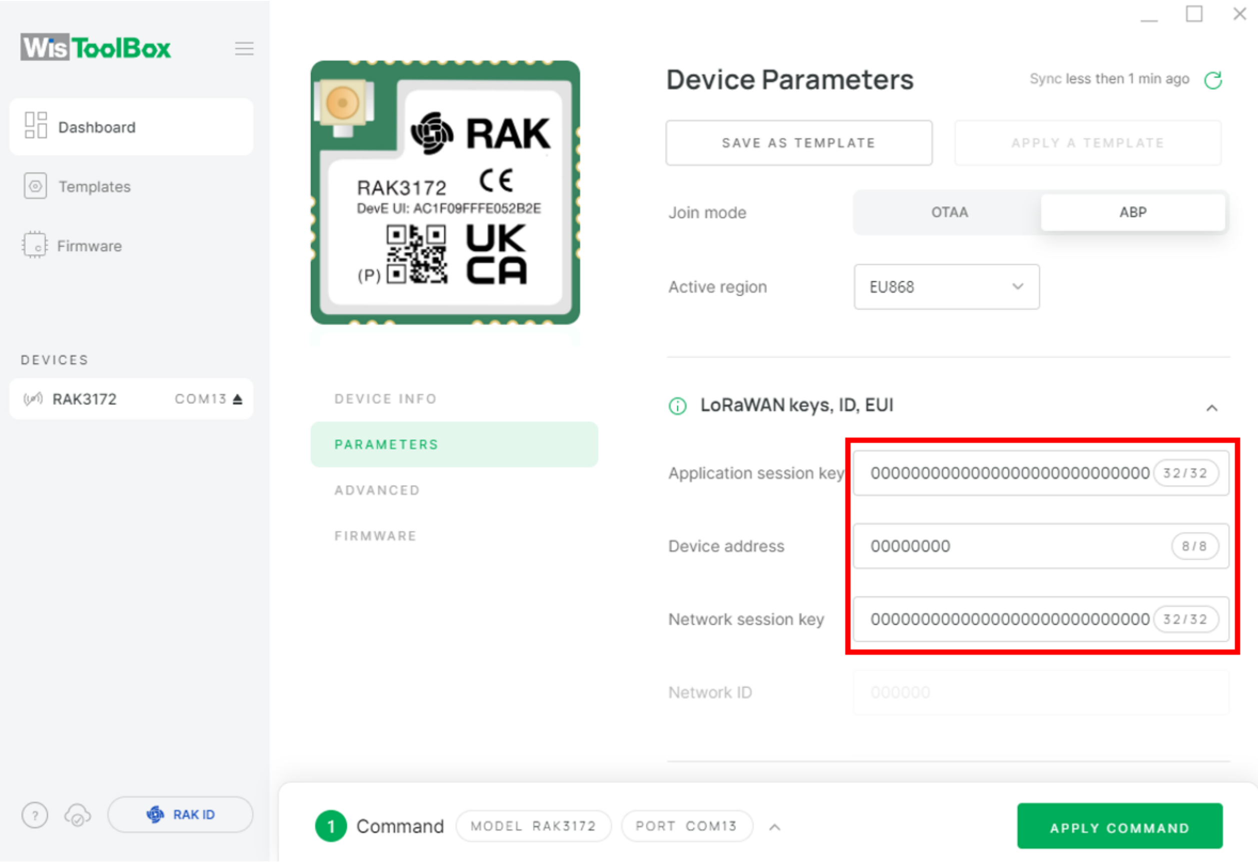


Network mode: LoRaWAN

Join mode: ABP

Active region: EU868

Figure 150: LoRaWAN keys, ID, EUI



Device Parameters

Sync less then 1 min ago

Join mode: ABP

Active region: EU868

LoRaWAN keys, ID, EUI

Application session key: 00000000000000000000000000000000 32/32

Device address: 00000000 8/8

Network session key: 00000000000000000000000000000000 32/32

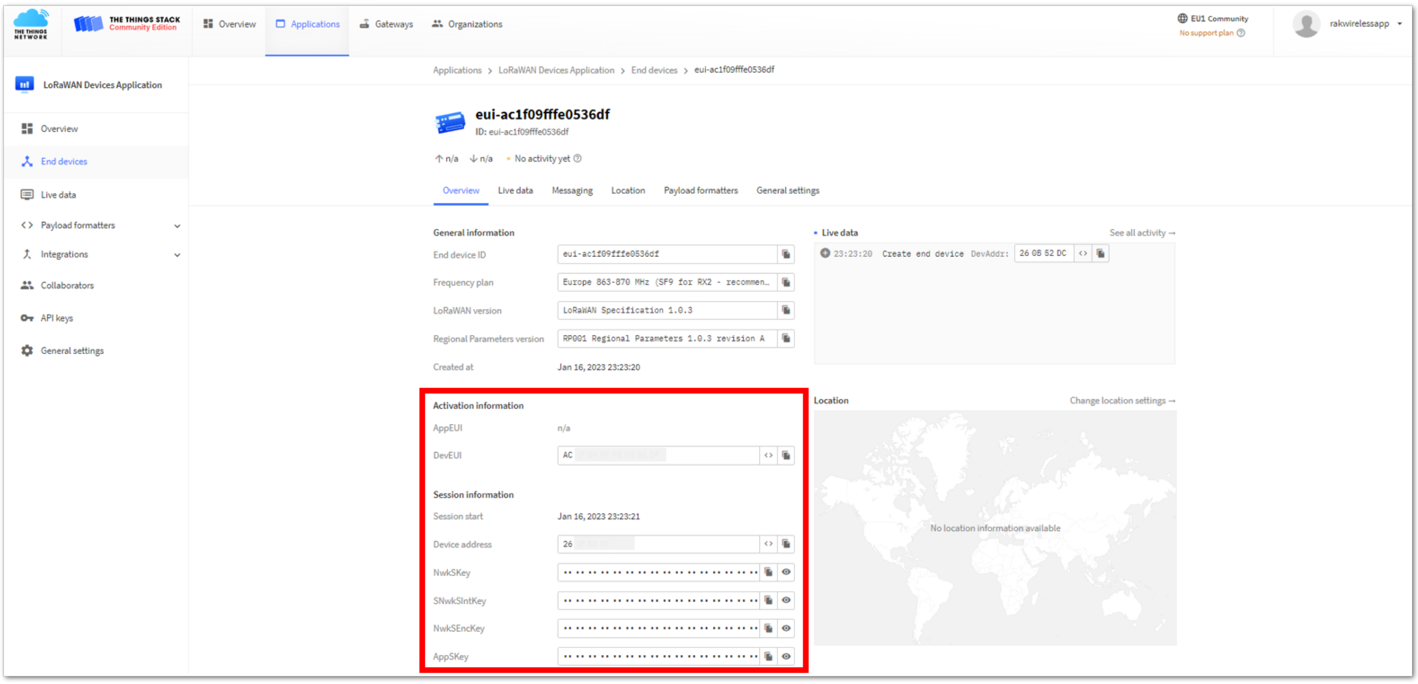
Network ID: 000000

Command MODEL RAK3172 PORT COM13

Figure 151: Setting up your device

8. Then go back to the console where your RAK3172 End device is created previously. Then copy all the credentials from there. Those will be the ones to be used also in the WisToolBox dashboard. Once encoded into the dashboard, click **APPLY COMMANDS** to update your device, as shown in **Figure 159**.

> 📝 **NOTE:**
>
> - The **AppSKey**, **Device address**, and **NwkSKey** are hidden in this section as these are unique from a specific device.



eui-ac1f09fffe0536df

ID: eui-ac1f09fffe0536df

↑ n/a ↓ n/a • No activity yet

General information

End device ID: eui-ac1f09fffe0536df

Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Jan 16, 2023 23:23:20

Live data

23:23:20 Create end device DevAddr: 26 0B 52 DC

Activation information

AppEUI: n/a

DevEUI: AC [redacted]

Session information

Session start: Jan 16, 2023 23:23:21

Device address: 26 [redacted]

NwkSKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

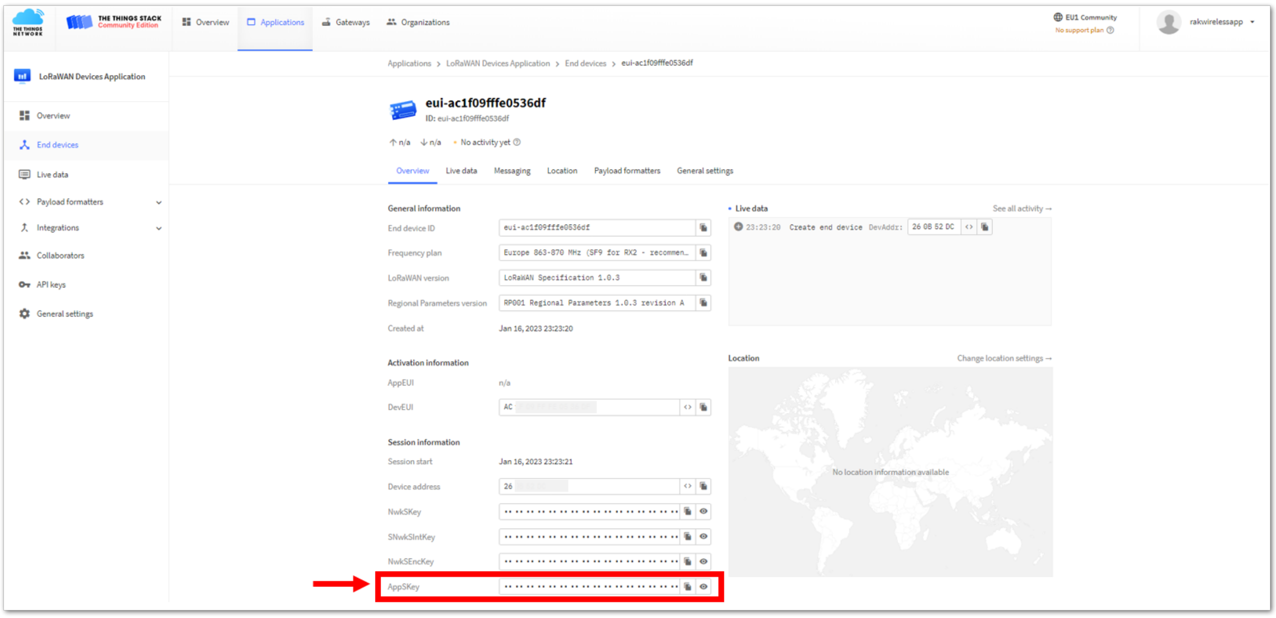
SNwkSIntKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

NwkSEncKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

AppSKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Figure 152: Your created ABP device from your console

- **For Application session key (AppSKey)**



eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df

↑ n/a ↓ n/a • No activity yet

General information

End device ID	eui-ac1f09fffe0536df
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 16, 2023 23:23:20

Activation information

AppEUI	n/a
DevEUI	AC [redacted]

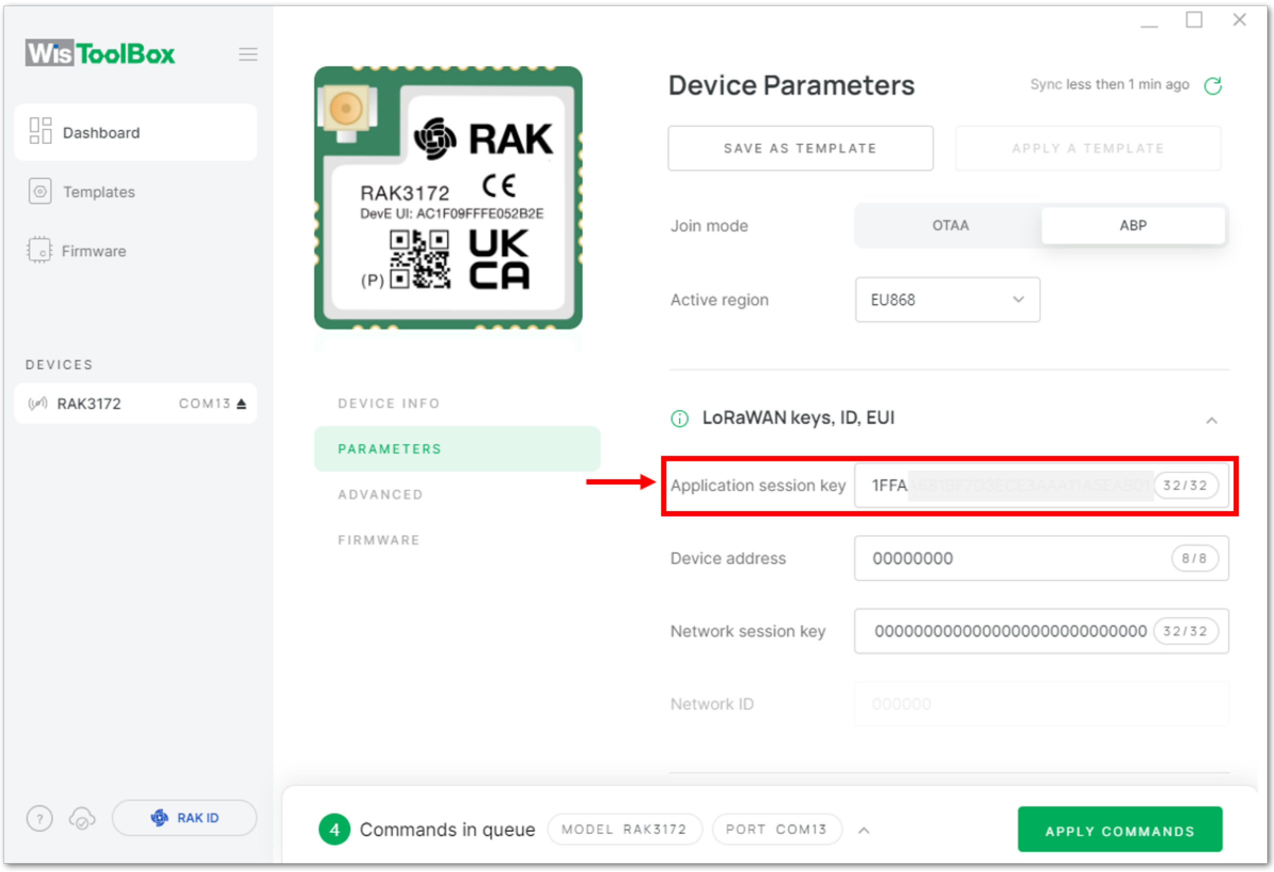
Session information

Session start	Jan 16, 2023 23:23:21
Device address	26 [redacted]
NwkSKey	••••••••••••••••••••••••••••••••
SNwkSIntKey	••••••••••••••••••••••••••••••••
NwkSEncKey	••••••••••••••••••••••••••••••••
AppSKey	••••••••••••••••••••••••••••••••

• Live data

```
23:23:20 Create end device DevAddr: 26 0B 52 DC
```

Figure 153: Copying the AppSKey credential from TTN to WisToolBox



Device Parameters

Sync less then 1 min ago

Join mode	OTAA / ABP
Active region	EU868

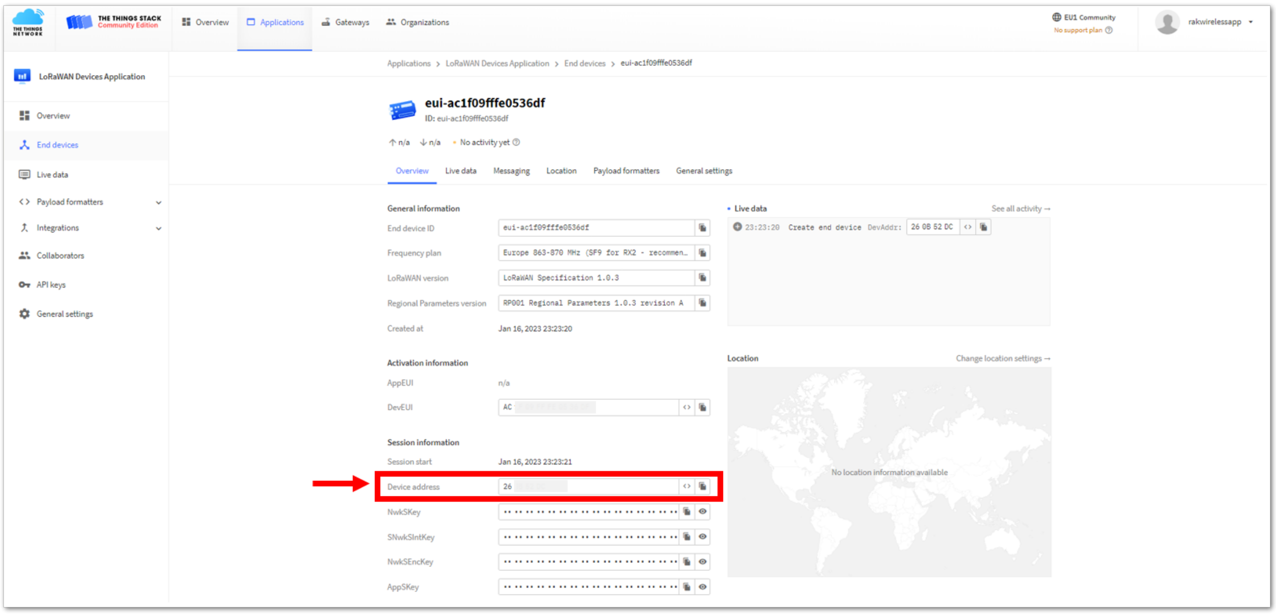
LoRaWAN keys, ID, EUI

Application session key	1FFA [redacted]	32/32
Device address	00000000	8/8
Network session key	00000000000000000000000000000000	32/32
Network ID	000000	

4 Commands in queue MODEL RAK3172 PORT COM13

Figure 154: Copying the AppSKey credential from TTN to WisToolBox

- **For Device address**



Applications > LoRaWAN Devices Application > End devices > eui-ac1f09fffe0536df

eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df

↑ n/a ↓ n/a • No activity yet

General information

End device ID	eui-ac1f09fffe0536df
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen...
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 16, 2023 23:23:20

Activation information

AppEUI	n/a
DevEUI	AC [redacted]

Session information

Session start	Jan 16, 2023 23:23:21
Device address	26 [redacted]
NwkSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

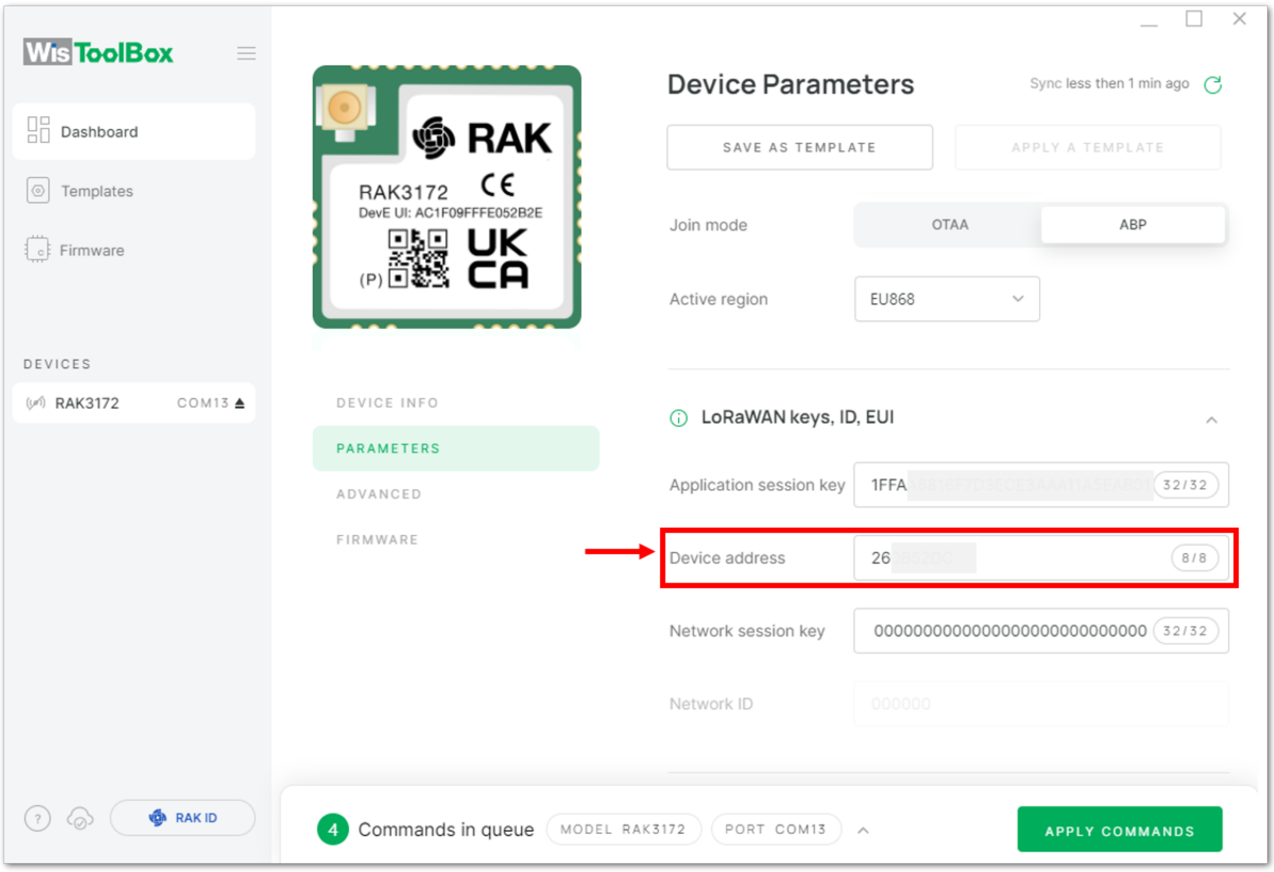
Live data

23:23:20 Create end device DevAddr: 26 0B 52 DC

Location

No location information available

Figure 155: Copying the Device address credential from TTN to WisToolBox



Device Parameters

Sync less then 1 min ago

Join mode	OTAA / ABP
Active region	EU868

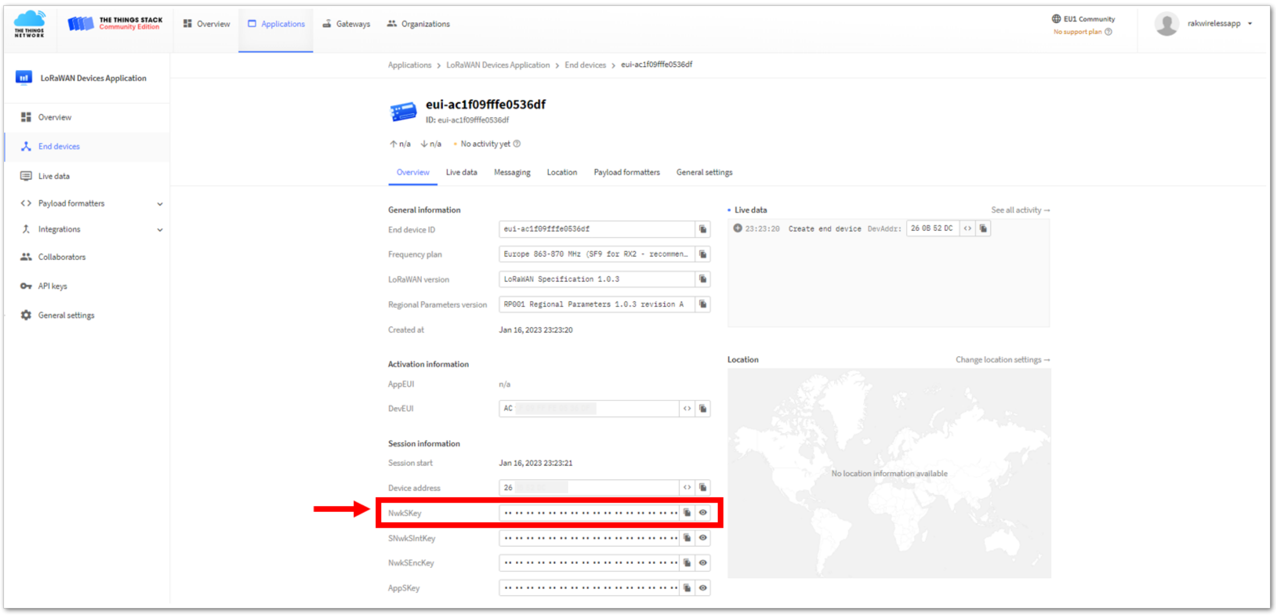
LoRaWAN keys, ID, EUI

Application session key	1FFA[redacted]	32/32
Device address	26[redacted]	8/8
Network session key	00000000000000000000000000000000	32/32
Network ID	000000	

