

RAK3272S Quick Start Guide

This guide covers the following topics:

- [RAK3272S Breakout Board as a Stand-Alone Device Using RUI3](#)
- [RAK3272S Breakout Board 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 RAK3272S Breakout Board, make sure to prepare the necessary items listed below:

Hardware

- [RAK3272S Breakout Board](#) 
- [RAKDAP1 Flash and Debug Tool](#)  (or any USB-Serial Adapter)
- Windows/Mac OS/Linux Computer with USB port

Software

- Download and install the [Arduino IDE](#)  .

WARNING

For Windows 10 users: 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 [RAK3272S as a supported board in Arduino IDE](#) by updating Board Manager URLs in **Preferences** settings of Arduino IDE with this JSON URL
`https://raw.githubusercontent.com/RAKWireless/RAKwireless-Arduino-BSP-Index/main/package_rakwireless.com_rui_index.json` . 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
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

Product Configuration

RAK3272S Breakout Board as a Stand-Alone Device Using RUI3

Hardware Setup

The RAK3272S 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 precaution 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 RAK3272S breakout board 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 and SWDIO). This method will require you to use external tools like ST-LINK or RAKDAP1.

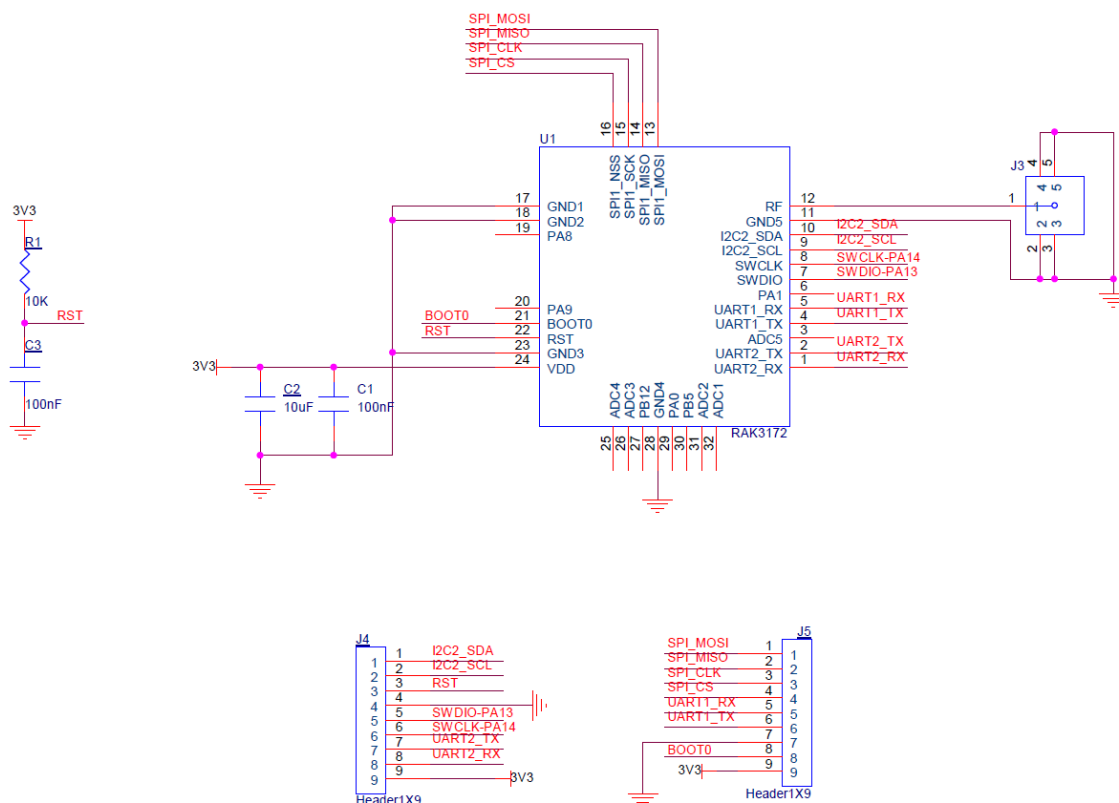


Figure 1: RAK3272S Schematic

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 RP-SMA connector.



Figure 2: LoRa Antenna

RAK3272S has RP-SMA connector compatible to the included LoRa antenna, as shown in **Figure 3**.

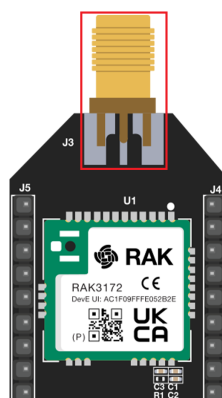


Figure 3: RP-SMA Connector of RAK3272S for LoRa Antenna

NOTE

Detailed information about the RAK3272S 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 RAK3272S is based on RUI3, which allows you to develop your custom firmware to connect sensors and other peripherals to it. To develop your custom firmware using Arduino IDE, first, you need to add **RAKwireless RUI STM32 Boards** in the Arduino board manager, which will be discussed in this guide. You can then use RUI3 APIs for your intended application and upload also the custom firmware via UART. The AT commands of RAK3272S is still available even if you compile custom firmware via RUI3. You can send AT commands via UART2 connection.

RAK3272S 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 installed successfully, you can now configure the IDE to add the RAK3272S in its board selection by following these steps.

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

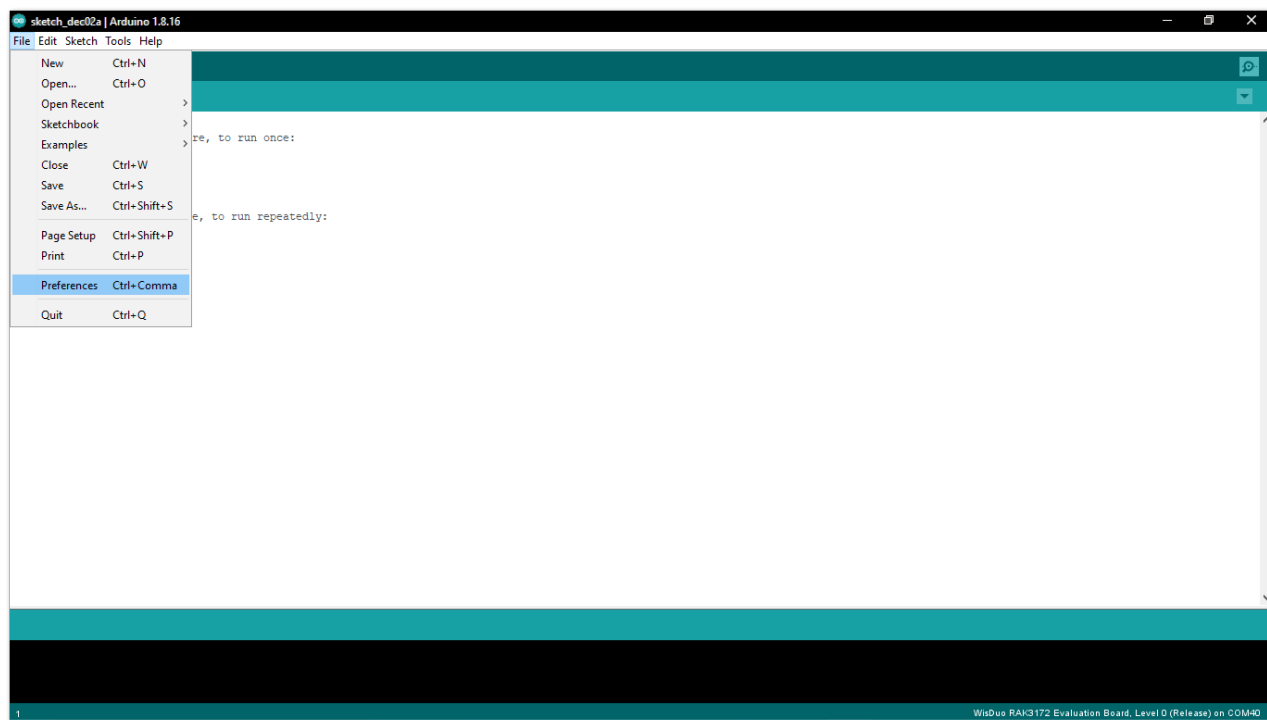


Figure 4: Arduino preferences

2. To add the RAK3272S to your Arduino Boards list, edit the **Additional Board Manager URLs** and click the icon, as shown in **Figure 5**.