4 Commands in queue MODEL RAK3172 PORT COM13

Figure 156: Copying the Device address credential from TTN to WisToolBox

- **For Network session key (NwkSKey)**



Applications > LoRaWAN Devices Application > End devices > eui-ac1f09fffe0536df

eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df

↑ n/a ↓ n/a • No activity yet

Overview | Live data | Messaging | Location | Payload formatters | General settings

General information

End device ID	eui-ac1f09fffe0536df
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 16, 2023 23:23:20

Activation information

AppEUI	n/a
DevEUI	AC [redacted]

Session information

Session start	Jan 16, 2023 23:23:21
Device address	26 [redacted]
NwkSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

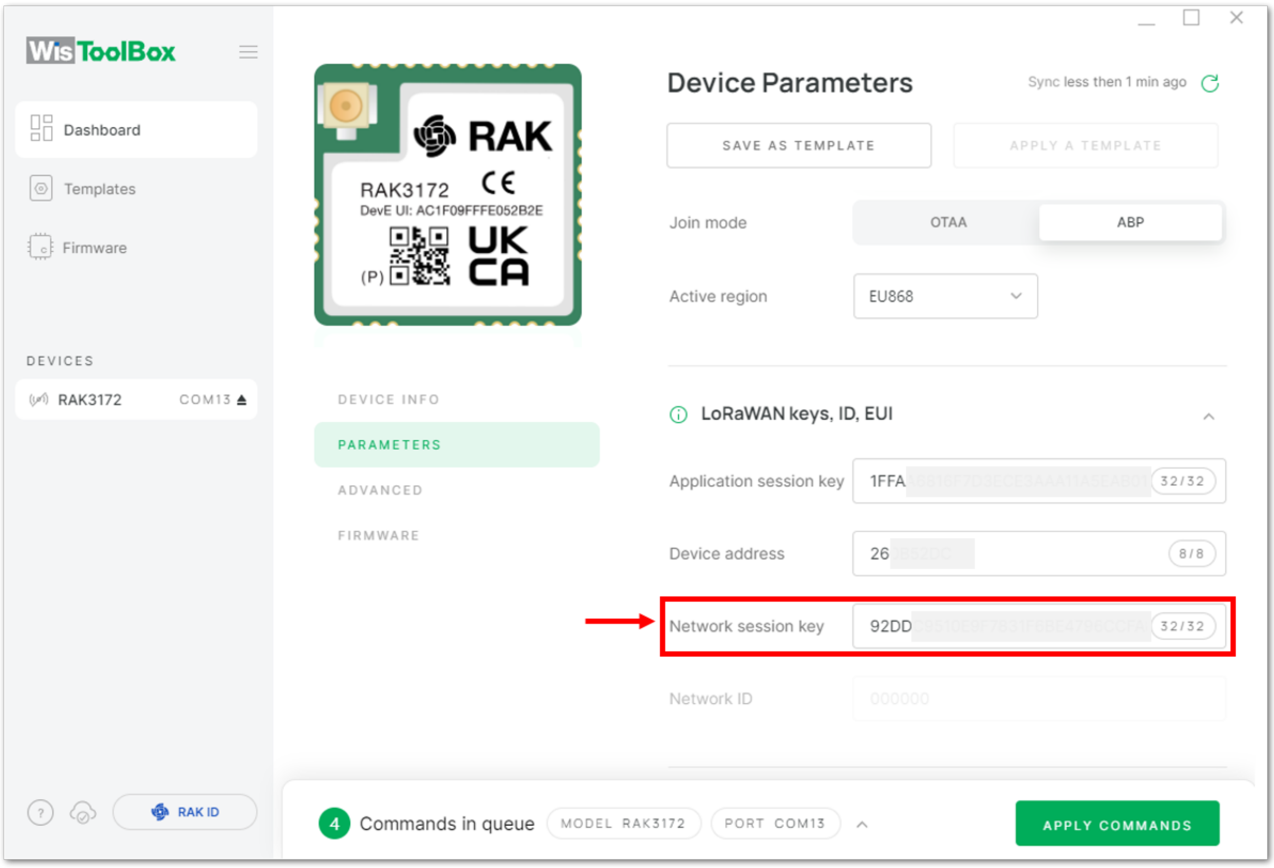
Live data — See all activity →

23:23:20 Create end device DevAddr: 26 0B 52 DC

Location — Change location settings →

No location information available

Figure 157: Copying the NwkSKey credential from TTN to WisToolBox



Device Parameters — Sync less then 1 min ago

SAVE AS TEMPLATE | APPLY A TEMPLATE

Join mode	OTAA / ABP
Active region	EU868

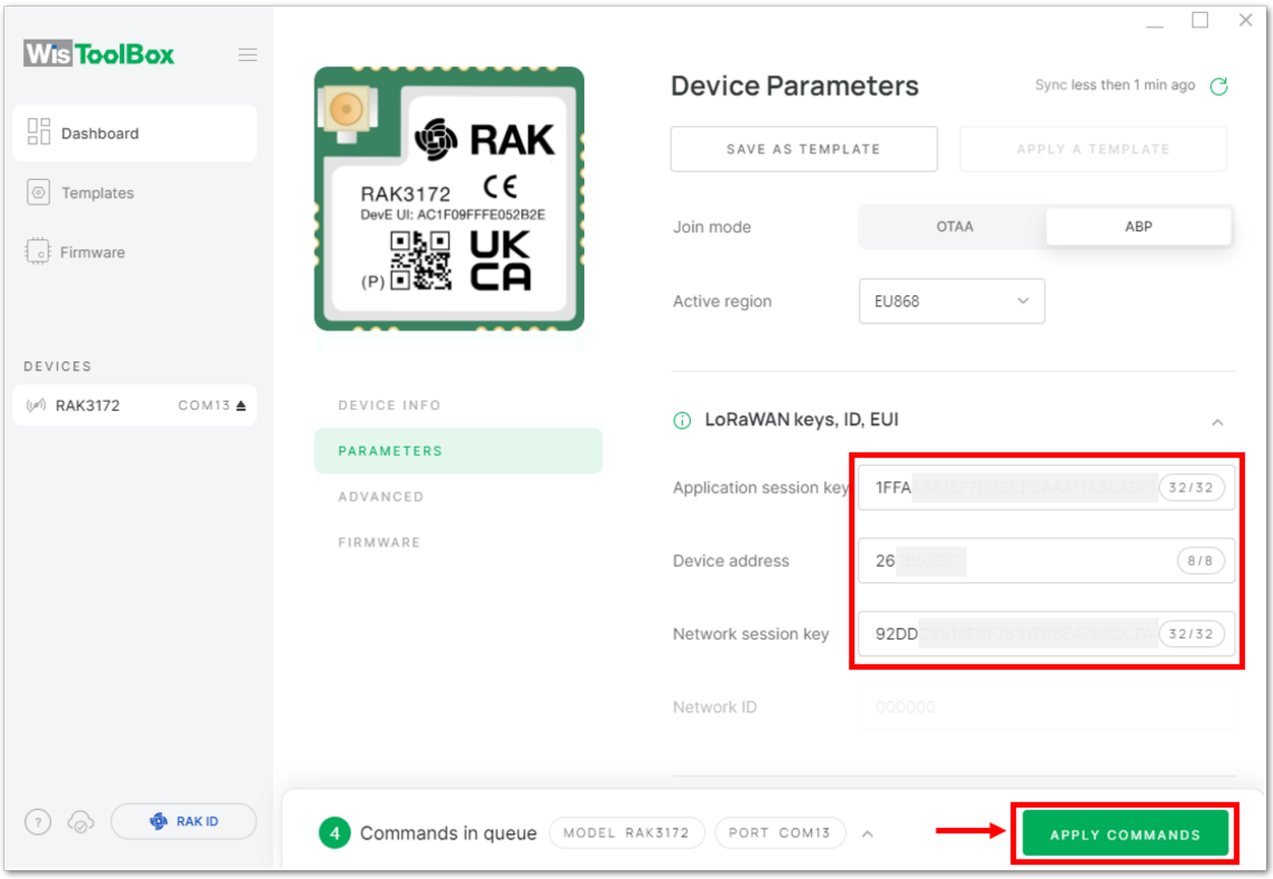
LoRaWAN keys, ID, EUI

Application session key	1FFA[redacted]	32/32
Device address	26[redacted]	8/8
Network session key	92DD[redacted]	32/32
Network ID	000000	

4 Commands in queue — MODEL RAK3172 — PORT COM13 — APPLY COMMANDS

Figure 158: Copying the NwkSKey credential from TTN to WisToolBox

- **WisToolBox Dashboard**



Device Parameters

Sync less then 1 min ago

Join mode: OTAA | ABP

Active region: EU868

LoRaWAN keys, ID, EUI

Application session key: 1FFA[redacted] 32/32

Device address: 26[redacted] 8/8

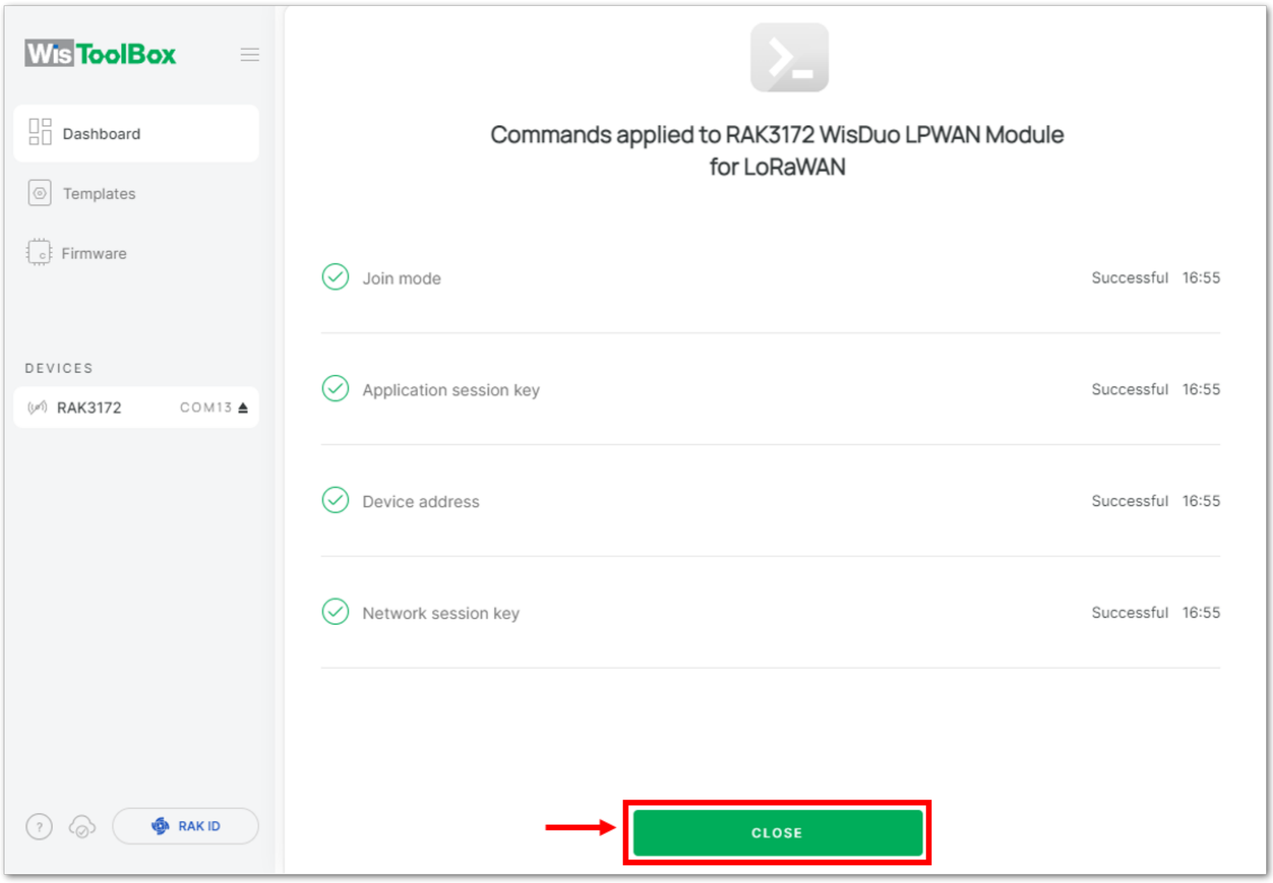
Network session key: 92DD[redacted] 32/32

Network ID: 000000

4 Commands in queue MODEL RAK3172 PORT COM13

Figure 159: Used credentials from your console in WisToolBox dashboard

9. Once done, you will see the summary of commands that is applied to your device. Then click **CLOSE**.



Commands applied to RAK3172 WisDuo LPWAN Module for LoRaWAN

Join mode Successful 16:55

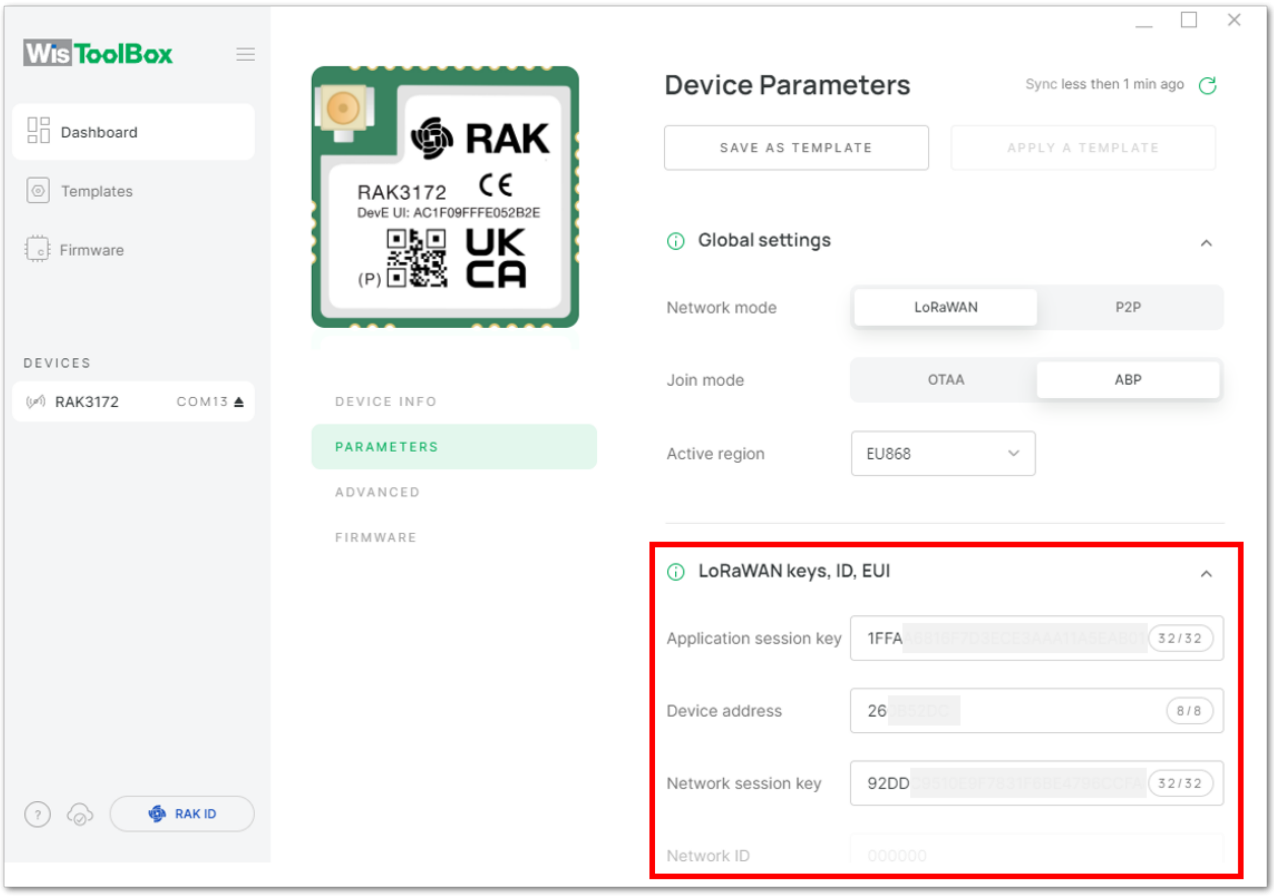
Application session key Successful 16:55

Device address Successful 16:55

Network session key Successful 16:55

Figure 160: Summary of commands

10. Now, you will see it returns to the dashboard with updated credentials of your device.



Device Parameters

Global settings

Network mode: LoRaWAN | P2P (LoRaWAN selected)

Join mode: OTAA | ABP (ABP selected)

Active region: EU868

LoRaWAN keys, ID, EUI

Application session key: 1FFA[redacted] 32/32

Device address: 26[redacted] 8/8

Network session key: 92DD[redacted] 32/32

Network ID: 000000

Figure 161: Successfully configured ABP device via WisToolBox dashboard

ABP Configuration for TTN via WisToolBox Console

Here's another way of ABP configuration using **WisToolBox Console**. Below are the steps on setting up your **RAK3172** using **WisToolBox Console**.

1. Connect your **RAK3172** with your chosen WisBlock base board to the PC via USB cable and open the **WisToolBox** application.

2. Click the **CONNECT DEVICE** button to launch the WisToolBox Dashboard.

   CONNECT DEVICE

   Figure 162: CONNECT DEVICE

3. Select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT**, as shown in **Figure 164**.

   Setting up your device

   Figure 163: Setting up your device

   Setting up your device

   Figure 164: Setting up your device

4. Once done, **RAK3172** will appear in the dashboard, and then select it.

   Device seen from WisToolBox dashboard

   Figure 165: Device seen from WisToolBox dashboard

5. Then click **ADVANCED**.

   Setting up your device

   Figure 166: Setting up your device

6. Once done, click **OPEN CONSOLE** to do the configuration.

OPEN CONSOLE

Figure 167: OPEN CONSOLE

Opening the Console terminal of WisToolBox

Figure 168: Opening the Console terminal of WisToolBox

Opening the Console terminal of WisToolBox

Figure 169: Opening the Console terminal of WisToolBox

7. To start the configuration, type **ATE** so you can echo the commands you input during your configuration. Then press **Enter**.

It is recommended to start by testing the console and verify that the current configuration is working by sending these two AT commands:

```
AT
```

```
ATE
```

`ATE` is useful for tracking the commands and troubleshooting.

You will receive `OK` when you input the two commands. After setting `ATE`, you can now see all the commands you input together with the replies.

> 📝 **NOTE:**
>
> If there is no `OK` or any reply, check if the device is powered correctly. If you are getting power from a USB port, ensure that you have a good USB cable.

Setting up your Console

Figure 170: Setting up your Console

Setting up your Console

Figure 171: Setting up your Console

Setting up your Console

Figure 172: Setting up your Console

8. Then configure the LoRaWAN join mode to **ABP**. You can check what parameter you will input by typing **AT+NJM?**, and then **Enter** into the console terminal. For **ABP**, you should input **AT+NJM=0**, and then press **Enter** as shown in **Figure 173**.

Setting up your Console

Figure 173: Setting up your Console

Setting up your Console

Figure 174: Setting up your Console

Setting up your Console

Figure 175: Setting up your Console

9. Once done, set up your LoRaWAN region to EU868. You can check what parameter you will input by typing **AT+BAND?**, and then **Enter** into the console terminal. For **EU868**, you should input **AT+BAND=4** then press **Enter**. If you wish to work on other regional bands, you may check the list of band parameter options below.

Set the frequency/region to EU868.

```
AT+BAND=4
```

> 📝 **NOTE:**
>
> Depending on the Regional Band you selected, you might need to configure the sub-band of your RAK3172 to match the gateway and LoRaWAN network server. This is especially important for Regional Bands like US915, AU915, and CN470.
>
> To configure the masking of channels for the sub-bands, you can use the `AT+MASK` command that can be found on the AT Command Manual .
>
> To illustrate, you can use sub-band 2 by sending the command `AT+MASK=0002` .

List of band parameter options

Code	Regional Band
0	EU433
1	CN470
2	RU864
3	IN865
4	EU868
5	US915
6	AU915
7	KR920
8	AS923-1
9	AS923-2
10	AS923-3
11	AS923-4

Setting up your Console

Figure 176: Setting up your Console

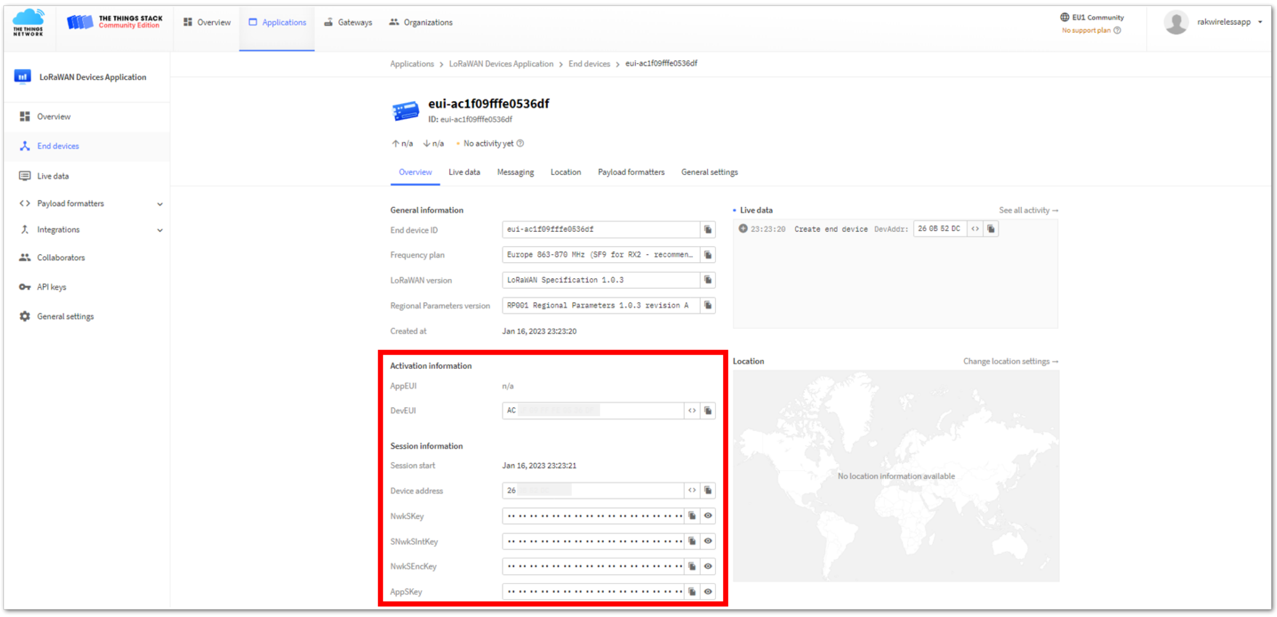
Setting up your Console

Figure 177: Setting up your Console

Setting up your Console

Figure 178: Setting up your Console

10. Then next to this will be updating the ABP credentials of your device. First to this will be the **Application session key (AppSKey)**. Go back to your console where your RAK3172 End device was created to copy the AppSKey credential then paste it to the WisToolBox Console then press **Enter**.



THE THINGS STACK Community Edition | Overview | Applications | Gateways | Organizations | EU1 Community No support plan | rakwirelessapp

Applications > LoRaWAN Devices Application > End devices > eui-ac1f09fffe0536df

LoRaWAN Devices Application: Overview, End devices, Live data, Payload formatters, Integrations, Collaborators, API keys, General settings

eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df
↑ n/a ↓ n/a • No activity yet

Overview | Live data | Messaging | Location | Payload formatters | General settings

General information

End device ID	eui-ac1f09fffe0536df
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 16, 2023 23:23:20

Activation information

AppEUI	n/a
DevEUI	AC [redacted]

Session information

Session start	Jan 16, 2023 23:23:21
Device address	26 [redacted]
NwkSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Live data — See all activity →

```
23:23:20  Create end device  DevAddr: 26 0B 52 DC
```

Location — Change location settings →

No location information available

Figure 179: Your created ABP device from your TTN console

Setting up your Console

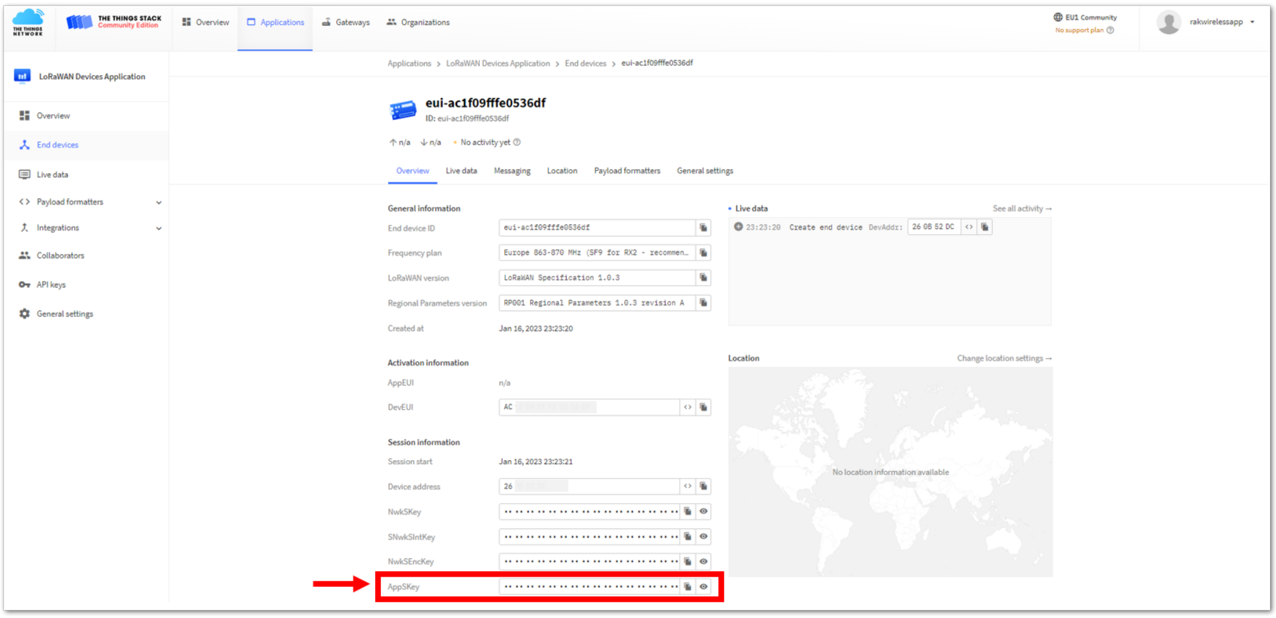
Figure 180: Setting up your Console

Setting up your Console

Figure 181: Setting up your Console

Setting up your Console

Figure 182: Setting up your Console



THE THINGS STACK Community Edition | Overview | Applications | Gateways | Organizations | EU1 Community No support plan | rakwirelessapp

Applications > LoRaWAN Devices Application > End devices > eui-ac1f09fffe0536df

LoRaWAN Devices Application: Overview, End devices, Live data, Payload formatters, Integrations, Collaborators, API keys, General settings

eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df
↑ n/a ↓ n/a • No activity yet

Overview | Live data | Messaging | Location | Payload formatters | General settings

General information

End device ID	eui-ac1f09fffe0536df
Frequency plan	Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version	LoRaWAN Specification 1.0.3
Regional Parameters version	RP001 Regional Parameters 1.0.3 revision A
Created at	Jan 16, 2023 23:23:20

Activation information

AppEUI	n/a
DevEUI	AC [redacted]

Session information

Session start	Jan 16, 2023 23:23:21
Device address	26 [redacted]
NwkSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey	•• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Live data — See all activity →

```
23:23:20  Create end device  DevAddr: 26 0B 52 DC
```

Location — Change location settings →

No location information available

Figure 183: Copying the AppSKey credential from TTN to WisToolBox

Setting up your Console

Figure 184: Setting up your Console

11. Once done, do the same procedure to **Device address** and **Network session key (NwkSKey)**.

- **For Device address**

Setting up your Console

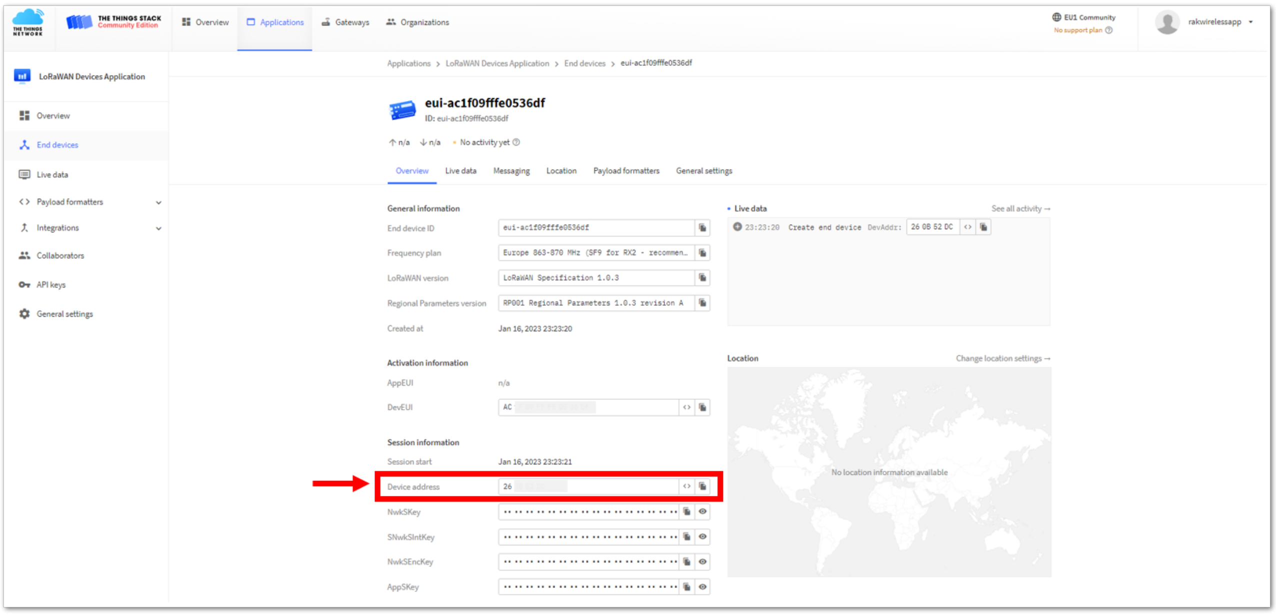
Figure 185: Setting up your Console

Setting up your Console

Figure 186: Setting up your Console

Setting up your Console

Figure 187: Setting up your Console



eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df

General information
End device ID: eui-ac1f09fffe0536df
Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version: LoRaWAN Specification 1.0.3
Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A
Created at: Jan 16, 2023 23:23:20

Activation information
AppEUI: n/a
DevEUI: AC [redacted]

Session information
Session start: Jan 16, 2023 23:23:21
Device address: 26 [redacted]
NwkSKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Live data
23:23:20 Create end device DevAddr: 26 0B 52 DC

Location

Figure 188: Copying the Device address credential from TTN to WisToolBox

Setting up your Console

Figure 189: Setting up your Console

- **For Network session key (NwkSKey)**

Setting up your Console

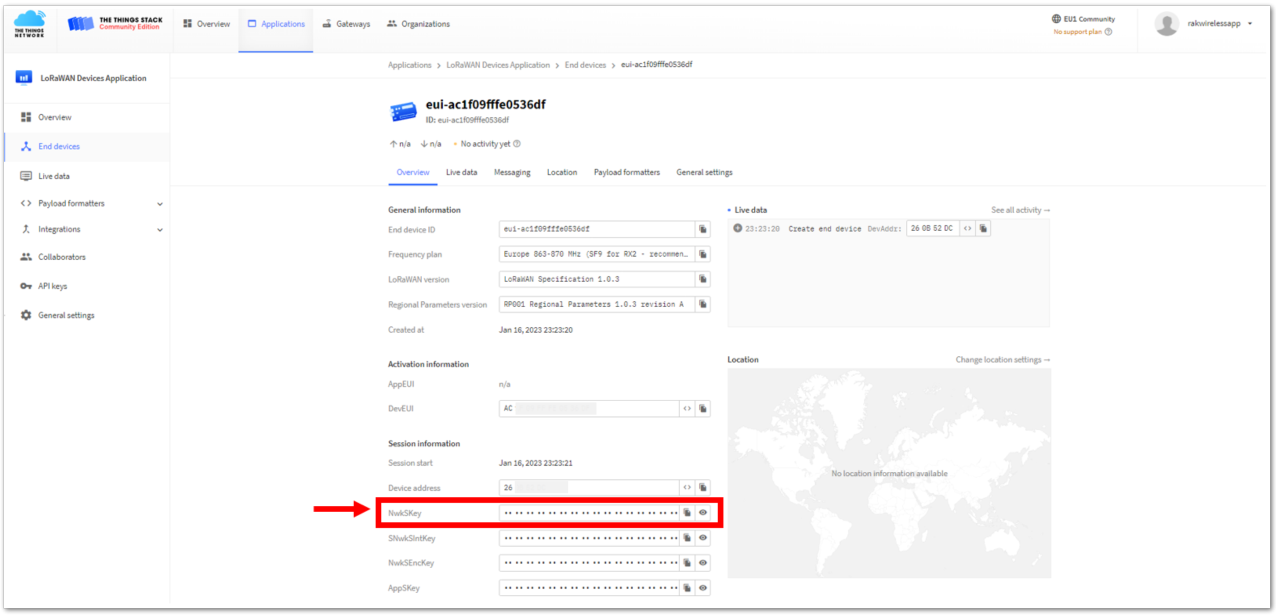
Figure 190: Setting up your Console

Setting up your Console

Figure 191: Setting up your Console

Setting up your Console

Figure 192: Setting up your Console



eui-ac1f09fffe0536df
ID: eui-ac1f09fffe0536df

General information
End device ID: eui-ac1f09fffe0536df
Frequency plan: Europe 863-870 MHz (SF9 for RX2 - recommen…
LoRaWAN version: LoRaWAN Specification 1.0.3
Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A
Created at: Jan 16, 2023 23:23:20

Activation information
AppEUI: n/a
DevEUI: AC [redacted]

Session information
Session start: Jan 16, 2023 23:23:21
Device address: 26 [redacted]
NwkSKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
SNwkSIntKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
NwkSEncKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••
AppSKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Live data
23:23:20 Create end device DevAddr: 26 0B 52 DC

Location

Figure 193: Copying the NwkSKey credential from TTN to WisToolBox

Setting up your Console

Figure 194: Setting up your Console

12. Once done, click **Dashboard** to check the updated credentials of your ABP device. Click **PARAMETERS** to open the **Global Settings** and **LoRaWAN keys, ID, EUI**, and check whether these portions are updated.

    Setting up your Console

    Figure 195: Setting up your Console

    Setting up your Console

    Figure 196: Setting up your Console

    PARAMETERS

    Figure 197: PARAMETERS

    Global settings and LoRaWAN keys, ID, EUI

    Figure 198: Global settings and LoRaWAN keys, ID, EUI

    Global settings and LoRaWAN keys, ID, EUI details

    Figure 199: Global settings and LoRaWAN keys, ID, EUI details

13. Now you have a configured ABP device using WisToolBox Console. **ABP-configured devices** are directly tied to the network once done with the above procedures so the joining procedure is not needed.

14. Now, you can try sending the payload to TTN. Open again the terminal console of WisToolBox to send some payload using it. Send command format: `AT+SEND=<port>:<payload>`

```
AT+SEND=2:12345678
```

    ABP device sending payload to the network

    Figure 200: ABP device sending payload to the network

    ABP device sending payload to the network

    Figure 201: ABP device sending payload to the network

    ABP device sending payload to the network

    Figure 202: ABP device sending payload to the network

    ABP device sending payload to the network

    Figure 203: ABP device sending payload to the network

15. You can see the data sent by the RAK3172 module on the TTN device console *Live data* section. Also, the *Last seen* info should be a few seconds or minutes ago.

    ABP Test Sample Data Sent Viewed in TTN

    Figure 204: ABP Test Sample Data Sent Viewed in TTN

Connecting with ChirpStack

This section shows how to connect the RAK3172 module to the ChirpStack platform.

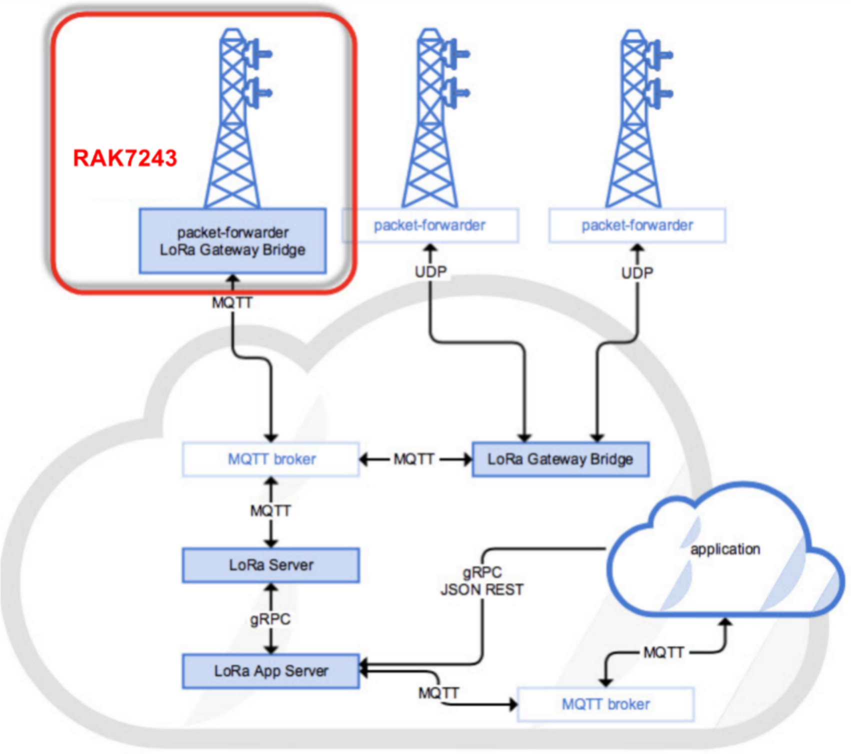


Figure 205: RAK3172 Module in the Context of the ChirpStack Platform

The ChirpStack, previously known as the LoRaServer project, provides open-source components for building LoRaWAN networks. In the case of TTN, the RAK3172 module is located in the periphery and will transmit the data to the backend servers through a LoRa gateway. Learn more about ChirpStack .

> 📝 **NOTE:**
>
> It is assumed that you are using a RAK Gateway and its built-in ChirpStack. Also, the gateway with the ChirpStack must be configured successfully. For further information, check the RAK documents for more details.

- In summary, these are the requirements:

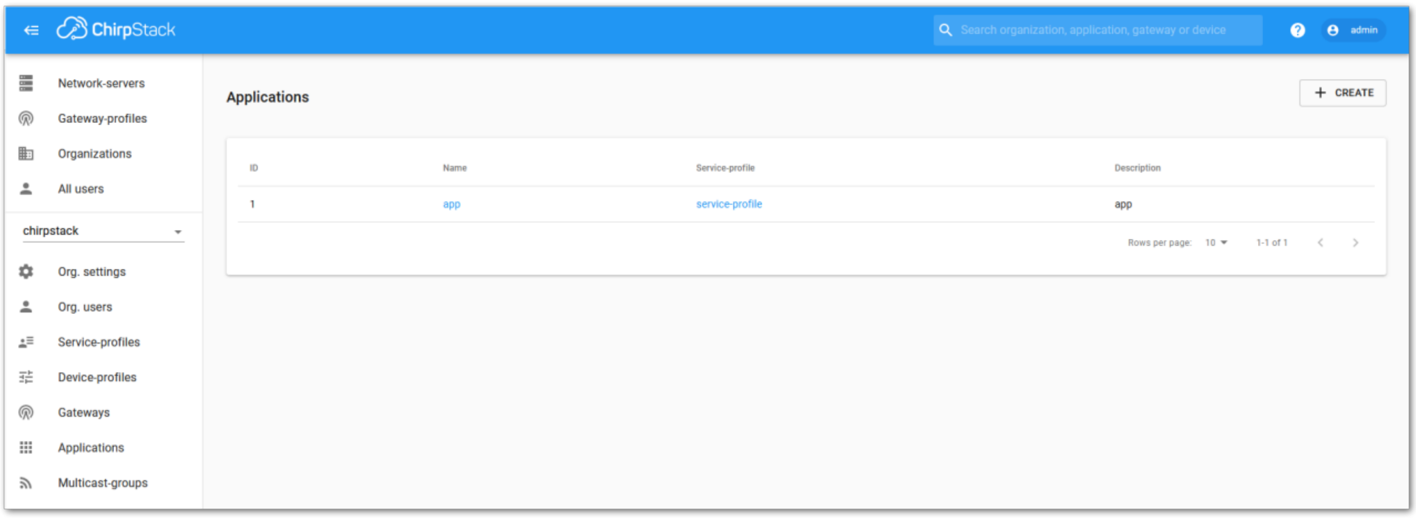
  1. In a ChirpStack online gateway, the frequency band of the nodes should be consistent with the frequency band of the gateway in use.
     - Connect the Gateway with Chirpstack
  2. The RAK Serial Port Tool provided by RAK
  3. RAK3172 module

> 📝 **NOTE:**
>
> The frequency band used in the demonstration is EU868. Use a high-frequency version of RAK3172. The product number should be **RAK3172 (H)**.

Create a New Application

1. Log in to the ChirpStack server using your account and password.

2. Go to the Application section, as shown in **Figure 206**.



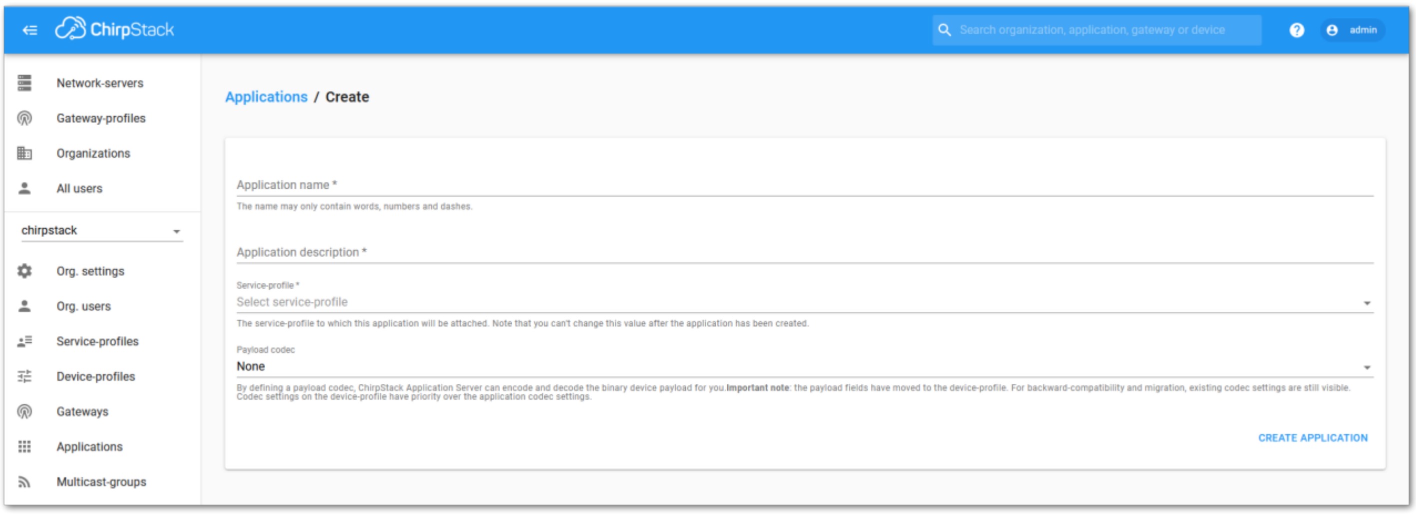
Applications

ID	Name	Service-profile	Description
1	app	service-profile	app

Rows per page: 10 1-1 of 1

Figure 206: Application Section

3. By default, you should create a new application, although you can reuse existing ones. For this setup, create a new Application by clicking on the **CREATE** button and filling in the required parameters, as shown in **Figure 207** and **Figure 208**.



Applications / Create

Application name *

The name may only contain words, numbers and dashes.