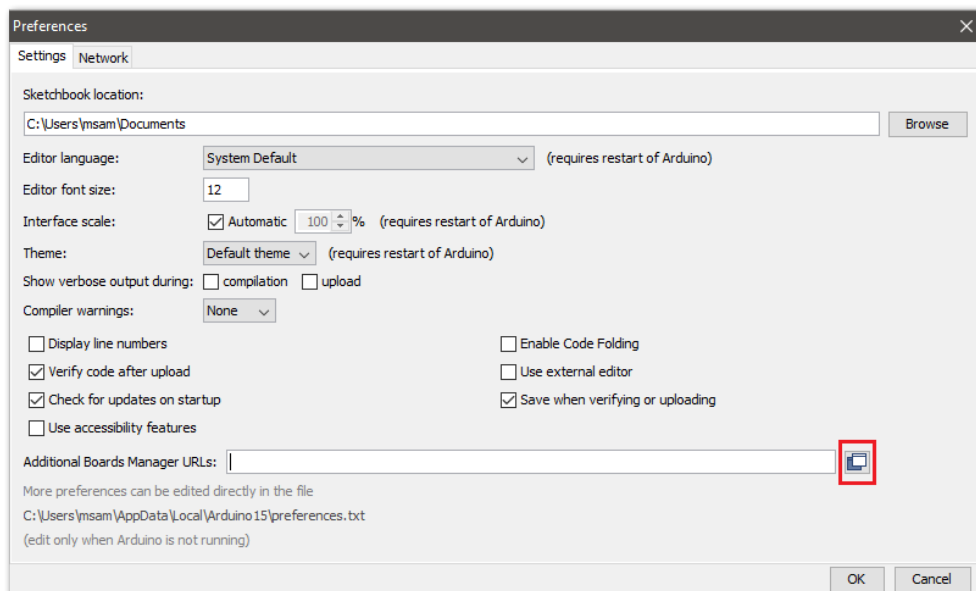


Figure 5: Modifying Additional Board Manager URLs

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

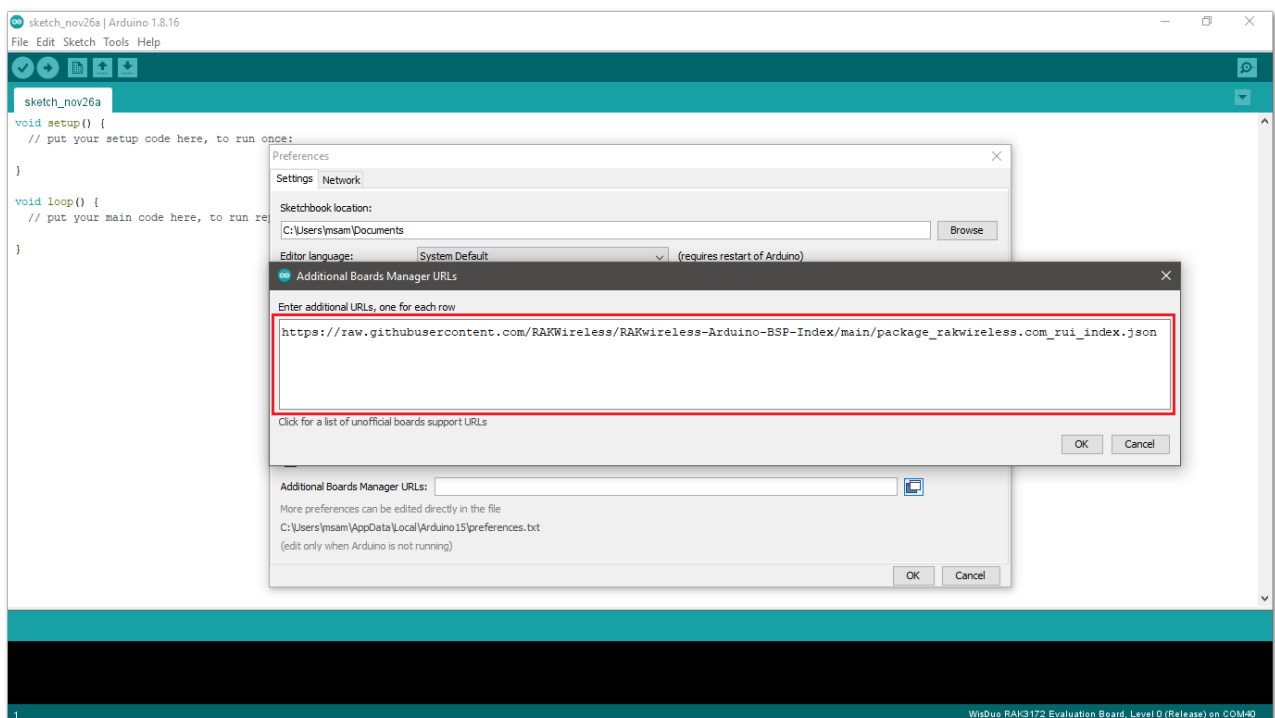


Figure 6: Add additional board manager URLs

- Restart the Arduino IDE.
- Open the Boards Manager from Tools Menu.

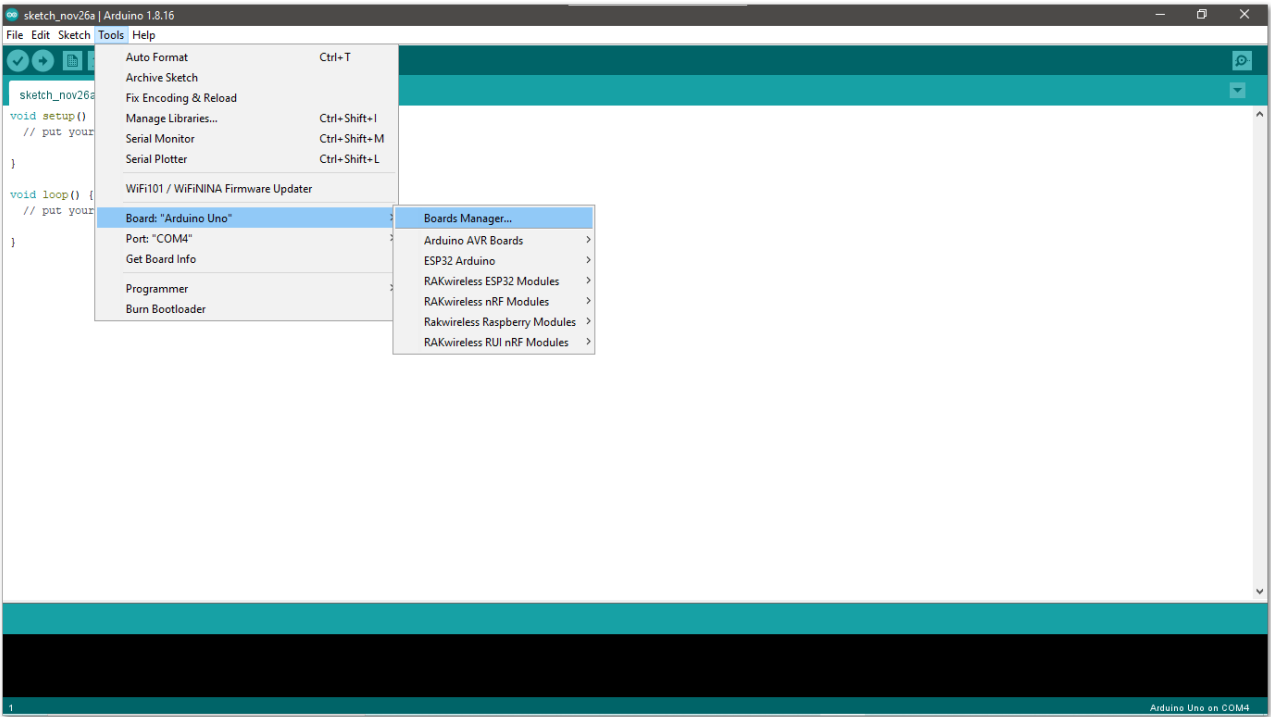


Figure 7: Opening Arduino boards manager

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. Select and install the latest version of the **RAKwireless RUI STM32 Boards**.

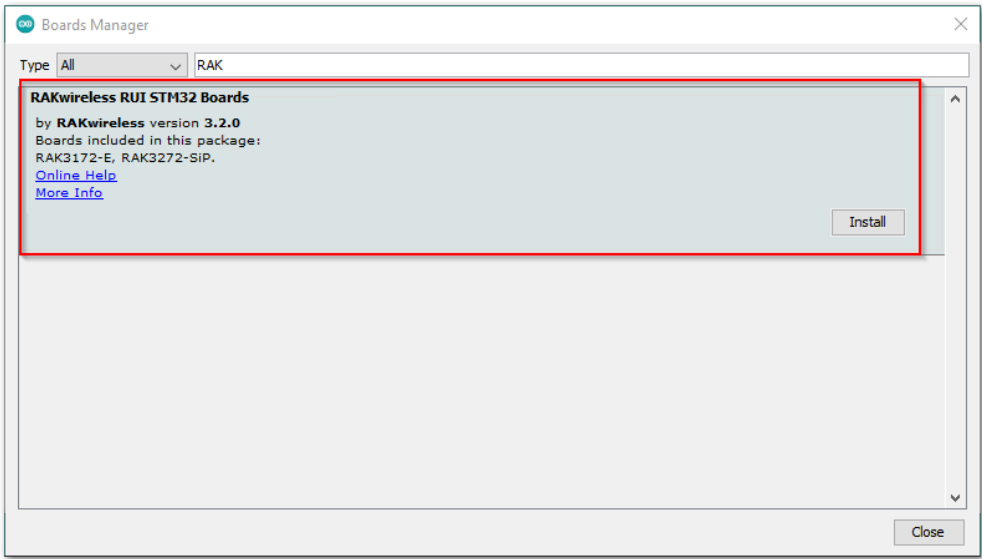


Figure 8: Installing RAKwireless RUI STM32 boards

7. Once the BSP is installed, select **Tools > Boards Manager > RAKWireless RUI STM32 Modules > WisDuo RAK3172 Evaluation Board**. The RAK3272S Breakout Board uses RAK3172 WisDuo module.