Application description *

Service-profile *

Select service-profile

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

None

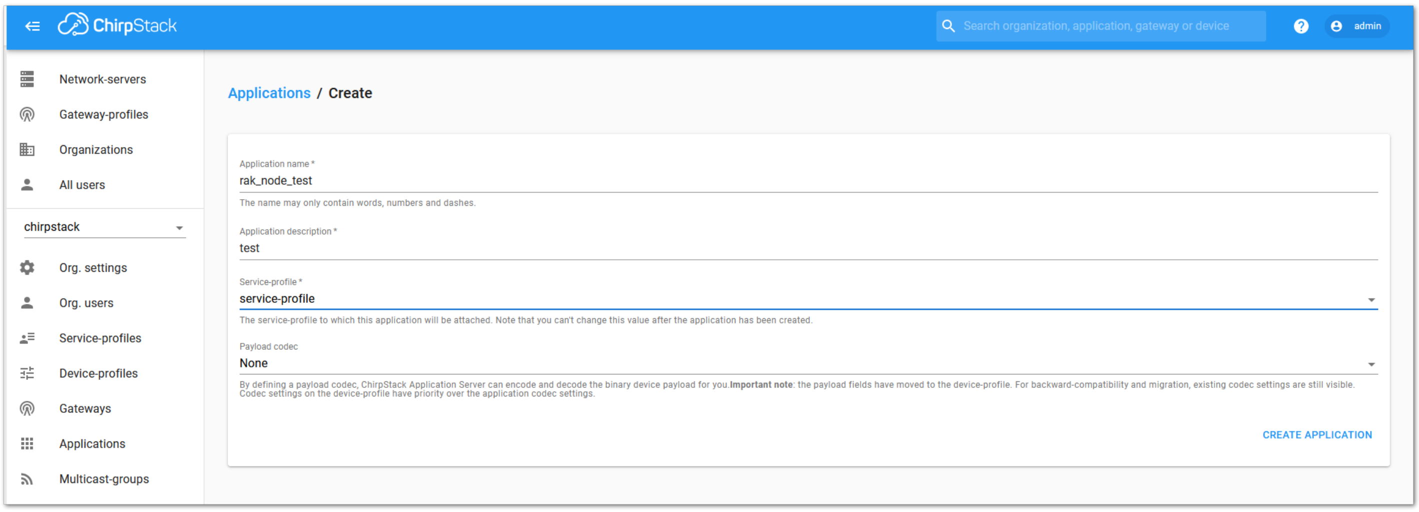
By defining a payload codec, ChirpStack Application Server can encode and decode the binary device payload for you. **Important note**: the payload fields have moved to the device-profile. For backward-compatibility and migration, existing codec settings are still visible. Codec settings on the device-profile have priority over the application codec settings.

Figure 207: Creating a New Application

- For this setup, create an Application named **rak_node_test**.

ChirpStack LoraServer supports multiple system configurations, with only one by default.

- **Service profile**: Field is to select the system profile.
- **Payload codec**: It is the parsing method for selecting load data, such as parsing LPP format data.



Applications / Create

Application name *
rak_node_test

The name may only contain words, numbers and dashes.

Application description *
test

Service-profile *
service-profile

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
None

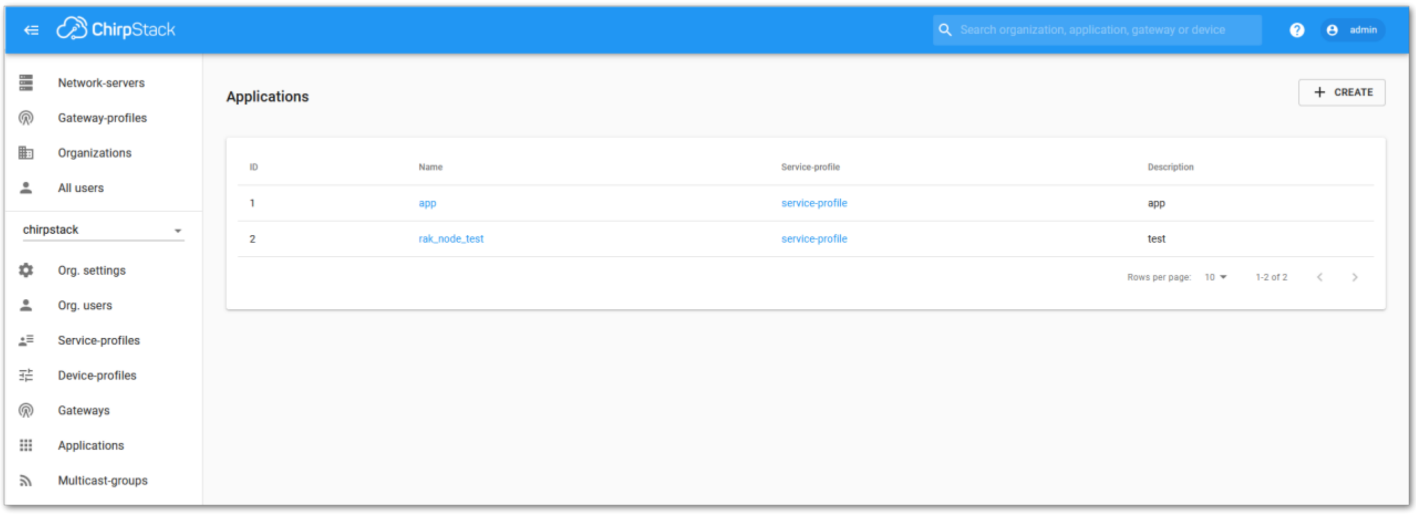
By defining a payload codec, ChirpStack Application Server can encode and decode the binary device payload for you. **Important note**: the payload fields have moved to the device-profile. For backward-compatibility and migration, existing codec settings are still visible. Codec settings on the device-profile have priority over the application codec settings.

Figure 208: Filling Parameters of an Application

Register a New Device

1. Choose the **Application** created in the previous step, then select the **DEVICES** tab, as shown in **Figure 209** and **Figure 210**.

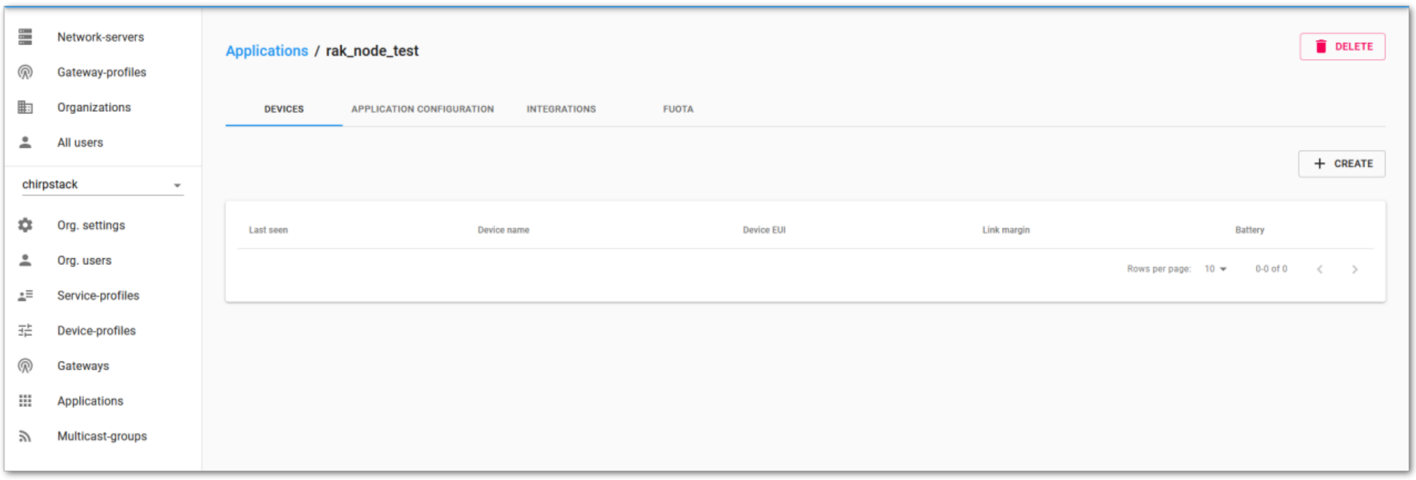
2. Once done, click the "**+ CREATE**" button.



ID	Name	Service-profile	Description
1	app	service-profile	app
2	rak_node_test	service-profile	test

Rows per page: 10 1-2 of 2

Figure 209: List of Applications Created

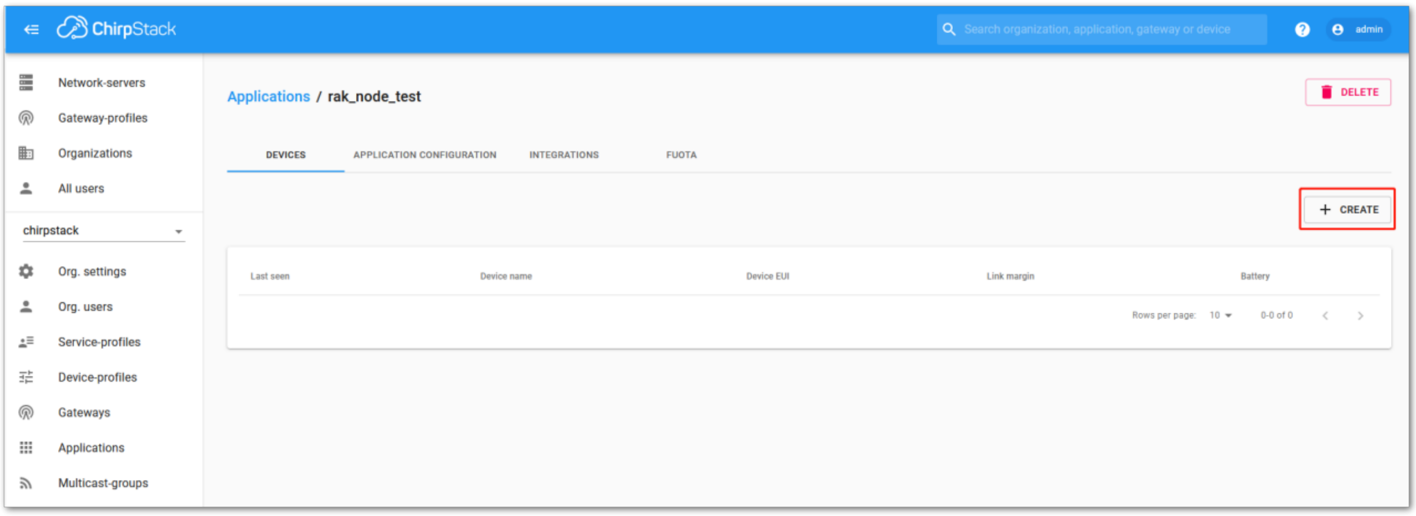


Last seen	Device name	Device EUI	Link margin	Battery

Rows per page: 10 0-0 of 0

Figure 210: Device Tab of an Application

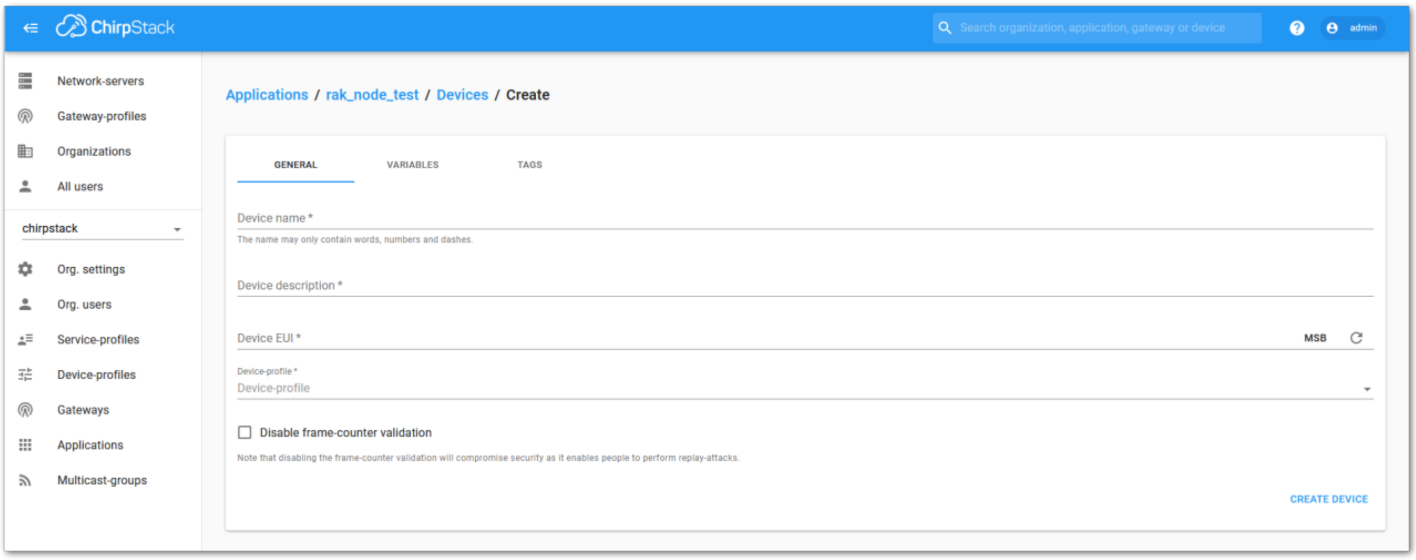
3. Once inside the **DEVICES** tab, create a new device (LoRaWAN node) by clicking on the "**+ CREATE**" button.



Last seen	Device name	Device EUI	Link margin	Battery

Rows per page: 10 0-0 of 0

Figure 211: Add a New Device



Applications / rak_node_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.

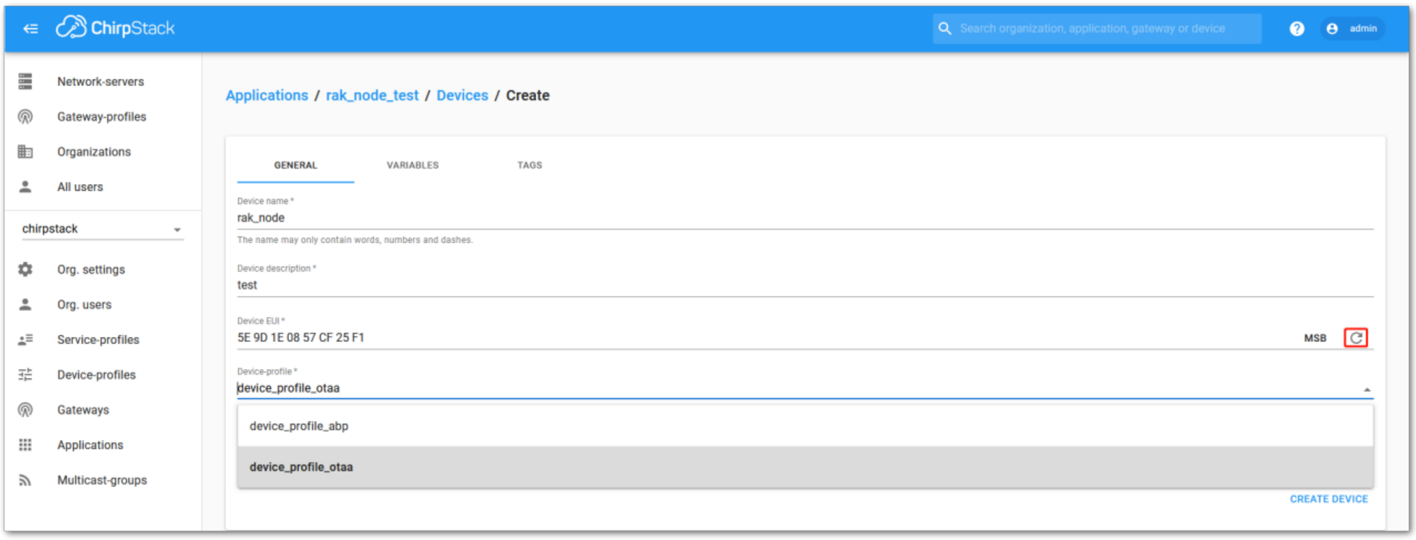
Figure 212: Chirpstack Adding Node into the RAK3172 Module

6. Once the node is created, fill in the necessary data. You can generate a Device EUI automatically by clicking the following icon, or you can write a correct Device EUI in the edit box.

Fill in the parameters requested:

- **Device name and Device description**: These are descriptive texts about your device.
- **Device EUI**: This interface allows you to generate a Device EUI automatically by clicking the generate icon. You can also add a specific Device EUI directly in the form.
- **Device Profile**:
  - If you want to join in OTAA mode, select **DeviceProfile_OTAA**.
  - If you want to join in ABP mode, select **DeviceProfile_ABP**.

> 📝 **NOTE:**
>
> Device profiles **DeviceProfile_OTAA** and **DeviceProfile_ABP** are only available if you are using the built-in Chirpstack LoRaWAN Server of RAK Gateways.
>
> If you have your own Chirpstack installation, you can set up the device profile with `LoRaWAN MAC version 1.0.3` and `LoRaWAN Regional Parameters revision B` to make it compatible with RAK3172.



Applications / rak_node_test / Devices / Create

Device name *
rak_node

The name may only contain words, numbers and dashes.

Device description *
test

Device EUI *
5E 9D 1E 08 57 CF 25 F1

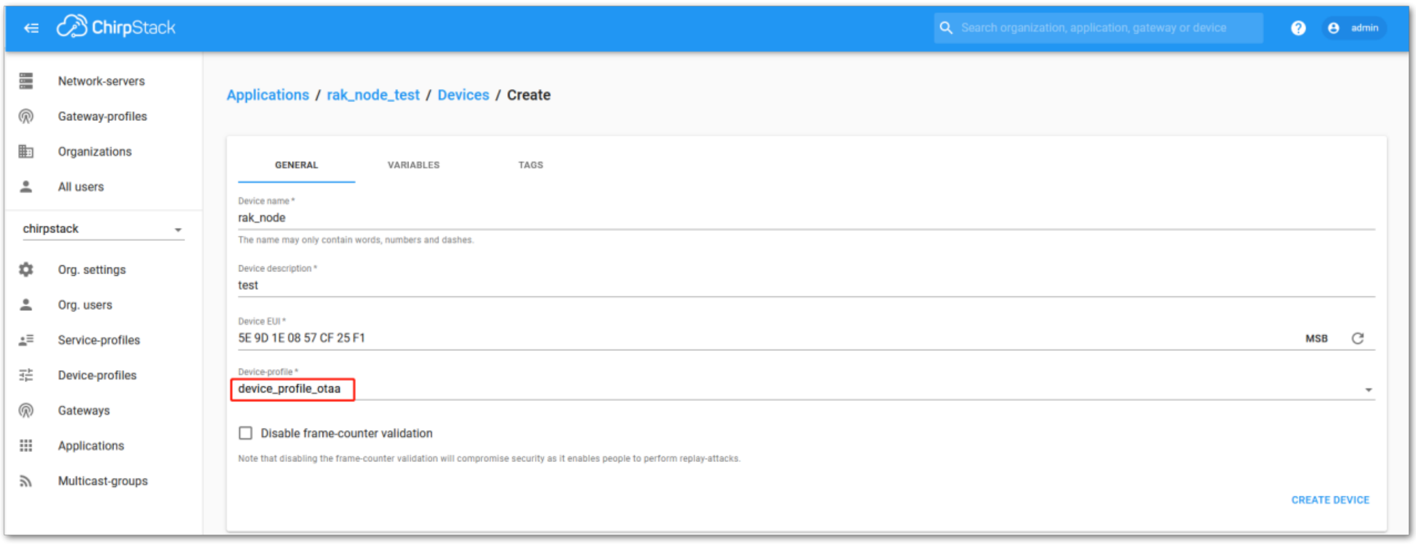
Device-profile *
device_profile_otaa

device_profile_abp
device_profile_otaa

Figure 213: Generate a New Device EUI

Chirpstack OTAA Device Registration

1. If you have selected **DeviceProfile_OTAA**, as shown in **Figure 214**, then after the device is created, an Application Key must be also created for this device.



Device name *
rak_node
The name may only contain words, numbers and dashes.

Device description *
test

Device EUI *
5E 9D 1E 08 57 CF 25 F1

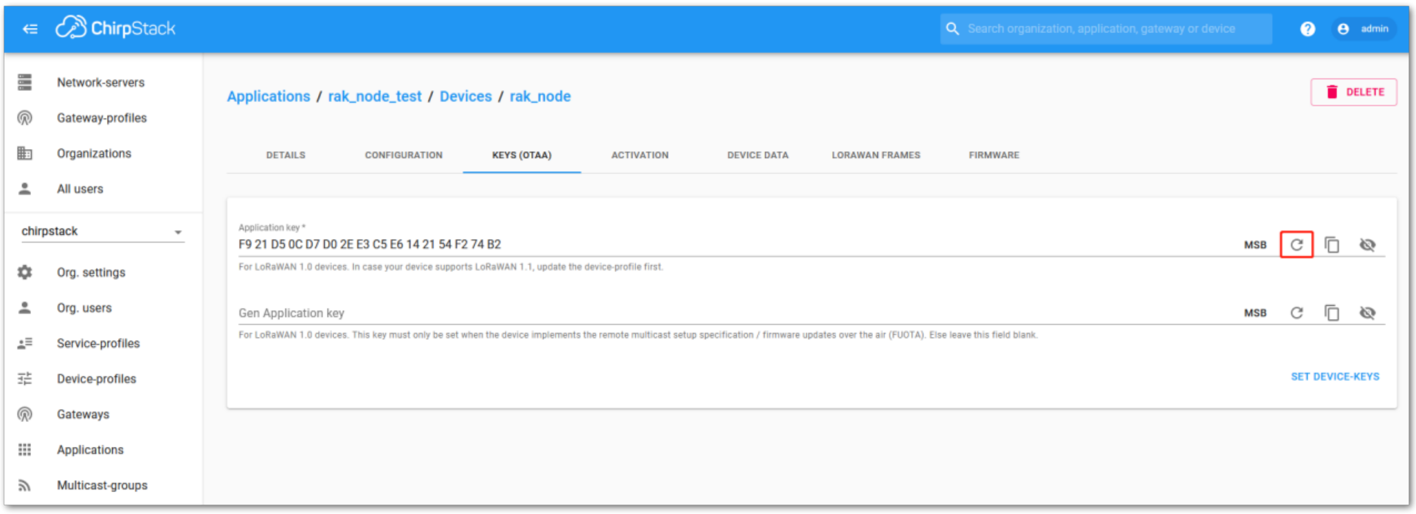
Device-profile *
device_profile_otaa

□ Disable frame-counter validation

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

Figure 214: Chirpstack OTAA Activation

2. A previously created Application Key can be entered here, or a new one can be generated automatically by clicking the icon highlighted in red in **Figure 215**.



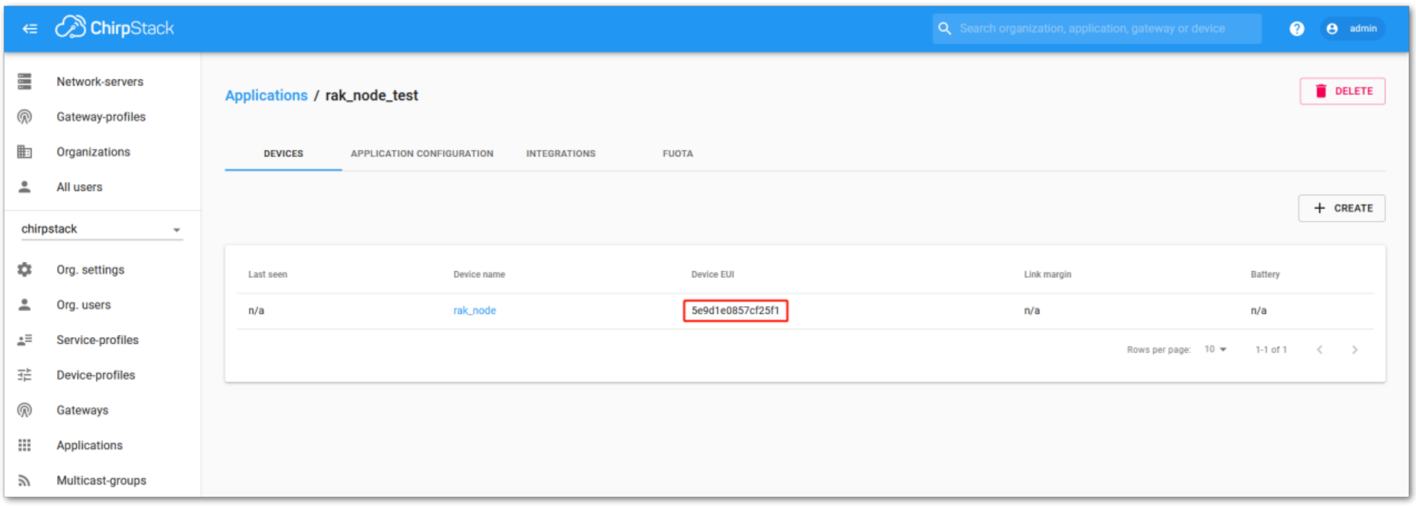
Application key *
F9 21 D5 0C D7 D0 2E E3 C5 E6 14 21 54 F2 74 B2
For LoRaWAN 1.0 devices. In case your device supports LoRaWAN 1.1, update the device-profile first.

Gen Application key
For LoRaWAN 1.0 devices. This key must only be set when the device implements the remote multicast setup specification / firmware updates over the air (FUOTA). Else leave this field blank.

Figure 215: Chirpstack OTAA Set Application Keys

3. Once the Application Key is added to the form, the process can be finalized by clicking on the **SET DEVICE-KEYS** button.

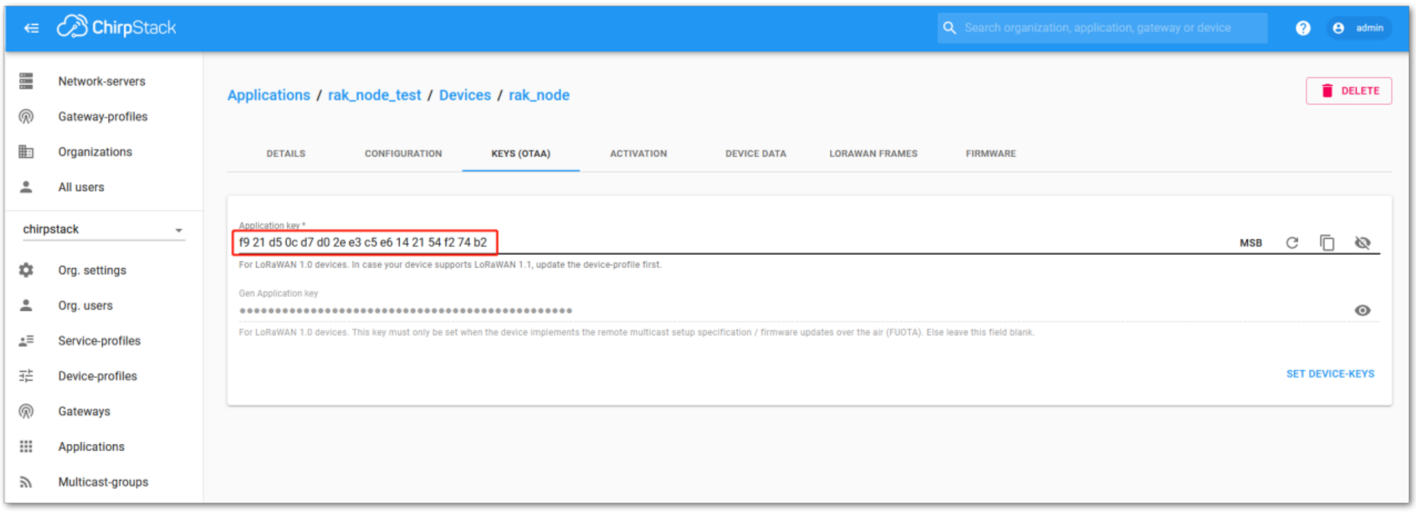
- As shown in **Figure 216**, a new device should be listed in the **DEVICES** tab. The most important parameters, such as the **Device EUI**, are shown in the summary.



Last seen	Device name	Device EUI	Link margin	Battery
n/a	rak_node	5e9d1e0857cf25f1	n/a	n/a

Figure 216: Chirpstack OTAA List of Device in the Device Tab

4. To end the process, it is a good practice to review that the Application Key is properly associated with this device. The Application Key can be verified in the **KEYS (OTAA)** tab, as shown in **Figure 217**.



Application key *
f9 21 d5 0c d7 d0 2e e3 c5 e6 14 21 54 f2 74 b2
For LoRaWAN 1.0 devices. In case your device supports LoRaWAN 1.1, update the device-profile first.

Gen Application key
••
For LoRaWAN 1.0 devices. This key must only be set when the device implements the remote multicast setup specification / firmware updates over the air (FUOTA). Else leave this field blank.

Figure 217: Application Key Associated with the New Device

> 📝 **NOTE:**
>
> Standard OTAA mode requires the **Device EUI**, **Application Key**, and **Application EUI**, but in ChirpStack's implementation, only the Device EUI and the Application Key are mandatory. The Application EUI is not required and not recorded in the Application tab. Nevertheless, you can reuse the Device EUI as the Application EUI during the configuration on the side of the node.

OTAA Configuration for Chirpstack

The RAK3172 module supports a series of AT commands to configure its internal parameters and control the functionalities of the module.

1. To set up the RAK3172 module to join the Chirpstack using OTAA, start by connecting the RAK3172 module to the Computer (see Figure 35) and open the RAK Serial Port Tool. Select the right COM port and set the baud rate to 115200.

It is recommended to start by testing the serial communication and verify that the current configuration is working by sending these two AT commands:

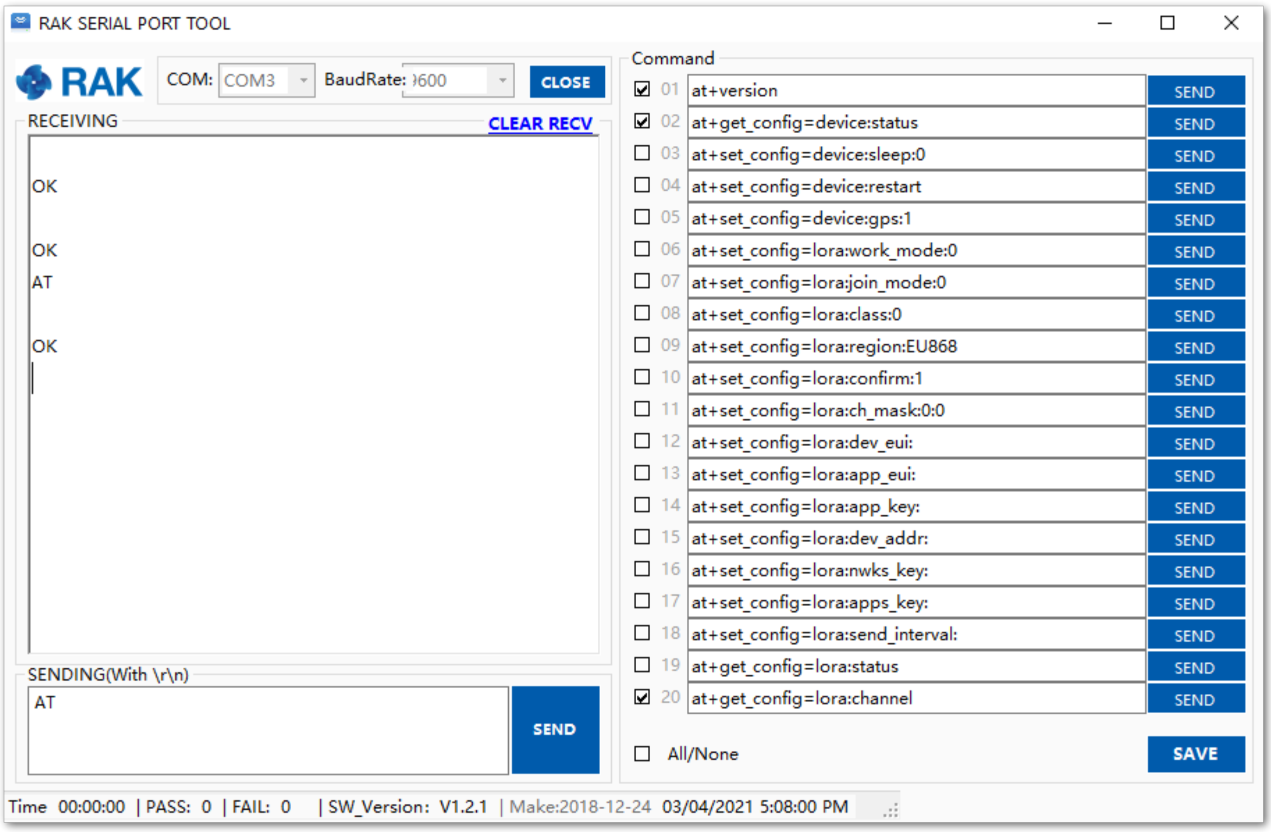
```
AT
```

```
ATE
```

`ATE` will echo the commands you input to the module, which is useful for tracking the commands and troubleshooting.

You will receive `OK` when you input the two commands. After setting `ATE`, you can now see all the commands you input together with the replies. Try again `AT` and you should see it on the terminal followed by `OK`, as shown in **Figure 218**.

> 📝 **NOTE:**
>
> If there is no `OK` or any reply, you need to check if the wiring of your UART lines is correct and if the baud is correctly configured to 115200. Also, you can check if the device is powered correctly. If you are getting power from a USB port, ensure that you have a good USB cable.



```
OK

OK
AT

OK
```

SENDING(With \r\n)

```
AT
```

☑	01	at+version	SEND
☑	02	at+get_config=device:status	SEND
□	03	at+set_config=device:sleep:0	SEND
□	04	at+set_config=device:restart	SEND
□	05	at+set_config=device:gps:1	SEND
□	06	at+set_config=lora:work_mode:0	SEND
□	07	at+set_config=lora:join_mode:0	SEND
□	08	at+set_config=lora:class:0	SEND
□	09	at+set_config=lora:region:EU868	SEND
□	10	at+set_config=lora:confirm:1	SEND
□	11	at+set_config=lora:ch_mask:0:0	SEND
□	12	at+set_config=lora:dev_eui:	SEND
□	13	at+set_config=lora:app_eui:	SEND
□	14	at+set_config=lora:app_key:	SEND
□	15	at+set_config=lora:dev_addr:	SEND
□	16	at+set_config=lora:nwks_key:	SEND
□	17	at+set_config=lora:apps_key:	SEND
□	18	at+set_config=lora:send_interval:	SEND
□	19	at+get_config=lora:status	SEND
☑	20	at+get_config=lora:channel	SEND

□ All/None SAVE

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 03/04/2021 5:08:00 PM

Figure 218: at+version command response

2. The next step is to configure the OTAA LoRaWAN parameters in RAK3172:

- LoRa work mode: **LoRaWAN**
- LoRaWAN join mode: **OTAA**
- LoRaWAN class: **Class A**
- LoRaWAN region: **EU868**

Set the work mode to LoRaWAN.

```
AT+NWM=1
```

Set the LoRaWAN activation to OTAA.

```
AT+NJM=1
```

Set the LoRaWAN class to Class A.

```
AT+CLASS=A
```

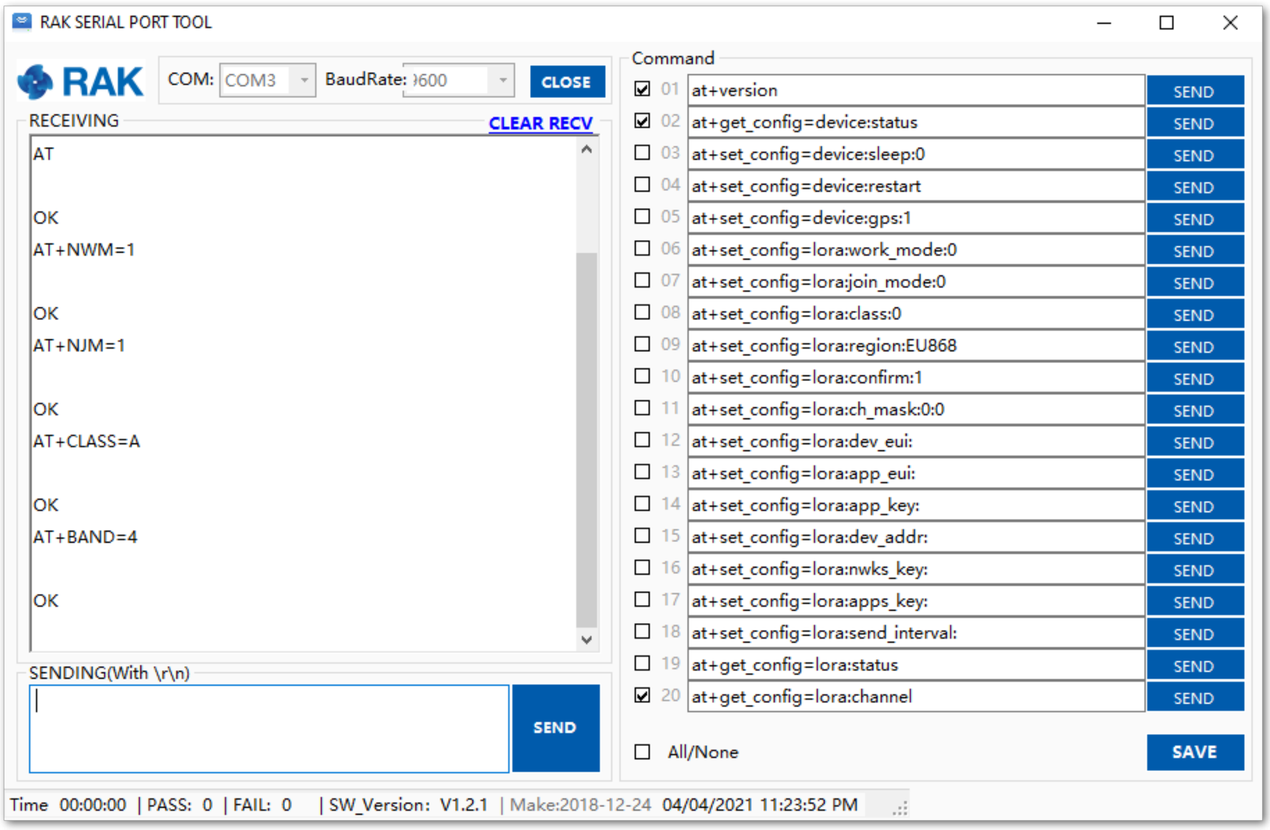
Set the frequency/region to EU868.

```
AT+BAND=4
```

> 📝 **NOTE:**
>
> Depending on the Regional Band you selected, you might need to configure the sub-band of your RAK3172 to match the gateway and LoRaWAN network server. This is especially important for Regional Bands like US915, AU915, and CN470.
>
> To configure the masking of channels for the sub-bands, you can use the `AT+MASK` command that can be found on the AT Command Manual .
>
> To illustrate, you can use sub-band 2 by sending the command `AT+MASK=0002` .

List of band parameter options

Code	Regional Band
0	EU433
1	CN470
2	RU864
3	IN865
4	EU868
5	US915
6	AU915
7	KR920
8	AS923-1
9	AS923-2
10	AS923-3
11	AS923-4



```
AT

OK
AT+NWM=1

OK
AT+NJM=1

OK
AT+CLASS=A

OK
AT+BAND=4

OK
```

☑ 01 at+version SEND
☑ 02 at+get_config=device:status SEND
□ 03 at+set_config=device:sleep:0 SEND
□ 04 at+set_config=device:restart SEND
□ 05 at+set_config=device:gps:1 SEND
□ 06 at+set_config=lora:work_mode:0 SEND
□ 07 at+set_config=lora:join_mode:0 SEND
□ 08 at+set_config=lora:class:0 SEND
□ 09 at+set_config=lora:region:EU868 SEND
□ 10 at+set_config=lora:confirm:1 SEND
□ 11 at+set_config=lora:ch_mask:0:0 SEND
□ 12 at+set_config=lora:dev_eui: SEND
□ 13 at+set_config=lora:app_eui: SEND
□ 14 at+set_config=lora:app_key: SEND
□ 15 at+set_config=lora:dev_addr: SEND
□ 16 at+set_config=lora:nwks_key: SEND
□ 17 at+set_config=lora:apps_key: SEND
□ 18 at+set_config=lora:send_interval: SEND
□ 19 at+get_config=lora:status SEND
☑ 20 at+get_config=lora:channel SEND

□ All/None

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 04/04/2021 11:23:52 PM

Figure 219: Configuring LoRa Parameters

3. After the configuration of the LoRaWAN parameters, the next step is to set up the DevEUI and AppKey. You need the use the values from the Chirpstack device console.

> 📝 **NOTE:**
>
> The Application EUI parameter is not required in the ChirpStack platform; therefore, it is possible to use the same id as the Device EUI.

- Device EUI: **5E9D1E0857CF25F1**
- Application EUI: **5E9D1E0857CF25F1**
- Application Key: **F921D50CD7D02EE3C5E6142154F274B2**

Set the Device EUI.

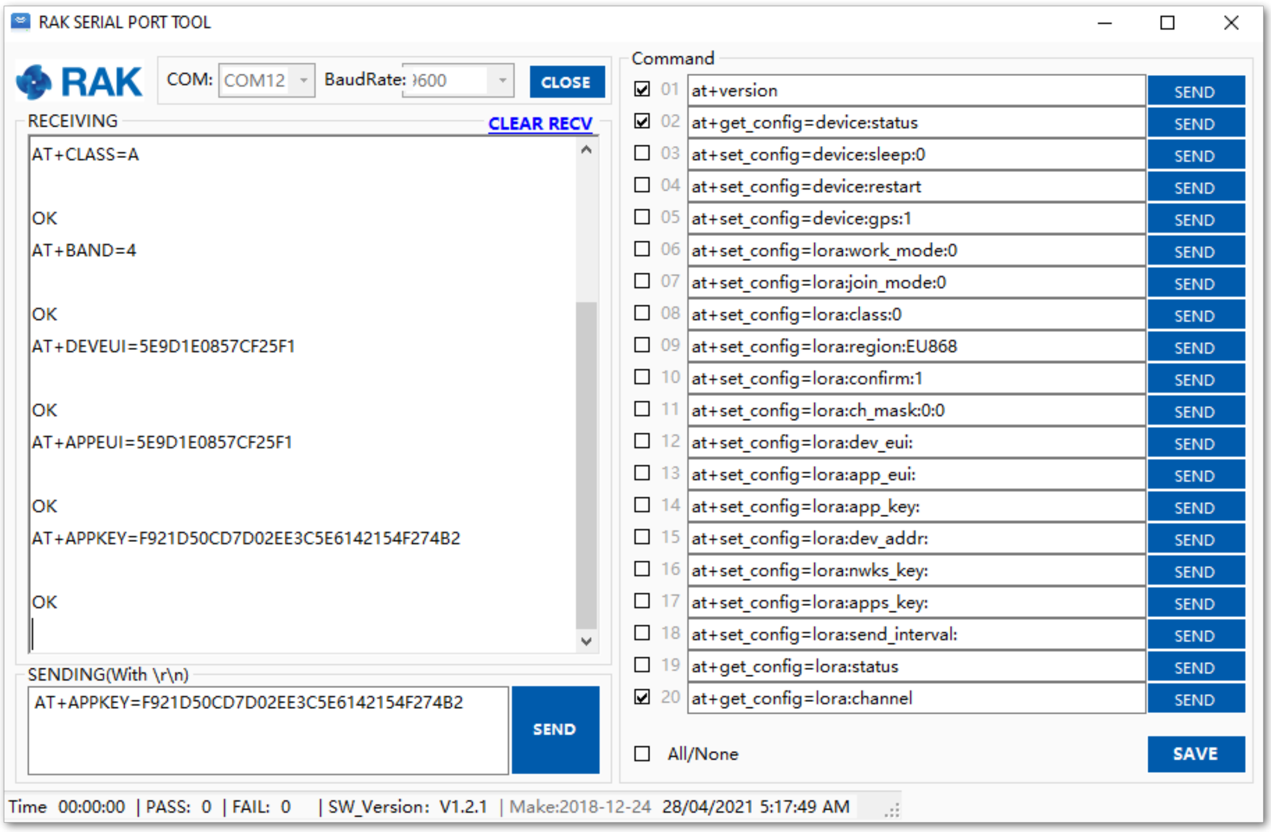
```
AT+DEVEUI=5E9D1E0857CF25F1
```

Set the Application EUI.

```
AT+APPEUI=5E9D1E0857CF25F1
```

Set the Application Key.

```
AT+APPKEY=F921D50CD7D02EE3C5E6142154F274B2
```



RAK SERIAL PORT TOOL

COM: COM12 BaudRate:)600 CLOSE

RECEIVING CLEAR RECV

```
AT+CLASS=A

OK
AT+BAND=4

OK
AT+DEVEUI=5E9D1E0857CF25F1

OK
AT+APPEUI=5E9D1E0857CF25F1

OK
AT+APPKEY=F921D50CD7D02EE3C5E6142154F274B2

OK
```

SENDING(With \r\n)

```
AT+APPKEY=F921D50CD7D02EE3C5E6142154F274B2
```

SEND

Command

☑	01	at+version	SEND
☑	02	at+get_config=device:status	SEND
□	03	at+set_config=device:sleep:0	SEND
□	04	at+set_config=device:restart	SEND
□	05	at+set_config=device:gps:1	SEND
□	06	at+set_config=lora:work_mode:0	SEND
□	07	at+set_config=lora:join_mode:0	SEND
□	08	at+set_config=lora:class:0	SEND
□	09	at+set_config=lora:region:EU868	SEND
□	10	at+set_config=lora:confirm:1	SEND
□	11	at+set_config=lora:ch_mask:0:0	SEND
□	12	at+set_config=lora:dev_eui:	SEND
□	13	at+set_config=lora:app_eui:	SEND
□	14	at+set_config=lora:app_key:	SEND
□	15	at+set_config=lora:dev_addr:	SEND
□	16	at+set_config=lora:nwks_key:	SEND
□	17	at+set_config=lora:apps_key:	SEND
□	18	at+set_config=lora:send_interval:	SEND
□	19	at+get_config=lora:status	SEND
☑	20	at+get_config=lora:channel	SEND

□ All/None SAVE

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 28/04/2021 5:17:49 AM

Figure 220: Configuring LoRa Parameters

4. After EUI and key configuration, the device can now join the network and send some payload.

```
AT+JOIN=1:0:10:8
```

> 📝 **NOTE:**
>
> `AT+JOIN` command parameters are optional. You can configure the settings for auto-join, reattempt interval, and the number of join attempts if your application needs it. If not configured, it will use the default parameter values.
>
> `AT+JOIN` and `AT+JOIN=1` also share the common functionality of trying to join the network.