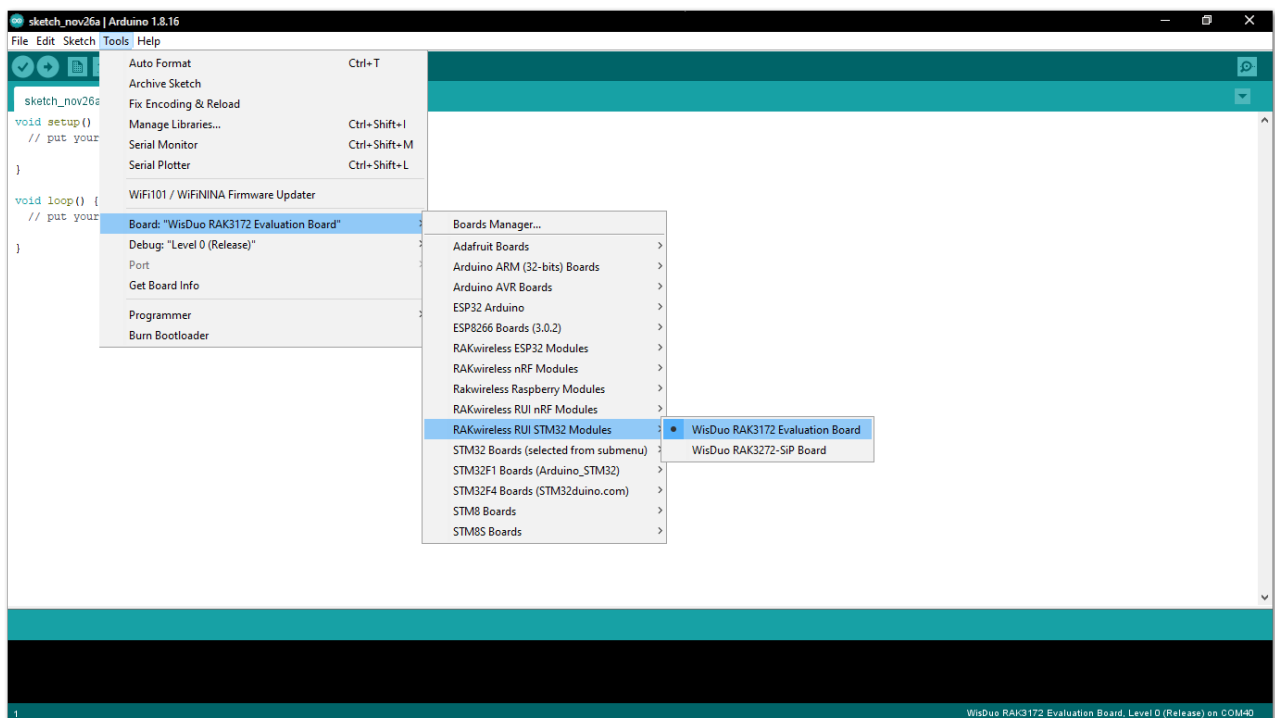


Figure 9: Selecting WisDuo RAK3172 Evaluation Board

Compile an Example with Arduino Serial

1. After completing the steps on adding your RAK3272S to the Arduino IDE, you can now try to run a simple program to test your setup. You need to add a USB connection to the schematic of the RAK3272S breakout board, as shown in **Figure 10**.

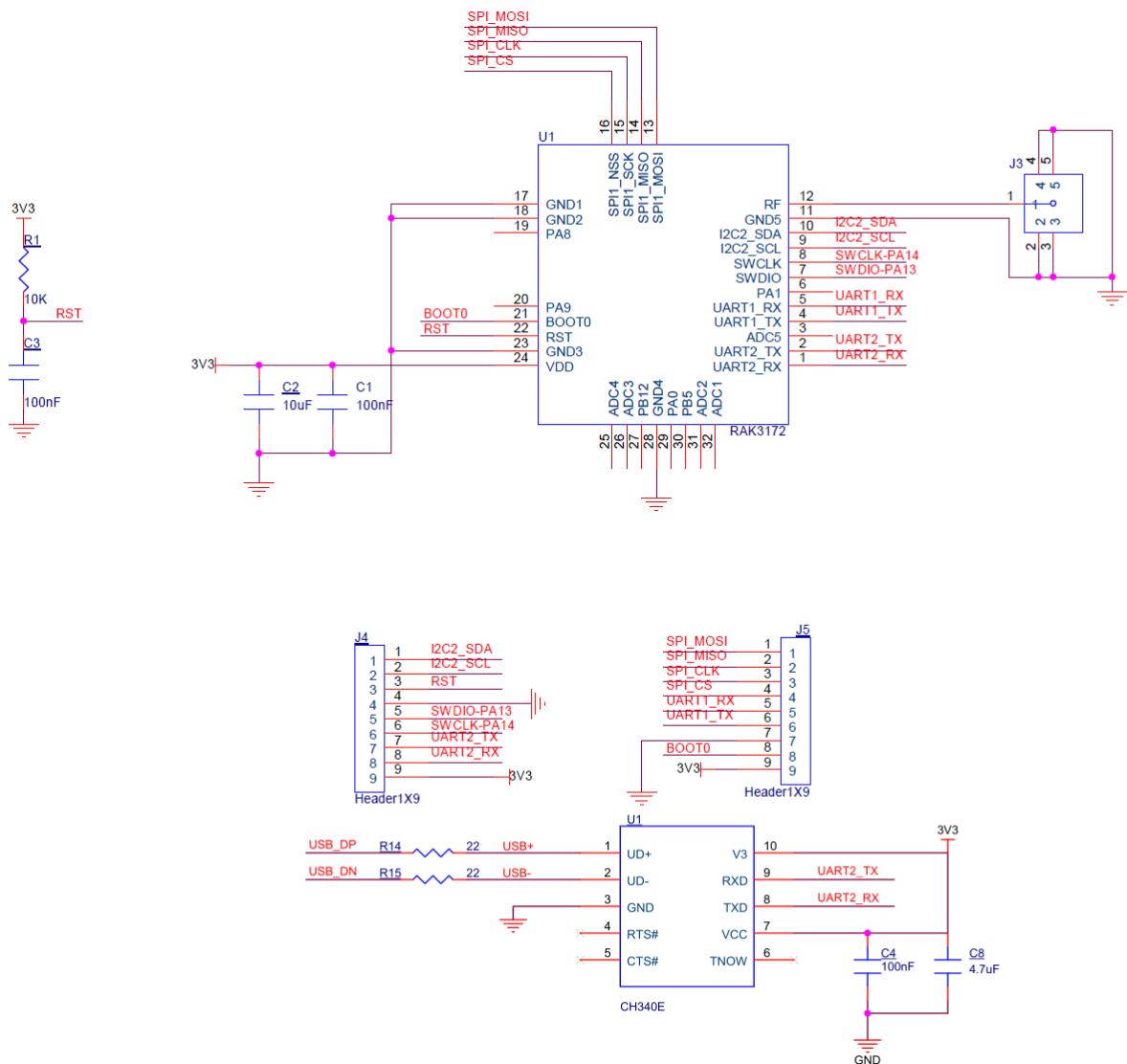


Figure 10: RAK3272S with USB to Serial Schematic

2. Connect the RAK3272S via UART and check RAK3272S COM Port using Windows **Device Manager**. Double-click the reset button if the module is not detected.

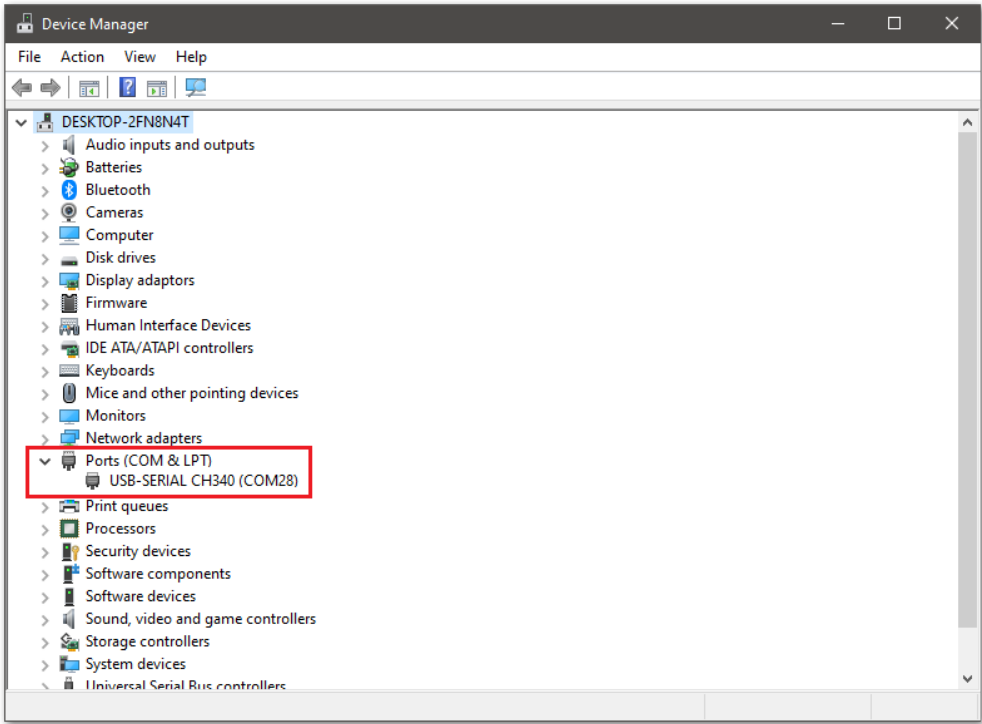


Figure 11: Device manager ports (COM & LPT)

3. Choose RAK3272S on board selection select via **Tools > Boards Manager > RAKWireless RUI STM32 Modules > WisDuo RAK3172 Evaluation Board**.