Join command format: **`AT+JOIN=w:x:y:z`**

Parameter	Description
w	Join command - 1: joining, 0: stop joining.
x	Auto-join config - 1: auto-join on power-up, 0: no auto-join
y	Reattempt interval in seconds (7-255) - 8 is the default.
z	Number of join attempts (0-255) - 0 is the default.

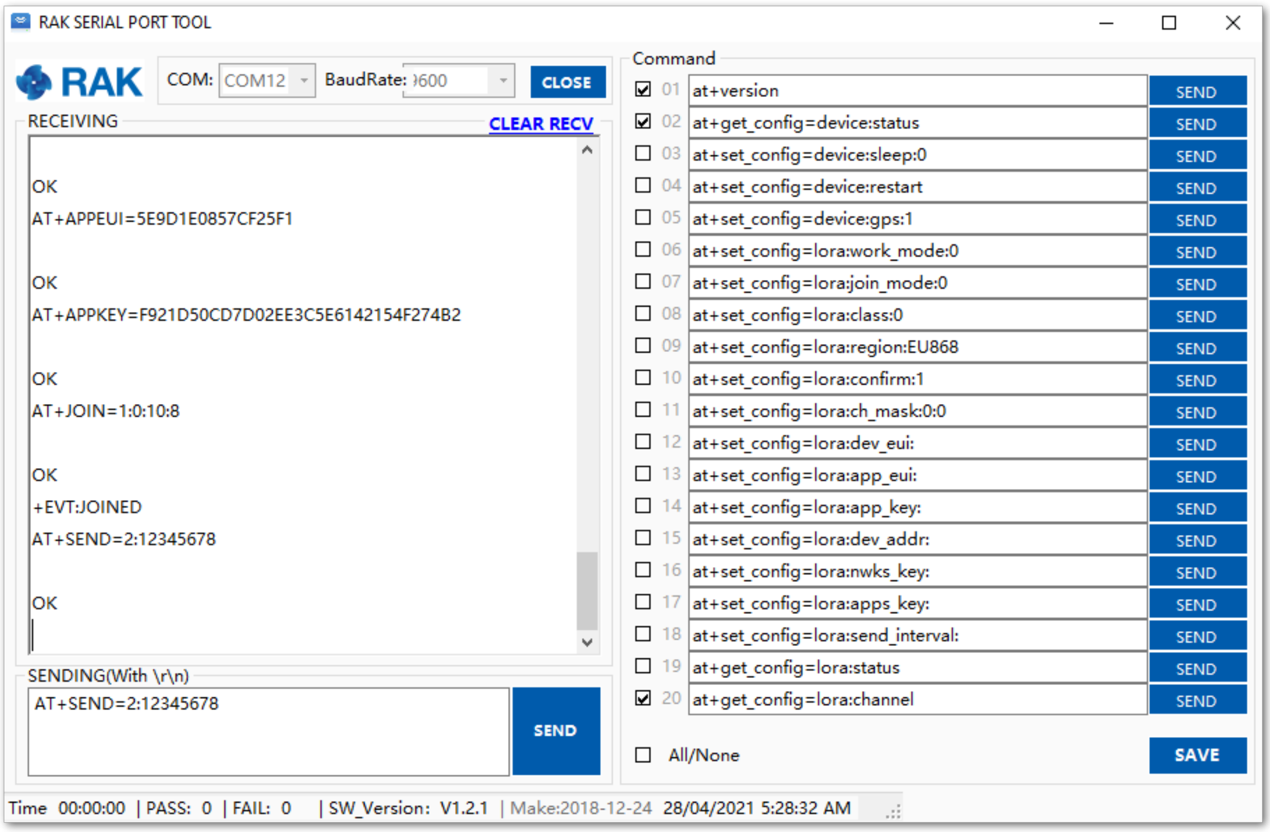
After 5 or 6 seconds, if the request is successfully received by a LoRa gateway, you should see the JOINED status reply.

> 📝 **NOTE:**
>
> If the OTAA device failed to join, you need to check if your device is within reach of a working LoRaWAN gateway that is configured to connect to Chirpstack. It is also important to check that all your OTAA parameters (DEVEUI and APPKEY) are correct, using the `AT+DEVEUI=?` and `AT+APPKEY=?` commands. Lastly, ensure that the antenna of your device is properly connected.
>
> After checking all the things above, try to join again.

5. With the end-device properly activated, you can now try to send some payload after a successful join.

```
AT+SEND=2:12345678
```

Send command format: **`AT+SEND=<port>:<payload>`**



```
OK
AT+APPEUI=5E9D1E0857CF25F1

OK
AT+APPKEY=F921D50CD7D02EE3C5E6142154F274B2

OK
AT+JOIN=1:0:10:8

OK
+EVT:JOINED
AT+SEND=2:12345678

OK
```

SENDING(With \r\n): AT+SEND=2:12345678

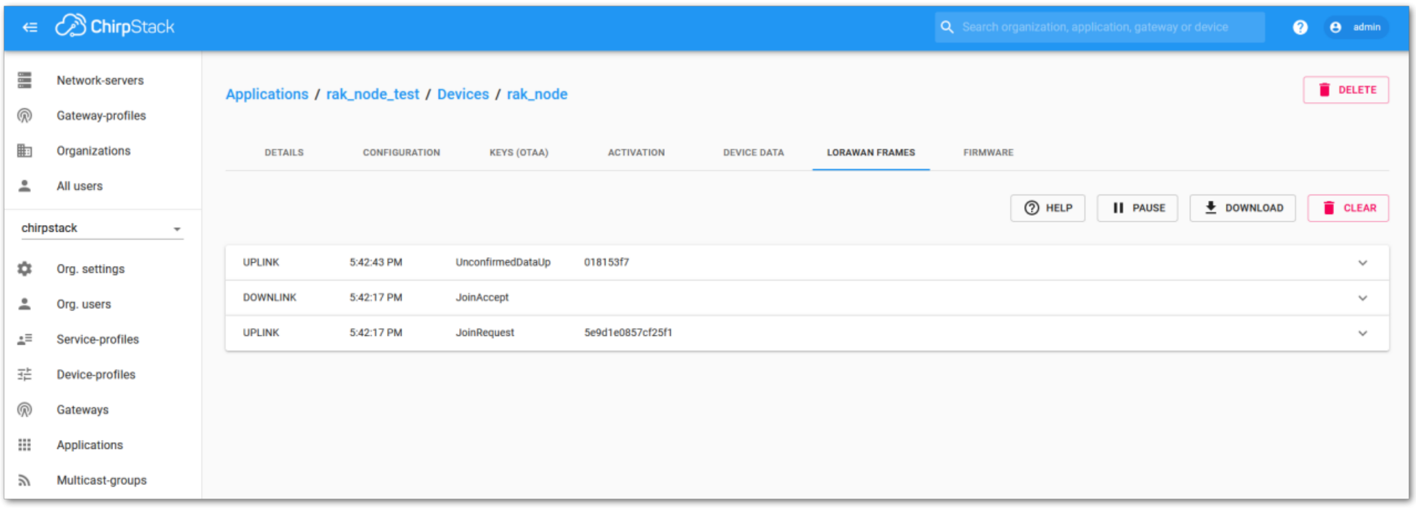
		Command
☑	01	at+version
☑	02	at+get_config=device:status
☐	03	at+set_config=device:sleep:0
☐	04	at+set_config=device:restart
☐	05	at+set_config=device:gps:1
☐	06	at+set_config=lora:work_mode:0
☐	07	at+set_config=lora:join_mode:0
☐	08	at+set_config=lora:class:0
☐	09	at+set_config=lora:region:EU868
☐	10	at+set_config=lora:confirm:1
☐	11	at+set_config=lora:ch_mask:0:0
☐	12	at+set_config=lora:dev_eui:
☐	13	at+set_config=lora:app_eui:
☐	14	at+set_config=lora:app_key:
☐	15	at+set_config=lora:dev_addr:
☐	16	at+set_config=lora:nwks_key:
☐	17	at+set_config=lora:apps_key:
☐	18	at+set_config=lora:send_interval:
☐	19	at+get_config=lora:status
☑	20	at+get_config=lora:channel

☐ All/None

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 28/04/2021 5:28:32 AM

Figure 221: OTAA Test Sample Data Sent via RAK Serial Port Tool

On the ChirpStack platform, you should see the join and uplink messages in the **LORAWAN FRAMES** tab, as shown in **Figure 222**. By convention, messages sent from nodes to gateways are considered as **Uplinks** while messages sent by gateways to nodes are considered as **Downlinks**.



Applications / rak_node_test / Devices / rak_node

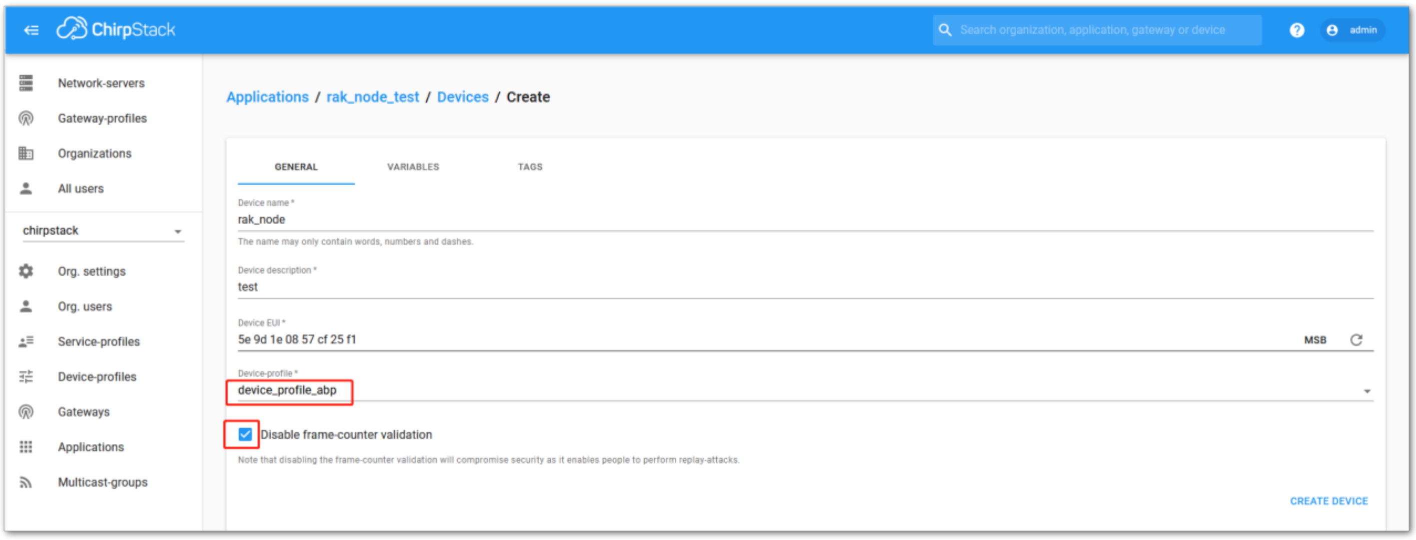
UPLINK	5:42:43 PM	UnconfirmedDataUp	018153f7
DOWNLINK	5:42:17 PM	JoinAccept	
UPLINK	5:42:17 PM	JoinRequest	5e9d1e0857cf25f1

Figure 222: Chirpstack Data Received Preview

Chirpstack ABP Device Registration

1. During the registration of a new device, if you select **DeviceProfile_ABP**, as shown in **Figure 223**, then the ChirpStack platform will assume that this device will join the LoRaWAN network using the ABP mode.

> 📝 **NOTE:**
>
> Tick the checkbox **Disable counting frame verification**. During the test, when the module is restarted, the frame counting number will also be restarted from zero. This would cause a synchronization problem with the ChirpStack server treating it as a replay attack. For testing purposes, it is safe to disable this feature, but remember to activate it in a production environment.



Applications / rak_node_test / Devices / Create

Device name *
rak_node
The name may only contain words, numbers and dashes.

Device description *
test

Device EUI *
5e 9d 1e 08 57 cf 25 f1

Device-profile *
device_profile_abp

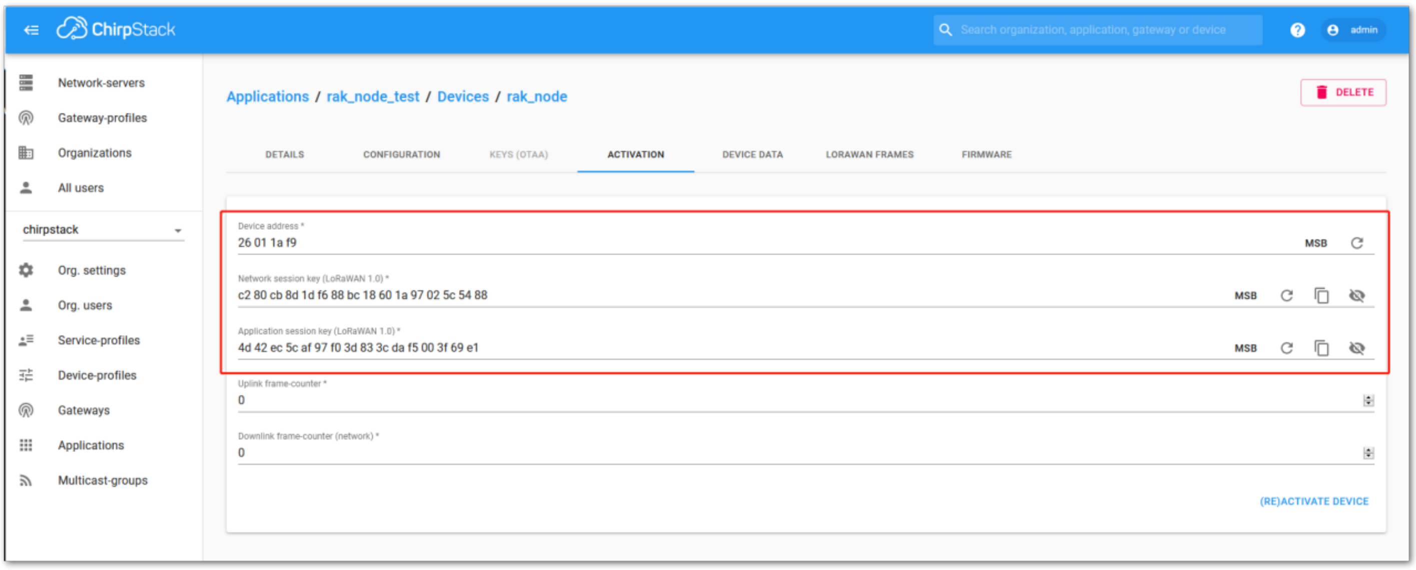
☑ Disable frame-counter validation
Note that disabling the frame-counter validation will compromise security as it enables people to perform replay-attacks.

Figure 223: ChirpStack Console, Configuring a Device

2. After selecting the ABP mode, the following parameters appear in the **ACTIVATION** tab:

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

- **Device address**
- **Network Session Key**
- **Application Session Key**



Applications / rak_node_test / Devices / rak_node

Device address *
26 01 1a f9

Network session key (LoRaWAN 1.0) *
c2 80 cb 8d 1d f6 88 bc 18 60 1a 97 02 5c 54 88

Application session key (LoRaWAN 1.0) *
4d 42 ec 5c af 97 f0 3d 83 3c da f5 00 3f 69 e1

Uplink frame-counter *
0

Downlink frame-counter (network) *
0

Figure 224: Chirpstack ABP Activation Parameters Needed

3. The parameters can be generated as random numbers by the platform or can be set with user values. Once these parameters are filled in properly, the process is completed by clicking on the **ACTIVATE DEVICE** button.

ABP Configuration for Chirpstack

1. To set up the RAK3172 module to join the Chirpstack using ABP, start by connecting the RAK3172 module to the Computer (see Figure 35) and open the RAK Serial Port Tool. Select the right COM port and set the baud rate to 115200.

It is recommended to start by testing the serial communication and verify that the current configuration is working by sending these two AT commands:

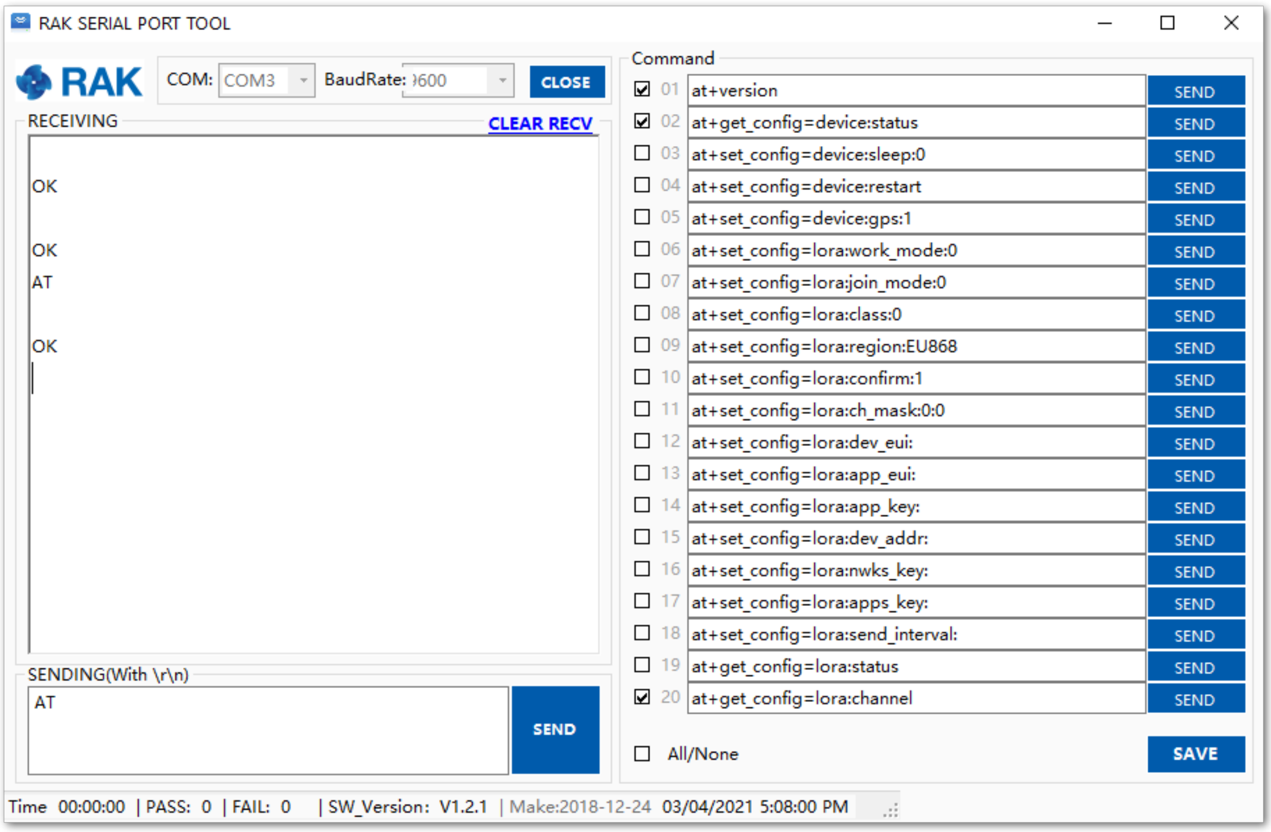
```
AT
```

```
ATE
```

`ATE` will echo the commands you input to the module, which is useful for tracking the commands and troubleshooting.

You will receive `OK` when you input the two commands. After setting `ATE`, you can now see all the commands you input together with the replies. Try again `AT` and you should see it on the terminal followed by `OK`, as shown in **Figure 225**.

> 📝 **NOTE:**
>
> If there is no `OK` or any reply, you need to check if the wiring of your UART lines is correct and if the baud is correctly configured to 115200. Also, you can check if the device is powered correctly. If you are getting power from a USB port, ensure that you have a good USB cable.