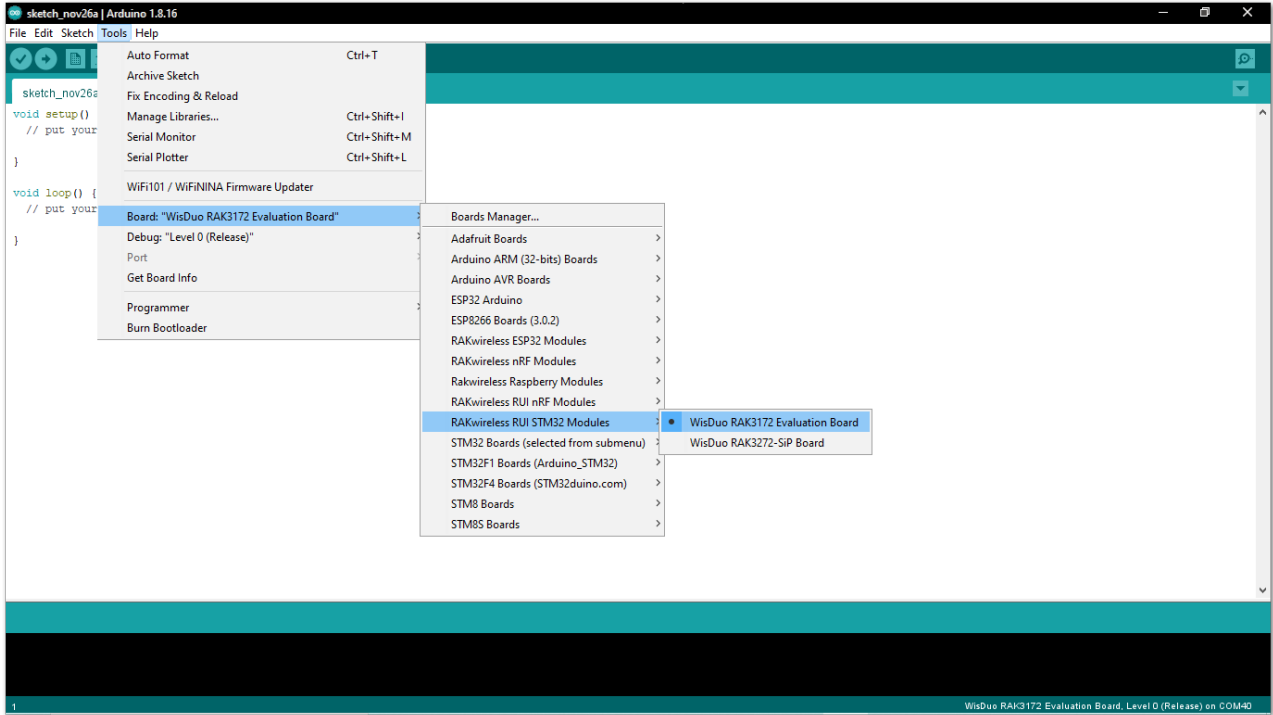


Figure 12: Selecting RAK3272S Breakout Board

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

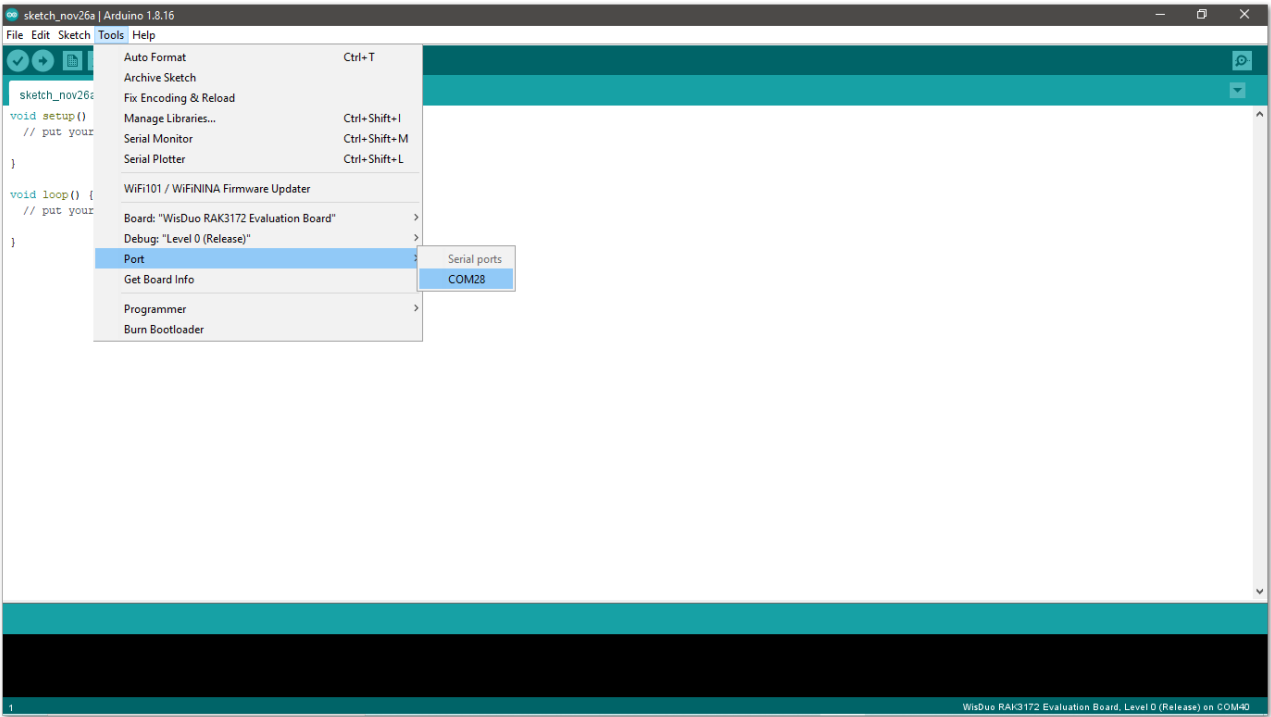


Figure 13: Select COM port

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

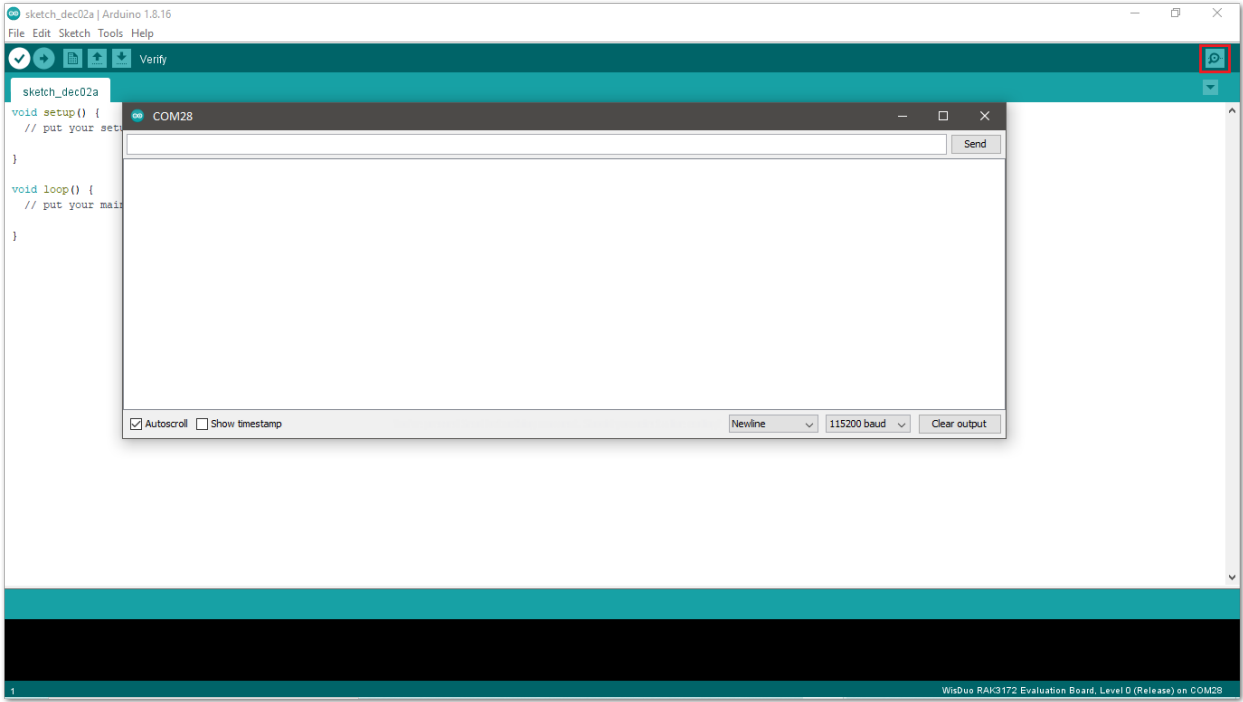


Figure 14: Open Arduino serial monitor

6. If the connection is successful, you can send AT Commands to RAK3272S. For example: To check the RUI version, type `AT+VER=?` on the text area, then click on the **Send** button, as shown in **Figure 15**.

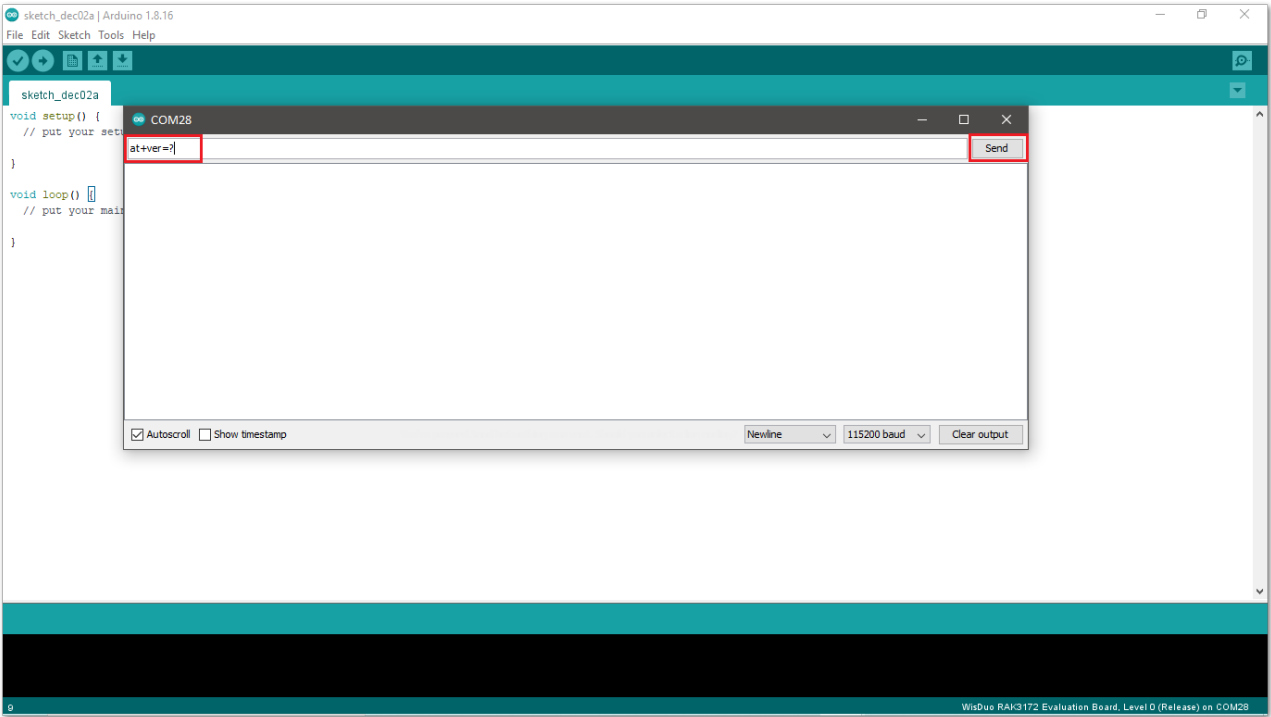


Figure 15: Send AT command

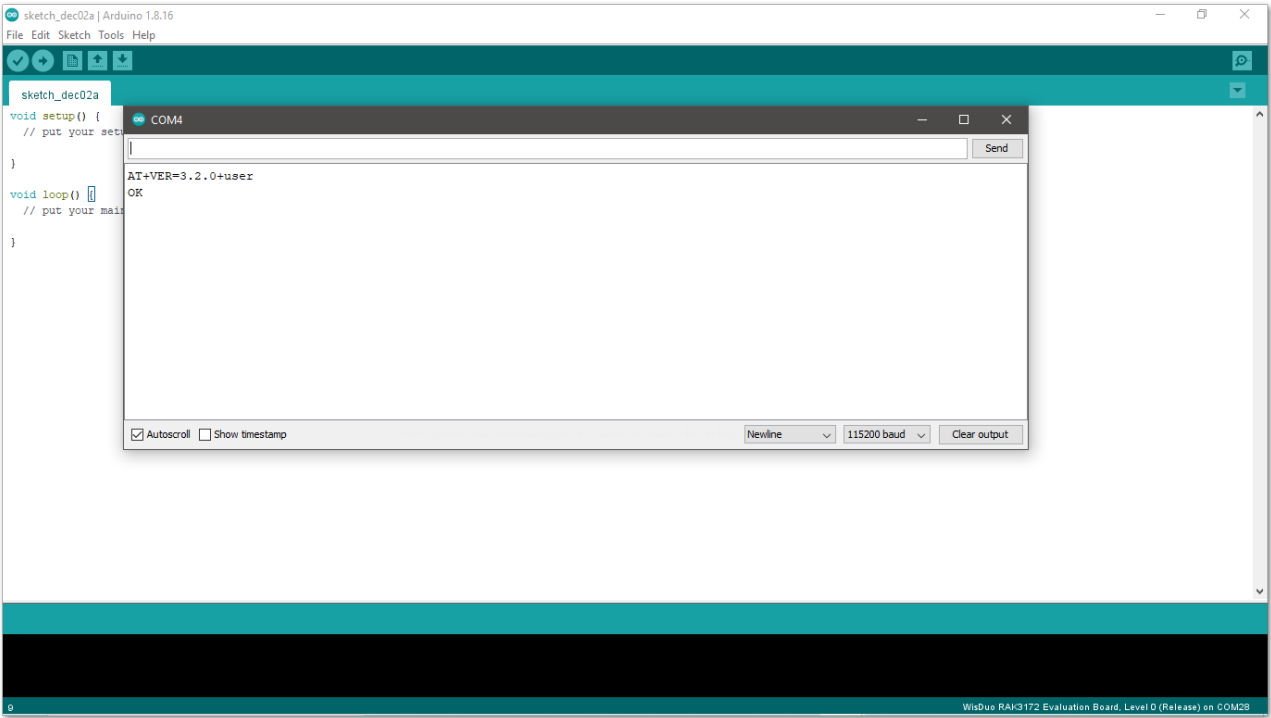


Figure 16: Arduino serial monitor COM28

7. Open the **Arduino_Serial** example code.

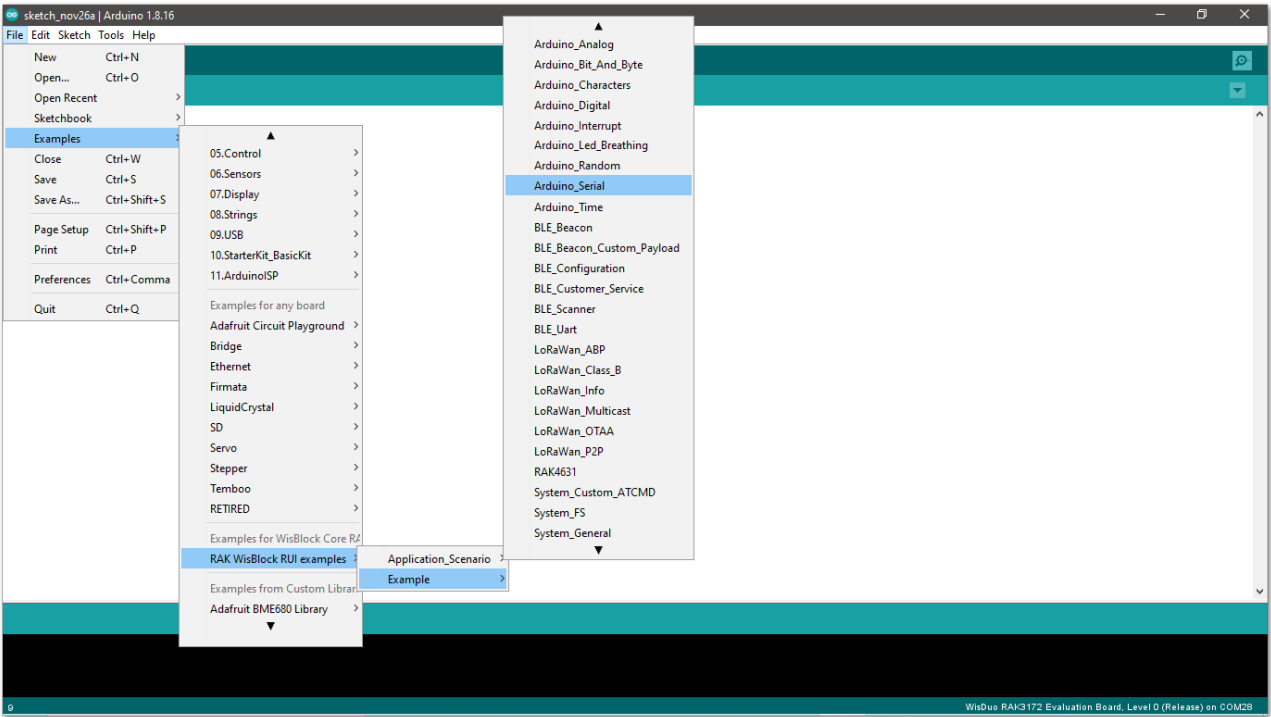


Figure 17: Arduino Serial example

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

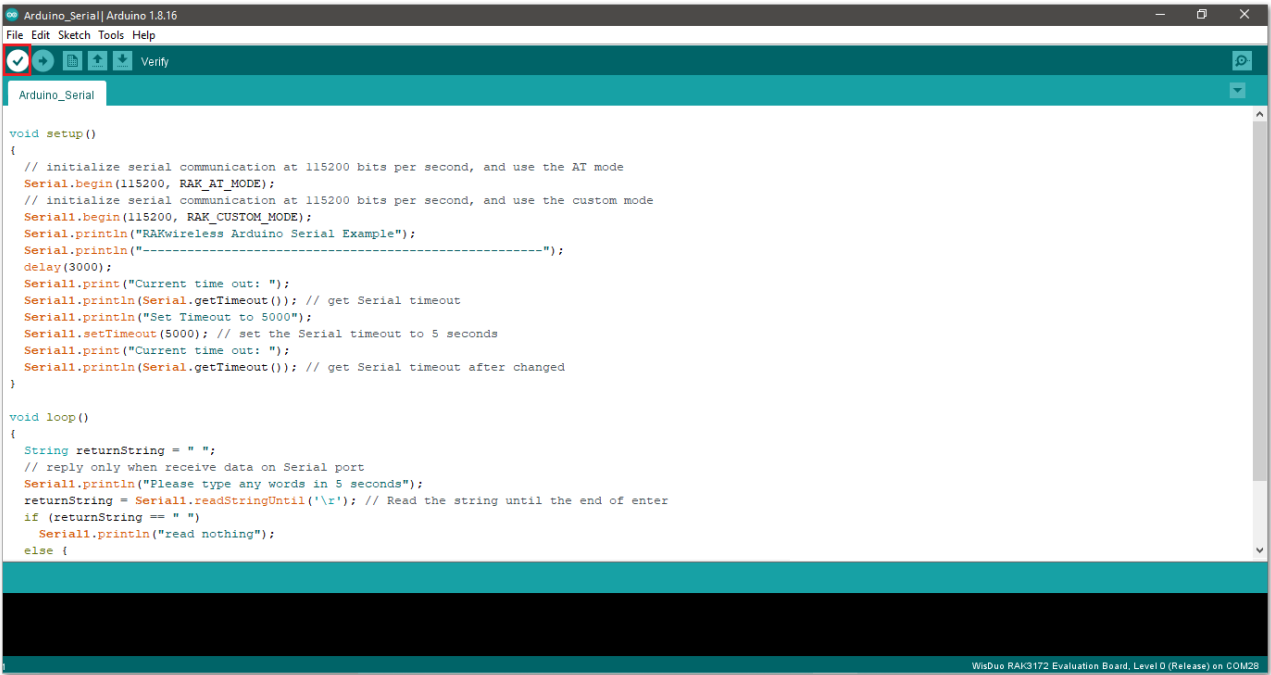


Figure 18: Verify the example code

9. Click the **Upload** icon to send the compiled firmware to your RAK3272S.

 **NOTE:**

RAK3272S 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.