COM: COM3 BaudRate: 9600

RECEIVING:

```
OK

OK
AT

OK
```

SENDING(With \r\n):

```
AT
```

Command:

```
☑ 01 at+version
☑ 02 at+get_config=device:status
☐ 03 at+set_config=device:sleep:0
☐ 04 at+set_config=device:restart
☐ 05 at+set_config=device:gps:1
☐ 06 at+set_config=lora:work_mode:0
☐ 07 at+set_config=lora:join_mode:0
☐ 08 at+set_config=lora:class:0
☐ 09 at+set_config=lora:region:EU868
☐ 10 at+set_config=lora:confirm:1
☐ 11 at+set_config=lora:ch_mask:0:0
☐ 12 at+set_config=lora:dev_eui:
☐ 13 at+set_config=lora:app_eui:
☐ 14 at+set_config=lora:app_key:
☐ 15 at+set_config=lora:dev_addr:
☐ 16 at+set_config=lora:nwks_key:
☐ 17 at+set_config=lora:apps_key:
☐ 18 at+set_config=lora:send_interval:
☐ 19 at+get_config=lora:status
☑ 20 at+get_config=lora:channel
```

☐ All/None

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 03/04/2021 5:08:00 PM

Figure 225: at+version command response

2. The next step is to configure the ABP LoRaWAN parameters in RAK3172:

- LoRa work mode: **LoRaWAN**
- LoRaWAN join mode: **ABP**
- LoRaWAN class: **Class A**
- LoRaWAN region: **EU868**

Set the work mode to LoRaWAN. It can be set to P2P as well but by default, the device is in LoRaWAN mode.

```
AT+NWM=1
```

Set the LoRaWAN activation to ABP.

```
AT+NJM=0
```

Set the LoRaWAN class to Class A.

```
AT+CLASS=A
```

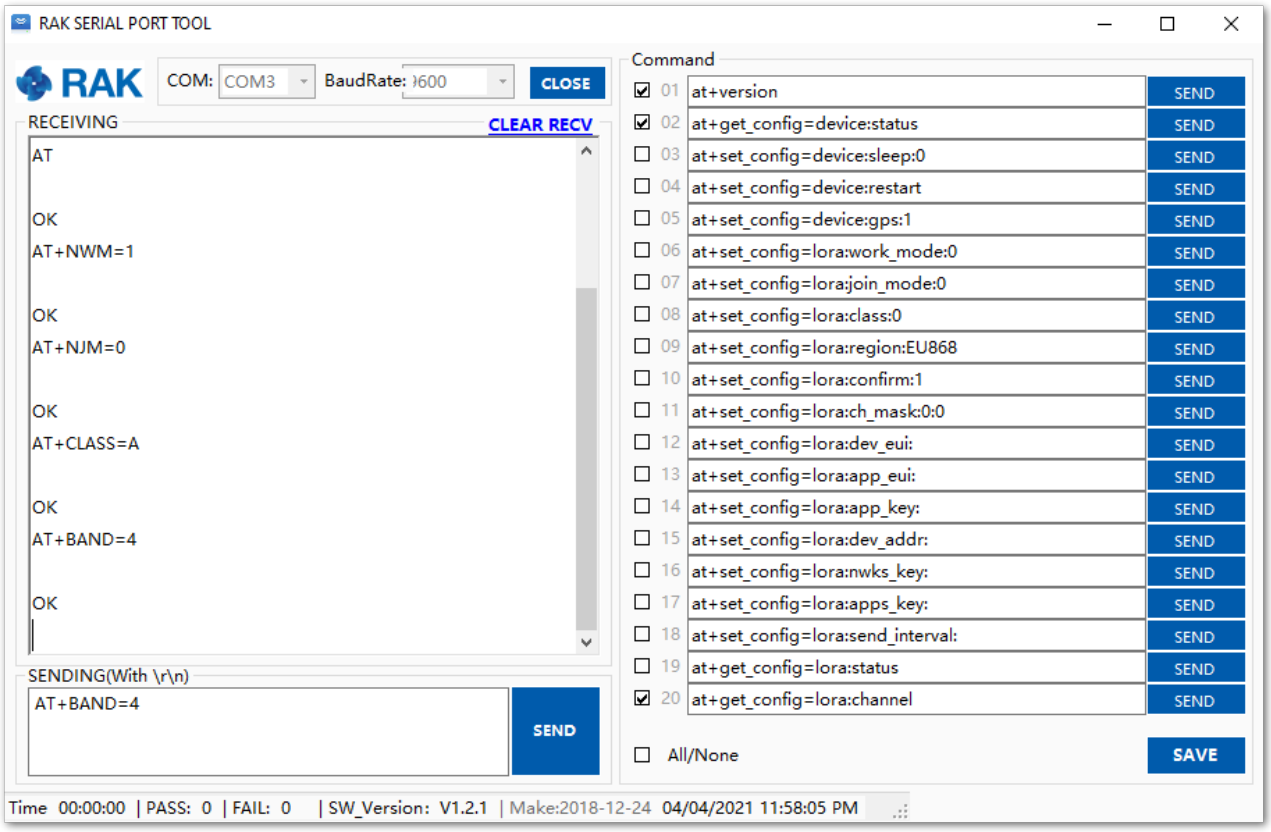
Set the frequency/region to EU868.

```
AT+BAND=4
```

> 📝 **NOTE:**
>
> Depending on the Regional Band you selected, you might need to configure the sub-band of your RAK3172 to match the gateway and LoRaWAN network server. This is especially important on Regional Bands like US915, AU915, and CN470.
>
> To configure the masking of channels for the sub-bands, you can use the AT+MASK command that can be found on the AT Commands Manual.
>
> To illustrate, you can use sub-band 2 by sending the command `AT+MASK=0002`.

List of band parameter options

Code	Regional Band
0	EU433
1	CN470
2	RU864
3	IN865
4	EU868
5	US915
6	AU915
7	KR920
8	AS923-1
9	AS923-2
10	AS923-3
11	AS923-4



```
AT

OK
AT+NWM=1

OK
AT+NJM=0

OK
AT+CLASS=A

OK
AT+BAND=4

OK
```

SENDING(With \r\n): AT+BAND=4

☑ 01 at+version
☑ 02 at+get_config=device:status
☐ 03 at+set_config=device:sleep:0
☐ 04 at+set_config=device:restart
☐ 05 at+set_config=device:gps:1
☐ 06 at+set_config=lora:work_mode:0
☐ 07 at+set_config=lora:join_mode:0
☐ 08 at+set_config=lora:class:0
☐ 09 at+set_config=lora:region:EU868
☐ 10 at+set_config=lora:confirm:1
☐ 11 at+set_config=lora:ch_mask:0:0
☐ 12 at+set_config=lora:dev_eui:
☐ 13 at+set_config=lora:app_eui:
☐ 14 at+set_config=lora:app_key:
☐ 15 at+set_config=lora:dev_addr:
☐ 16 at+set_config=lora:nwks_key:
☐ 17 at+set_config=lora:apps_key:
☐ 18 at+set_config=lora:send_interval:
☐ 19 at+get_config=lora:status
☑ 20 at+get_config=lora:channel

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 04/04/2021 11:58:05 PM

Figure 226: Configuring LoRa Parameters

3. After the configuration of the LoRaWAN parameters, the next step is to set up the device address and session keys. You need the use the values from the TTN device console.

- Device Address: **26011AF9**
- Application Session Key: **4D42EC5CAF97F03D833CDAf5003F69E1**

- Network Session Key: **C280CB8D1DF688BC18601A97025C5488**

Set the Device Address.

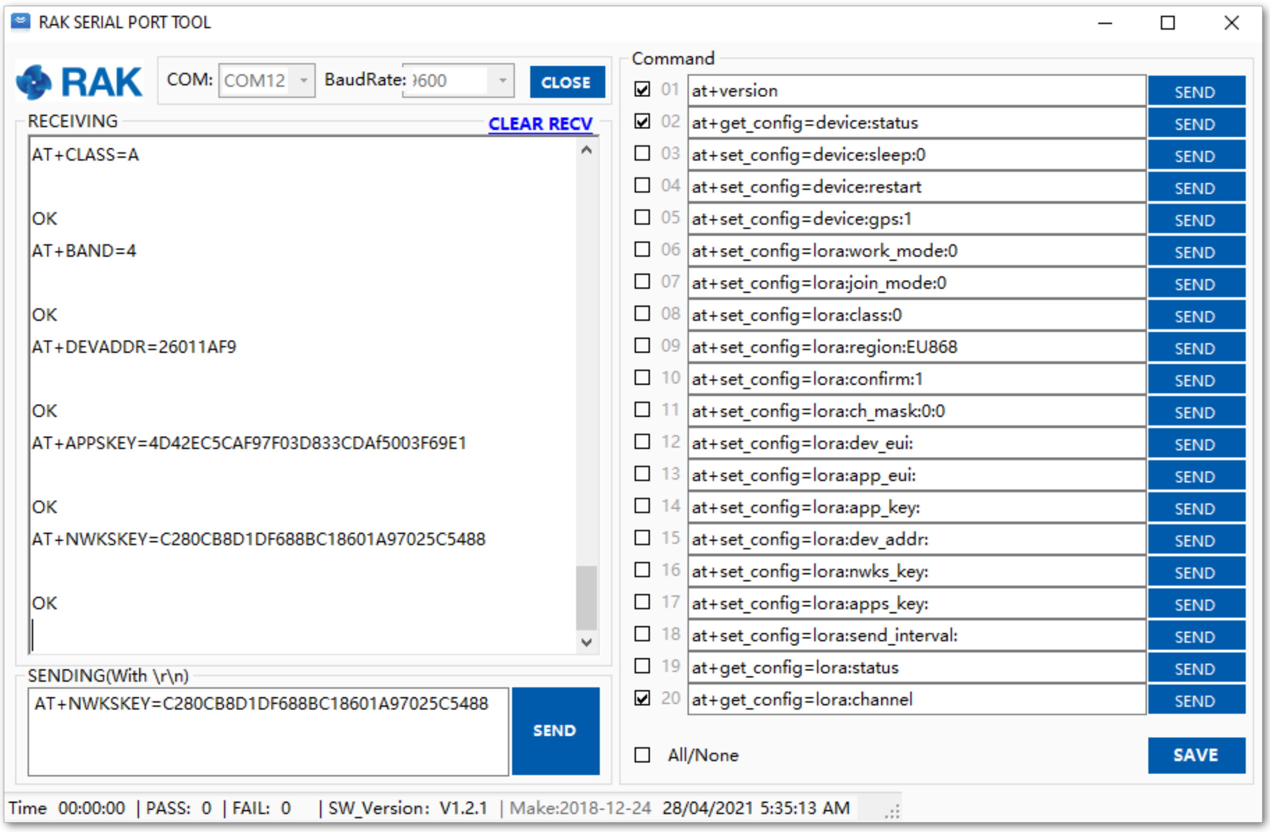
```
AT+DEVADDR=26011AF9
```

Set the Application Session Key.

```
AT+APPSKEY=4D42EC5CAF97F03D833CDAf5003F69E1
```

Set the Network Session Key.

```
AT+NWKSKEY=C280CB8D1DF688BC18601A97025C5488
```



```
AT+CLASS=A

OK
AT+BAND=4

OK
AT+DEVADDR=26011AF9

OK
AT+APPSKEY=4D42EC5CAF97F03D833CDAf5003F69E1

OK
AT+NWKSKEY=C280CB8D1DF688BC18601A97025C5488

OK
```

Sending: AT+NWKSKEY=C280CB8D1DF688BC18601A97025C5488

☑ 01 at+version SEND
☑ 02 at+get_config=device:status SEND
☐ 03 at+set_config=device:sleep:0 SEND
☐ 04 at+set_config=device:restart SEND
☐ 05 at+set_config=device:gps:1 SEND
☐ 06 at+set_config=lora:work_mode:0 SEND
☐ 07 at+set_config=lora:join_mode:0 SEND
☐ 08 at+set_config=lora:class:0 SEND
☐ 09 at+set_config=lora:region:EU868 SEND
☐ 10 at+set_config=lora:confirm:1 SEND
☐ 11 at+set_config=lora:ch_mask:0:0 SEND
☐ 12 at+set_config=lora:dev_eui: SEND
☐ 13 at+set_config=lora:app_eui: SEND
☐ 14 at+set_config=lora:app_key: SEND
☐ 15 at+set_config=lora:dev_addr: SEND
☐ 16 at+set_config=lora:nwks_key: SEND
☐ 17 at+set_config=lora:apps_key: SEND
☐ 18 at+set_config=lora:send_interval: SEND
☐ 19 at+get_config=lora:status SEND
☑ 20 at+get_config=lora:channel SEND

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 28/04/2021 5:35:13 AM

Figure 227: Configuring LoRa Parameters

After EUI and keys configuration, the device can now join the network and send some payload.

```
AT+JOIN=1:0:10:8
```

> 📝 **NOTE:**
>
> `AT+JOIN` command parameters are optional. You can configure the settings for auto-join, reattempt interval, and the number of join attempts if your application needs it. If not configured, it will use the default parameter values.
>
> `AT+JOIN` and `AT+JOIN=1` also share the common functionality of trying to join the network.

Join command format: **`AT+JOIN=w:x:y:z`**

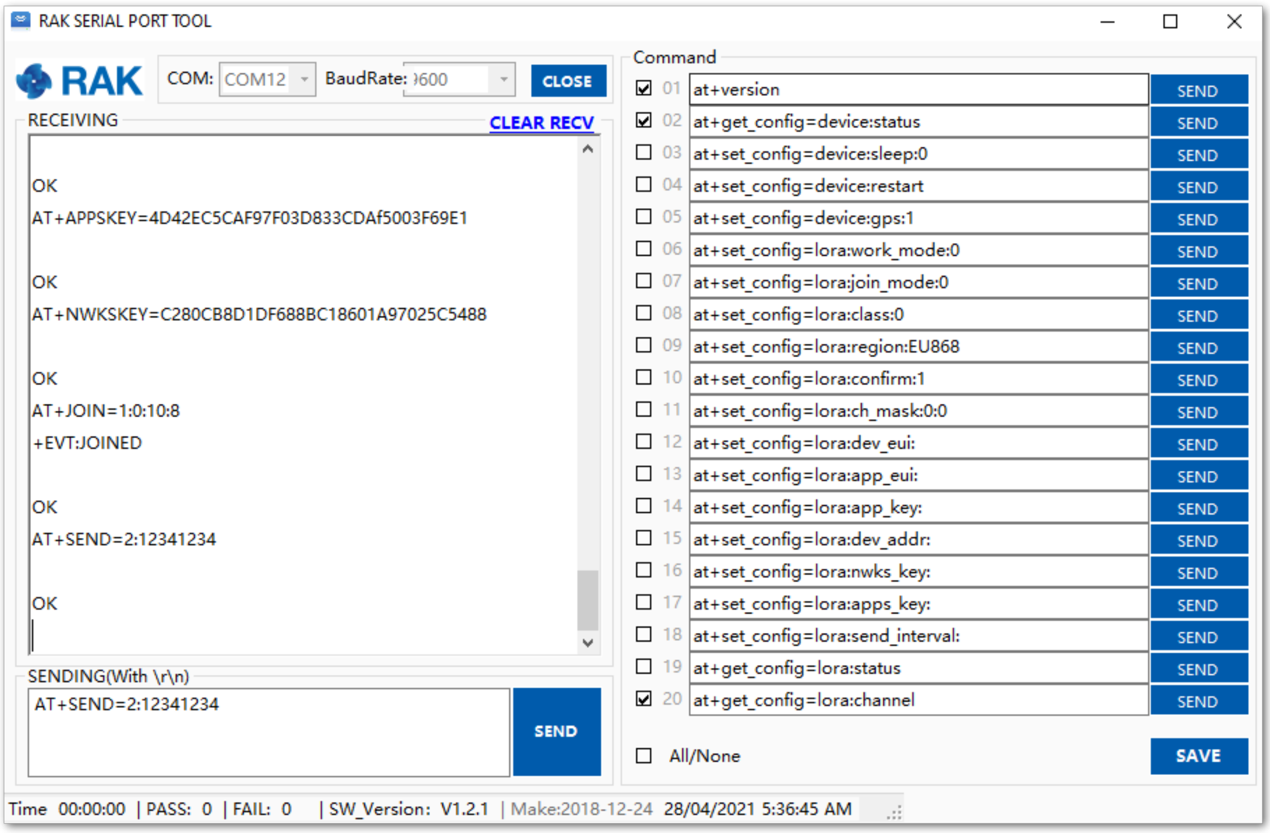
Parameter	Description
w	Join command - 1: joining, 0: stop joining.
x	Auto-join config - 1: auto-join on power-up, 0: no auto-join
y	Reattempt interval in seconds (7-255) - 8 is the default.
z	Number of join attempts (0-255) - 0 is the default.

4. After 5 or 6 seconds, if the request is successfully received by a LoRa gateway, then you should see the JOINED status reply.

5. You can now try to send some payload after a successful join.

```
AT+SEND=2:12341234
```

Send command format: `AT+SEND=<port>:<payload>`



```
OK
AT+APPSKEY=4D42EC5CAF97F03D833CDAf5003F69E1

OK
AT+NWKSKEY=C280CB8D1DF688BC18601A97025C5488

OK
AT+JOIN=1:0:10:8
+EVT:JOINED

OK
AT+SEND=2:12341234

OK
```

AT+SEND=2:12341234

☑ 01 at+version
☑ 02 at+get_config=device:status
☐ 03 at+set_config=device:sleep:0
☐ 04 at+set_config=device:restart
☐ 05 at+set_config=device:gps:1
☐ 06 at+set_config=lora:work_mode:0
☐ 07 at+set_config=lora:join_mode:0
☐ 08 at+set_config=lora:class:0
☐ 09 at+set_config=lora:region:EU868
☐ 10 at+set_config=lora:confirm:1
☐ 11 at+set_config=lora:ch_mask:0:0
☐ 12 at+set_config=lora:dev_eui:
☐ 13 at+set_config=lora:app_eui:
☐ 14 at+set_config=lora:app_key:
☐ 15 at+set_config=lora:dev_addr:
☐ 16 at+set_config=lora:nwks_key:
☐ 17 at+set_config=lora:apps_key:
☐ 18 at+set_config=lora:send_interval:
☐ 19 at+get_config=lora:status
☑ 20 at+get_config=lora:channel

☐ All/None

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12-24 28/04/2021 5:36:45 AM

Figure 228: ABP Test Sample Data Sent via RAK Serial Port Tool

LoRa P2P Mode

This section will show you how to set up and connect two RAK3172 units to work in the LoRa P2P mode. The configuration of the RAK3172 units is done by connecting the two modules to a general-purpose computer using a USB-UART converter. The setup of each RAK3172 can be done separately, but testing the LoRa P2P mode will require having both units connected simultaneously. This could be done by having one computer with two USB ports or two computers with one USB port each.

It is recommended to start by testing the serial communication and verify the current configuration is working by sending these two AT commands:

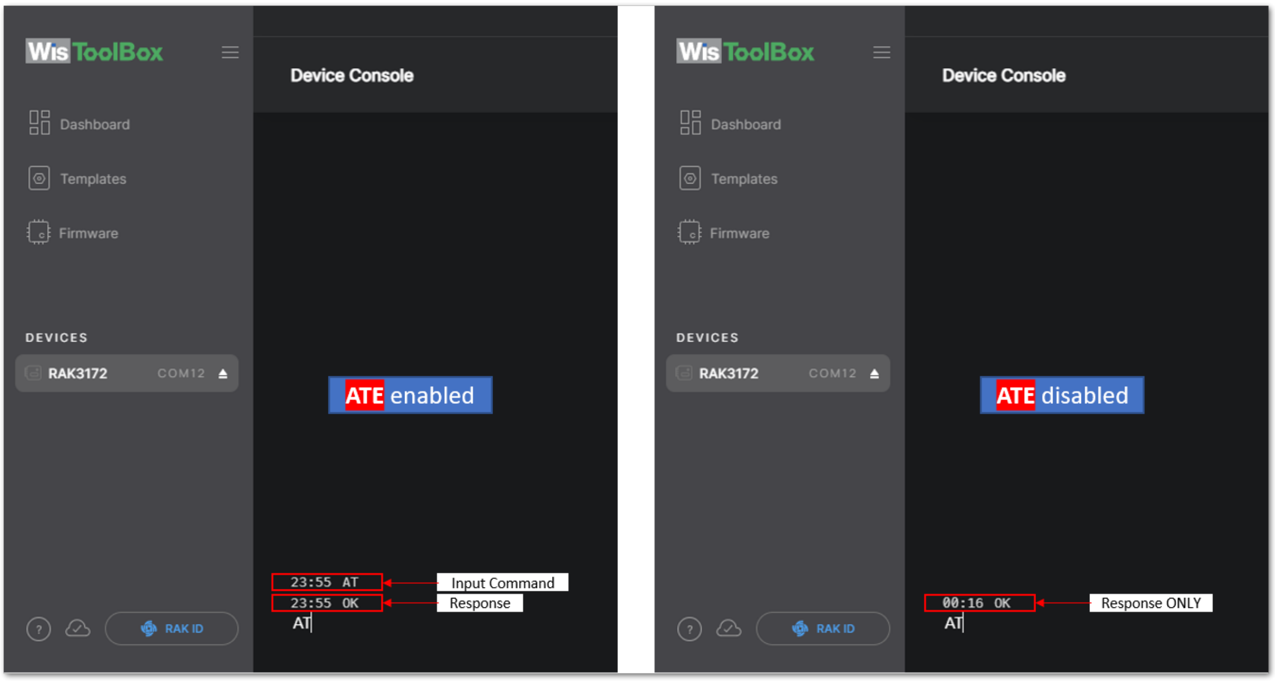
```
AT
```

```
ATE
```

`ATE` will echo the commands you input to the module, which is useful for tracking the commands and troubleshooting.

You will receive `OK` when you input the two commands. After setting `ATE`, you can now see all the commands you input together with the replies.

Try again `AT` and you should see it on the terminal followed by `OK`.



```
23:55  AT
23:55  OK
AT
```

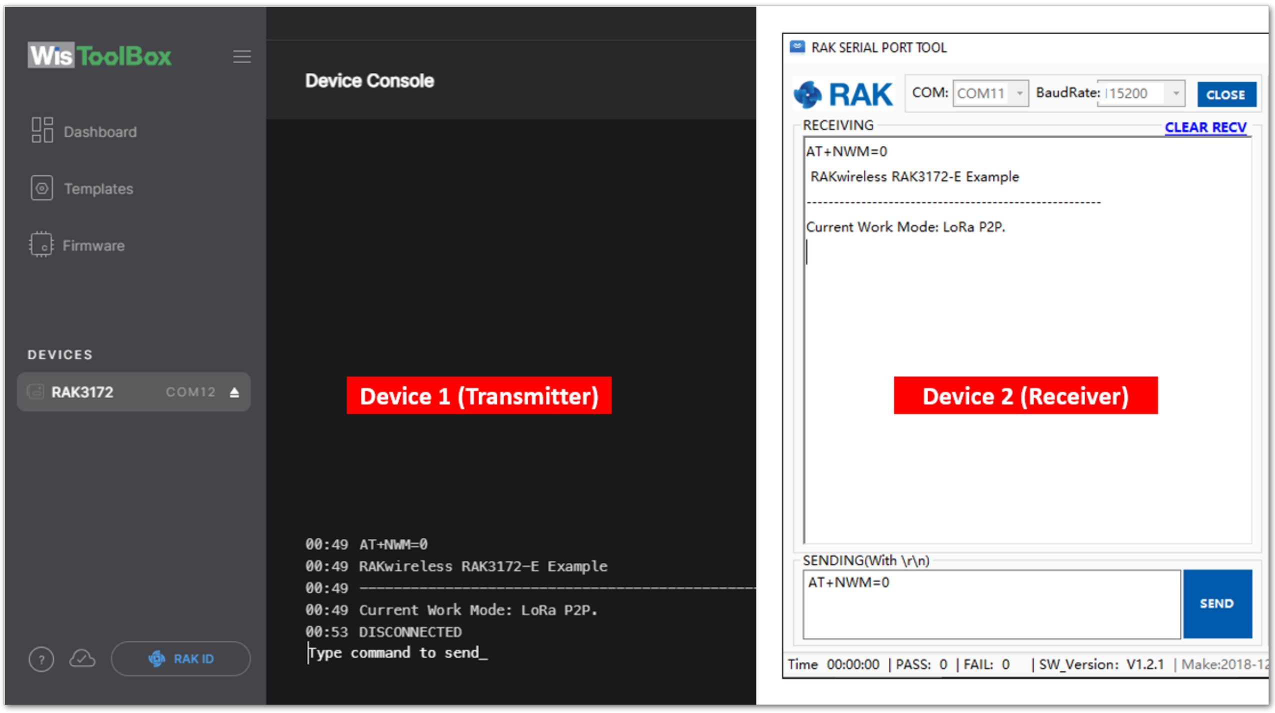
```
00:16  OK
AT
```

Figure 229: at+version command response

1. In setting up the RAK3172 to work in LoRa P2P mode, you need to change the LoRa network work mode command on both RAK3172 modules.

```
AT+NWM=0
```

`AT+NWM` parameter mode can be either 0=LoRa P2P or 1=LoRaWAN.



```
00:49 AT+NWM=0
00:49 RAKwireless RAK3172-E Example
00:49 ----------------------------------------------------------------
00:49 Current Work Mode: LoRa P2P.
00:53 DISCONNECTED
Type command to send_
```

```
AT+NWM=0
 RAKwireless RAK3172-E Example
----------------------------------------------------
Current Work Mode: LoRa P2P.
```

```
AT+NWM=0
```

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-12

Figure 230: P2P Mode

> 📝 **NOTE:**
>
> - The device will start automatically if you change modes from LoRaWAN to LoRa P2P and vice-versa.
> - You might need to input the `ATE` command again to ensure that your succeeding commands on P2P mode echo on the terminal.