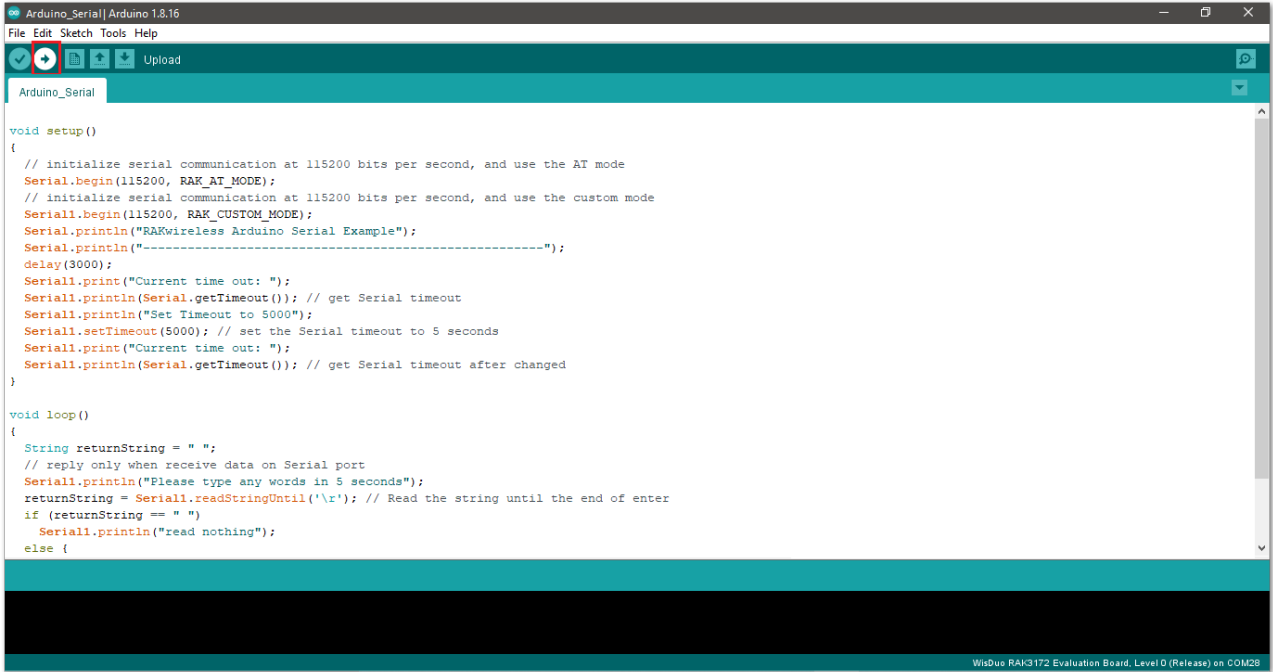


Figure 19: Upload the example code

10. If the upload is successful, you will see the **Device programmed** message.

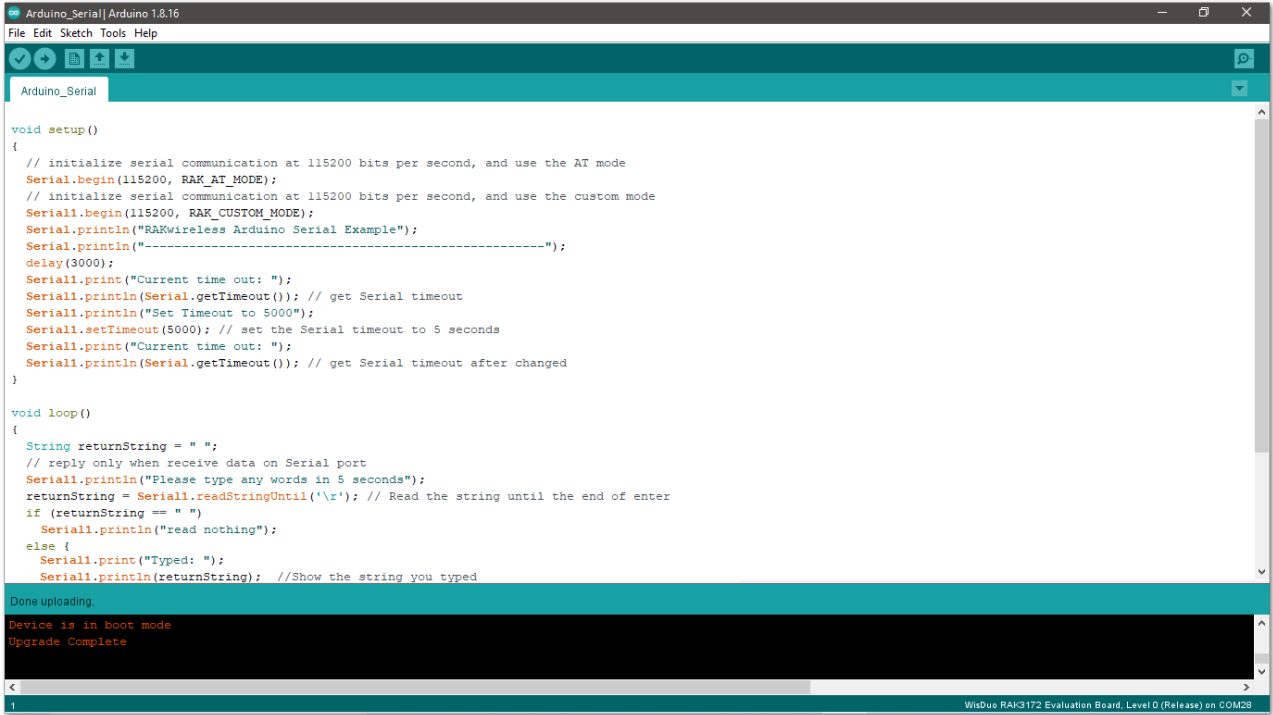


Figure 20: Device programmed successfully

11. After the Device Programmed is completed, you will see the working Arduino_Serial example.

RAK3272S Breakout Board as a LoRa/LoRaWAN Modem via AT Command

AT Command via UART2

RAK3272S breakout board can be configured using AT commands via the UART2 interface. You need a USB to UART TTL adapter to connect the RAK3272S 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 save.

⚠ WARNING

Firmware update and AT command functionality is done via UART2 pins. If you will connect the module to an external host MCU that will send AT commands via UART2, take extra precaution 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 RAK3272S 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 and SWDIO). This method will require you to use external tools like ST-LINK or RAKDAP1.

Connect to the RAK3272S Breakout Board

1. Connect the RAK3272S to the serial port of a general-purpose computer (USB port) using a USB to UART TTL adapter like [RAKDAP1](#) , as shown in **Figure 21**.

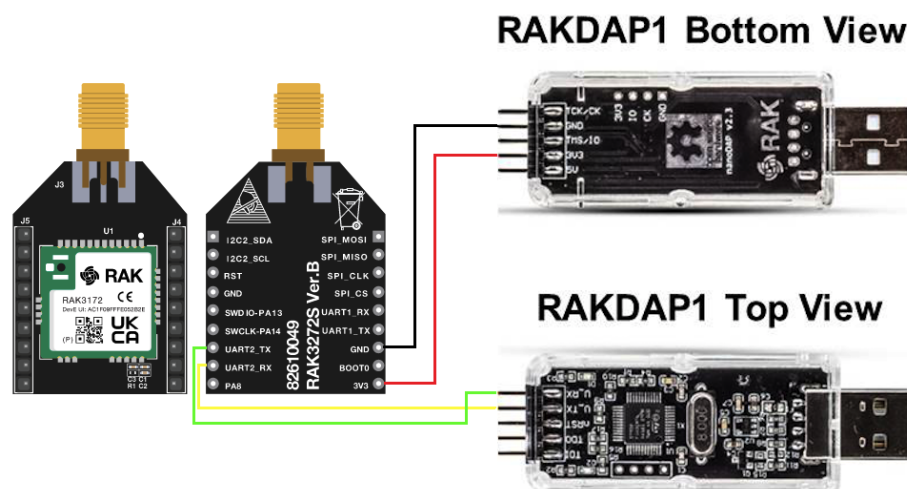


Figure 21: RAK3272S Breakout Board to USB-Serial Connection

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**

RAK3272S I/O Pins and Peripherals

This section discusses how to use and access RAK3272S peripherals pins using RUI3 APIs. It shows basic code on using Digital I/O Pins, UART, and I2C.

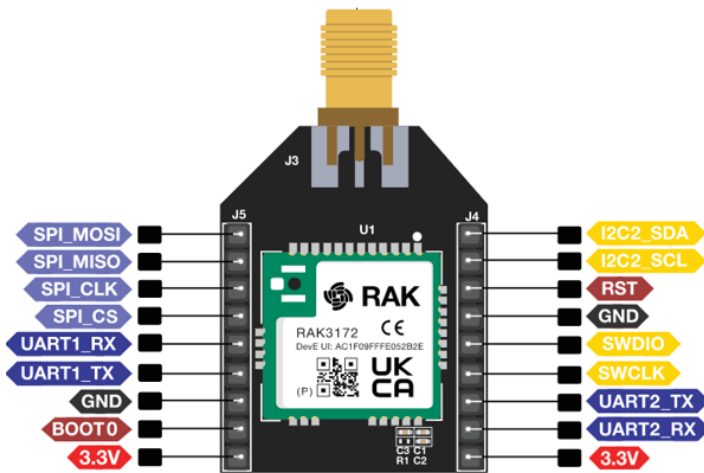


Figure 22: Available Peripheral pins in RAK3272S Breakout Board

How to Use Digital I/O

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

Pin Name	J Connector pin
PA4	J5 pin 4
PA5	J5 pin 3
PA6	J5 pin 2
PA7	J5 pin 1
PA8	J4 pin 1
PA9	J4 pin 2
PB6	J5 pin 5
PB7	J5 pin 6

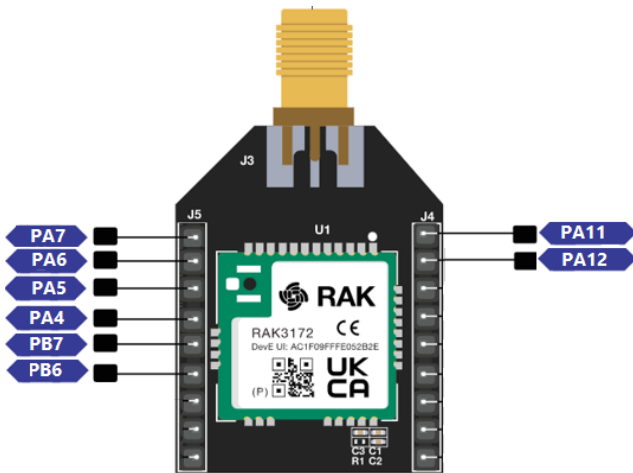


Figure 23: Available Digital I/O pins in RAK3172S

- 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

```
void setup()
{
  pinMode(PA7, OUTPUT); //Change the P0_04 to any digital pin you want. Also, you can set this to
}

void loop()
{
  digitalWrite(PA7,HIGH); //Change the PA0 to any digital pin you want. Also, you can set this to
  delay(1000); // delay for 1 second
  digitalWrite(PA7,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 Serial Interfaces

UART

There are two UART peripherals available on the RAK3272S. There are also different [Serial Operating Modes](#) possible in RUI3, namely [Binary Mode](#) , [AT Mode](#) , and [Custom Mode](#) .

Serial Port	UART pin number	Serial Instance Assignment	Default Mode
UART1_TX	J5 pin 6	Serial1	Custom Mode
UART1_RX	J5 pin 5	Serial1	Custom Mode
UART2_TX	J4 pin 7	Serial	AT Command
UART2_RX	J4 pin 8	Serial	AT Command

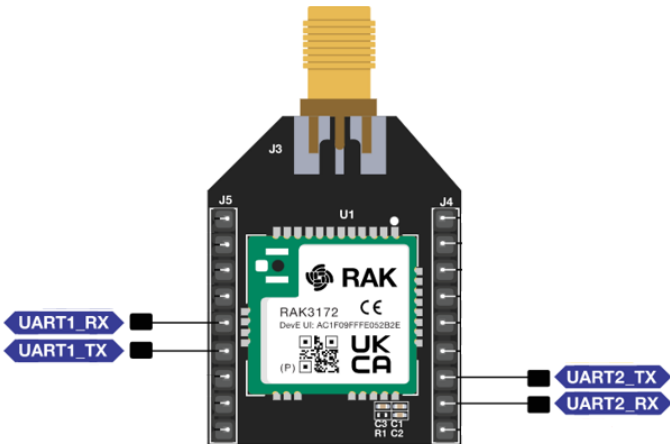


Figure 24: Available UART pins in RAK3272S

Example Code

```
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 RAK3272S.