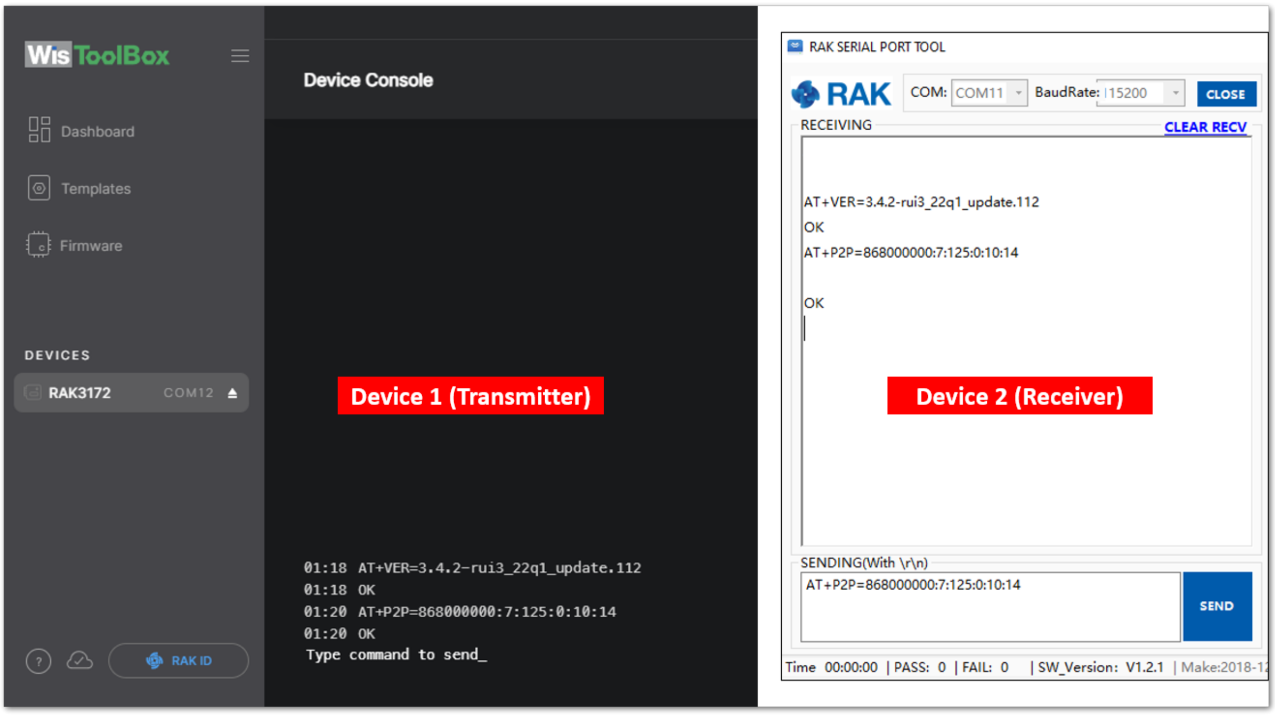
2. You need to input the P2P setup on both RAK3172 modules. The parameters should be exactly the same in the two modules.

```
AT+P2P=868000000:7:125:0:10:14
```

For this P2P setup, the LoRa parameters are the following:

- Link frequency: **868000000 Hz**
- Spreading factor: **7**
- Bandwidth: **125 kHz**
- Coding Rate: 0 = **4/5**
- Preamble Length: **10**
- Power: **14 dBm**

> 📝 **NOTE:**
>
> Refer to the P2P Mode section of the AT command documentation to learn more about the definition of the parameters used and the individual commands if you want specific parameters changed.



```
01:18  AT+VER=3.4.2-rui3_22q1_update.112
01:18  OK
01:20  AT+P2P=868000000:7:125:0:10:14
01:20  OK
Type command to send_
```

```
AT+VER=3.4.2-rui3_22q1_update.112
OK
AT+P2P=868000000:7:125:0:10:14

OK
```

```
AT+P2P=868000000:7:125:0:10:14
```

Time 00:00:00 | PASS: 0 | FAIL: 0 | SW_Version: V1.2.1 | Make:2018-1

Figure 231: Configuring P2P in both RAK3172 Module

3. To set one module as the receiver (RX), you need to set the value of the P2P receive command.

> 📝 **NOTE:**
>
> LoRa P2P default setting is Transmitter (TX) mode. This consumes lower power compared to Receiver (RX) mode where the radio is always listening for LoRa packets.

a. P2P LoRa RX configurable duration value is from 1 to 65533 ms. In this example, the device will listen and wait for LoRa P2P Packets for 30000 ms or 30 seconds. It will automatically disable RX mode and switch to TX mode after the timeout. If the device did not receive any packets within the time period, then the callback after timeout is `+EVT:RXP2P RECEIVE TIMEOUT`.

```
AT+PRECV=30000
```

b. If the `AT+PRECV` value is set to **65535**, the device will listen to P2P LoRa packets without a timeout, but it will stop listening once a P2P LoRa packet is received. After receiving the packets, it will disable RX mode and automatically switch to TX mode again.

```
AT+PRECV=65535
```

c. If the `AT+PRECV` value is set to **65534**, the device will continuously listen to P2P LoRa packets without any timeout. They will continuously stay in RX mode until `AT+PRECV` is set to **0**.

```
AT+PRECV=65534
```

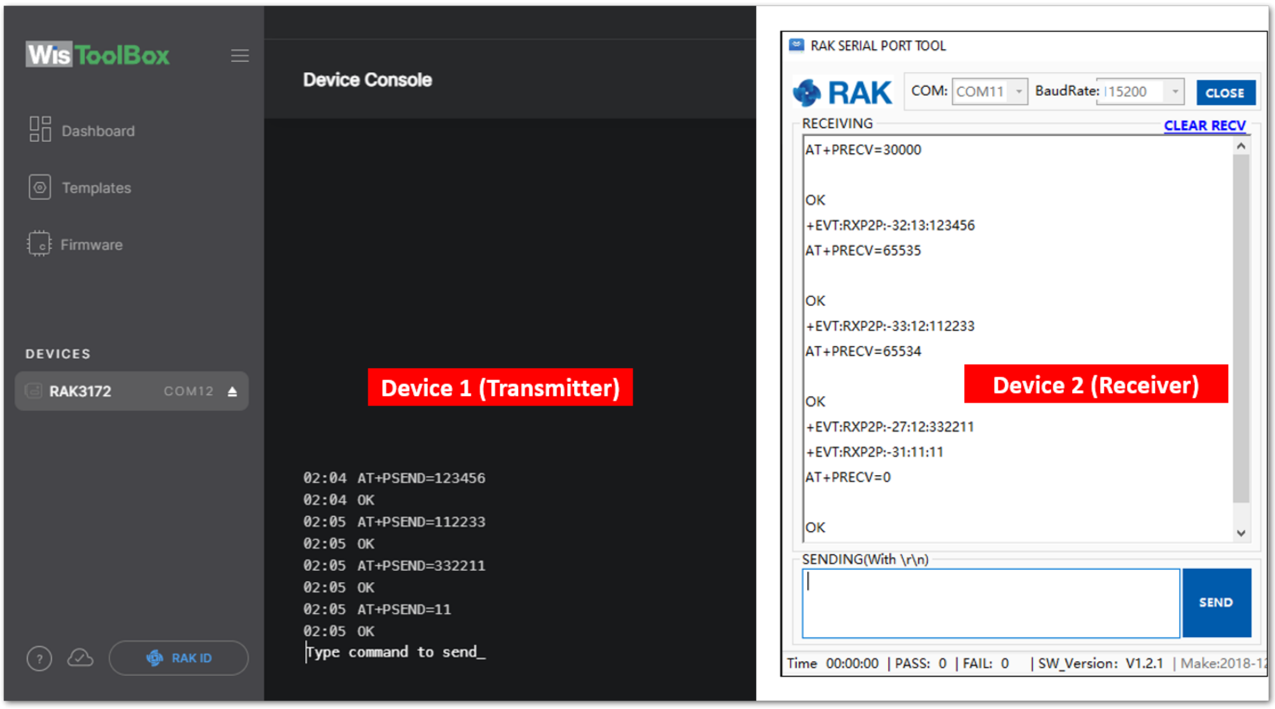
d. If the `AT+PRECV` value is set to **0**, the device will stop listening to P2P LoRa packets. It disables LoRa P2P RX mode and switches to TX mode.

```
AT+PRECV=0
```

4. With one module configured as Transmitter (TX) and the other device will be the Receiver (RX), you can now try to send or transmit P2P payload data.

```
AT+PSEND= <payload>
```

> 📝 **NOTE:**
>
> - `AT_PARAM_ERROR` is returned when setting the wrong or malformed value.
> - `AT_BUSY_ERROR` is returned if the device is still in RX mode and you try to send or reconfigure the RX period. If the `AT+PRECV` command is set to **65534**, you need to execute first `AT+PRECV=0` to be able to configure again the TX and RX state and avoid `AT_BUSY_ERROR`.
> - `<payload>` : 2~500 digit length, must be an even number of digits and character 0-9, a-f, A-F only, representing 1~256 hexadecimal numbers. For example, if the payload is like `0x03, 0xAA, 0x32`, then the AT command should be `AT+PSEND = 03AA32`.



```
02:04  AT+PSEND=123456
02:04  OK
02:05  AT+PSEND=112233
02:05  OK
02:05  AT+PSEND=332211
02:05  OK
02:05  AT+PSEND=11
02:05  OK
Type command to send_
```

```
AT+PRECV=30000

OK
+EVT:RXP2P:-32:13:123456
AT+PRECV=65535

OK
+EVT:RXP2P:-33:12:112233
AT+PRECV=65534

OK
+EVT:RXP2P:-27:12:332211
+EVT:RXP2P:-31:11:11
AT+PRECV=0

OK
```

Figure 232: Configuring P2P in both RAK3172 Module

Miscellaneous

Upgrading the Firmware

If you want to upgrade to the latest version of the firmware of the module, you can follow this section. The latest firmware can be found in the software section of RAK3172 Datasheet.

> 📝 **NOTE:**
>
> **What if the RAK3172 module stops responding to AT commands and firmware updates?**
>
> You can recover your device by using the `.hex` file in the datasheet and uploading it using STM32CubeProgrammer. The guide on updating STM32 firmware using STM32CubeProgrammer can be found in the Learn section.
>
> **WARNING:** Uploading the **.hex** file via STM32CubeProgrammer will erase all configured data on the device.

Firmware Upgrade Through UART2

Minimum Hardware and Software Requirements

Refer to the table for the minimum hardware and software required to perform the firmware upgrade via UART2:

Hardware/Software	Requirement
Computer	A Windows/Ubuntu/Mac computer
Firmware File	Bin firmware file downloaded from the website
Others	A USB to TTL module

Firmware Upgrade Procedure

Execute the following procedure to upgrade the firmware in Device Firmware Upgrade (DFU) mode through the UART2 interface.

> 📝 **NOTE:**
>
> RAK3172 should automatically go to BOOT mode when the firmware is uploaded via RAK DFU Tool or WisToolBox.
>
> If BOOT mode is not initiated, you can manually send `AT+BOOT` command to start bootloader mode.

1. Download the latest application firmware of the RAK3172.
   - RAK3172 Firmware
2. Download the RAK Device Firmware Upgrade (DFU) tool.
   - RAK Device Firmware Upgrade (DFU) Tool
3. Connect the RAK3172 module with a computer through a USB to TTL. Refer to Figure 35.
4. Open the Device Firmware Upgrade tool. Select the serial port and baud rate (115200) of the module and click the **Select Port** button.

> 📝 **NOTE:**
>
> If your firmware upload always fails, check your current baud rate setting using the `AT+BAUD=?` command and use that baud rate value in the RAK DFU Tool. You can also check if you selected the right COM port.

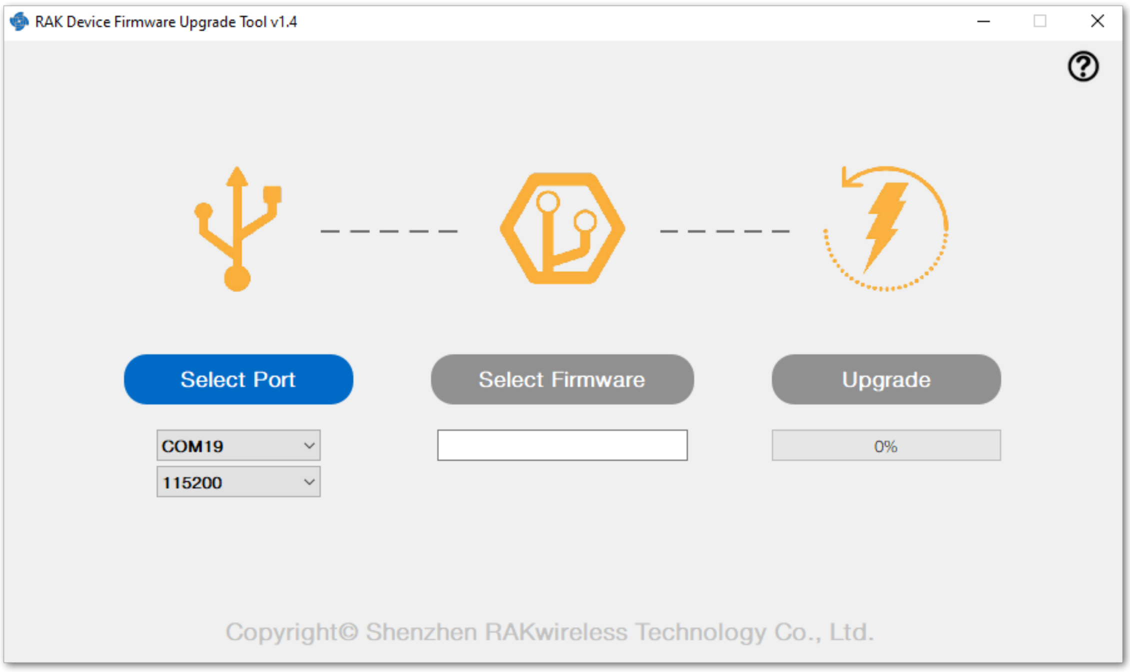

RAK Device Firmware Upgrade Tool v1.4
Select Port
Select Firmware
Upgrade
COM19
115200
0%
Copyright© Shenzhen RAKwireless Technology Co., Ltd.

Figure 233: Device Firmware Upgrade Tool

5. Select the application firmware file of the module with the suffix **.bin**.

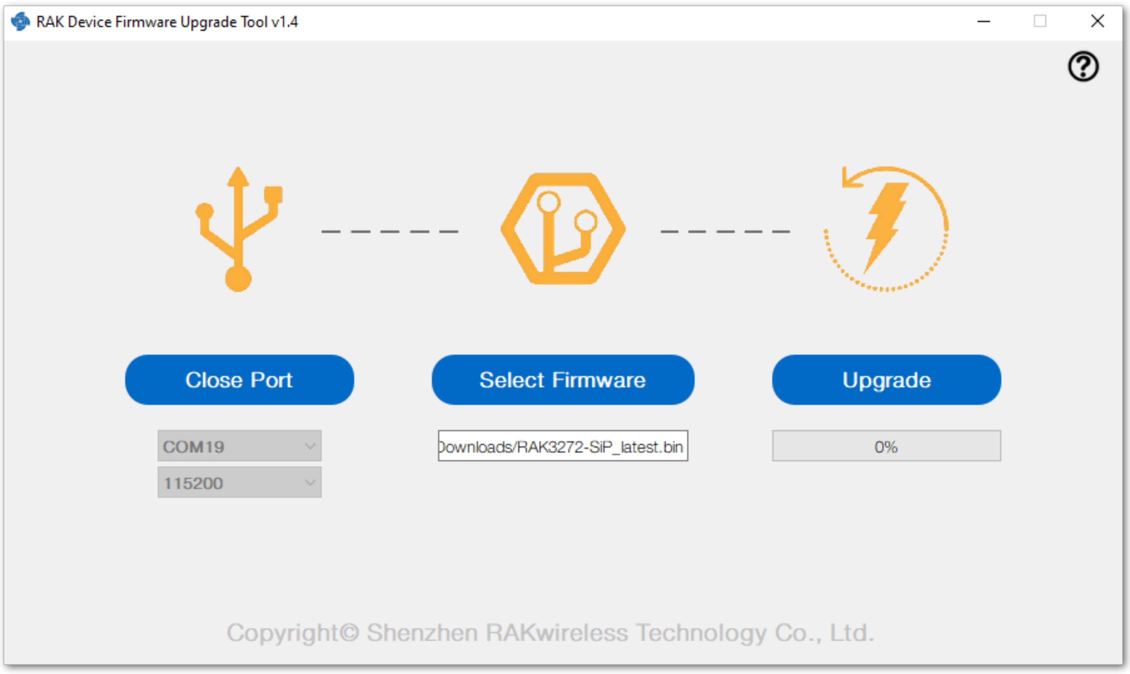
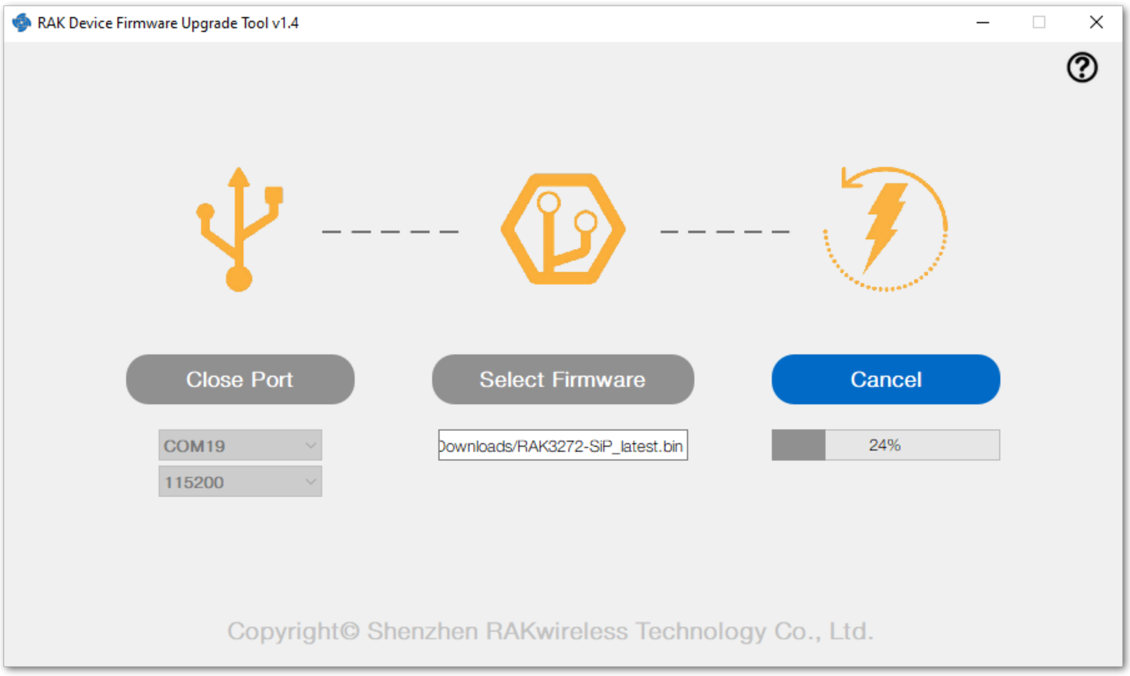

RAK Device Firmware Upgrade Tool v1.4
Close Port
Select Firmware
Upgrade
COM19
115200
Downloads/RAK3272-SiP_latest.bin
0%
Copyright© Shenzhen RAKwireless Technology Co., Ltd.

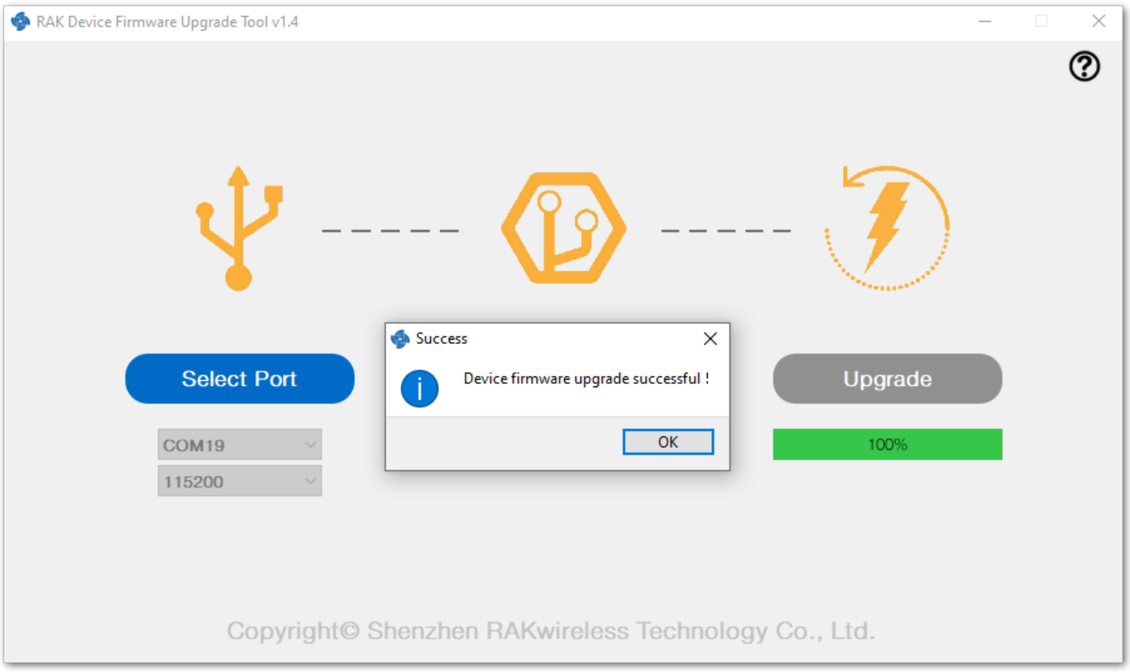
Figure 234: Select firmware

6. Click the **Upgrade** button to upgrade the device. After the upgrade is complete, the RAK3172 will be ready to work with the new firmware.



RAK Device Firmware Upgrade Tool v1.4

Figure 235: Firmware upgrading



RAK Device Firmware Upgrade Tool v1.4

Success

Device firmware upgrade successful !

Figure 236: Upgrade successful

Arduino Installation

Refer to Software section.

Last Updated: 4/24/2023, 7:45:53 AM