I2C Pin Number	I2C Pin Name
J4 pin 1	I2C2_SDA
J4 pin 2	I2C2_SCL

- Use Arduino [Wire](#) library to communicate with I2C devices.

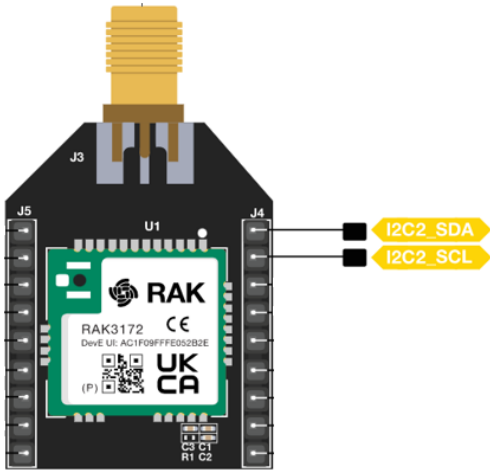


Figure 25: Available I2C pins in RAK3272S

Example Code

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



```
#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 J5 pins 1 to 4 are reserved for RUI3 SPI interface.

SPI Pin Number	SPI Pin Name
J5 pin 1	SPI_MOSI
J5 pin 2	SPI_MISO
J5 pin 3	SPI_CLK
J5 pin 4	SPI_CS

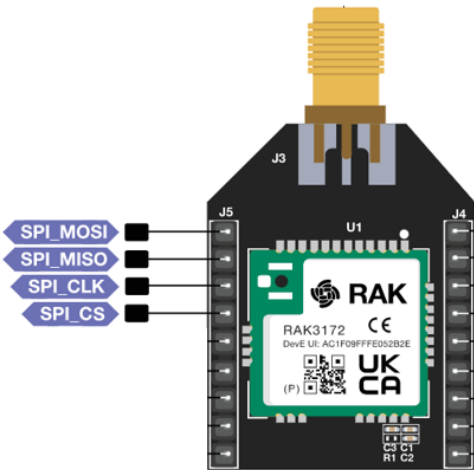


Figure 26: Available SPI pins in RAK3272S

 **NOTE:**

The J4 pins 5 and 6 are reserved for SWD debug interface. Check the [Connect to the RAK3272S Breakout Board](#) section.

RAK3272S Configuration for LoRaWAN or LoRa P2P

To enable the RAK3272S breakout board 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 RAK3272S in two ways:

- [LoRaWAN End-Device](#) - RAK3272S Breakout Board as LoRaWAN IoT device.
- [LoRa P2P](#) - Point-to-point communication between two RAK3272S breakout board modules.

Configuring RAK3272S Breakout Board as LoRaWAN End-Device

To enable the RAK3272S breakout board 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 associate AT commands for the RAK3272S breakout board.

This guide covers the following topics:

- [TheThingsNetwork Guide](#) - How to login, register new accounts and create new applications on TTN.
- [RAK3272S Breakout Board TTN OTAA Guide](#) - How to add OTAA device on TTN and what AT commands to use on RAK3272S OTAA activation.
- [RAK3272S Breakout Board TTN ABP Guide](#) - How to add ABP device on TTN and what AT commands to use on RAK3272S ABP activation.
- [Chirpstack Guide](#) - How to create new applications on Chirpstack.
- [RAK3272S Breakout Board Chirpstack OTAA Guide](#) - How to add OTAA device to Chirpstack and what AT commands to use on RAK3272S OTAA activation.
- [RAK3272S Breakout Board Chirpstack ABP Guide](#) - How to add ABP device on Chirpstack and what AT commands to use on RAK3272S ABP activation.

Connecting to The Things Network (TTN)

In this section, a quick tutorial guide will show how to connect the RAK3272S Breakout Board to the TTN platform.

 NOTE:

In this guide, you need to have a working gateway that is connected to TTN or coverage of the TTN community network.

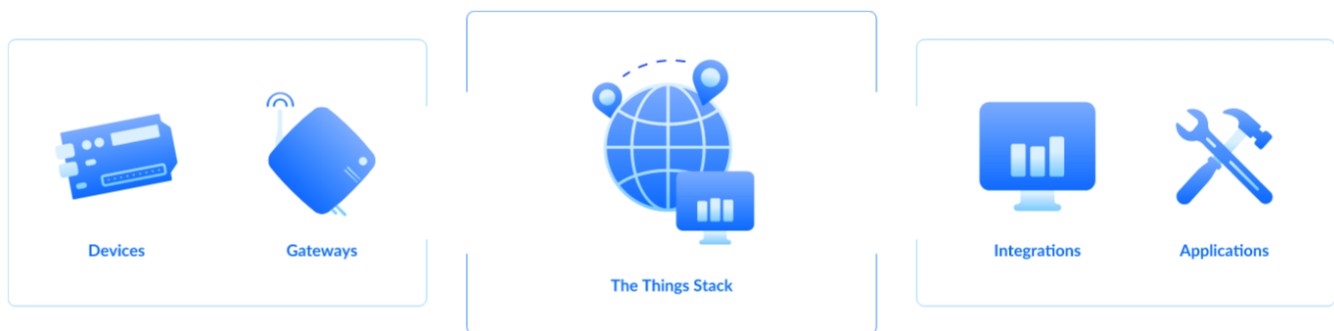


Figure 27: RAK3272S in the context of the TTN

As shown in **Figure 22**, 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 RAK3272S Breakout Board 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 RAK3272S Breakout Board 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 28**. Then select a cluster as shown in **Figure 30**.

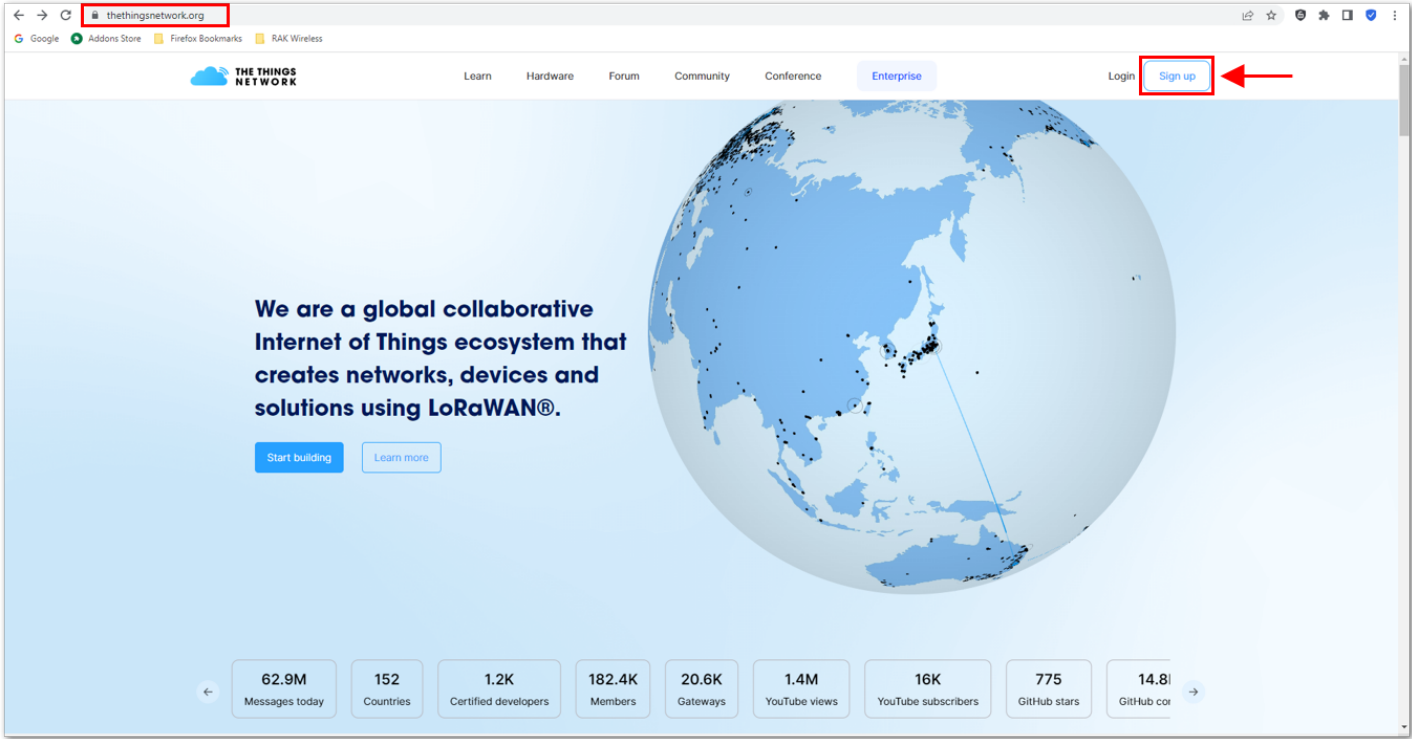


Figure 28: Signing up an account in TTN

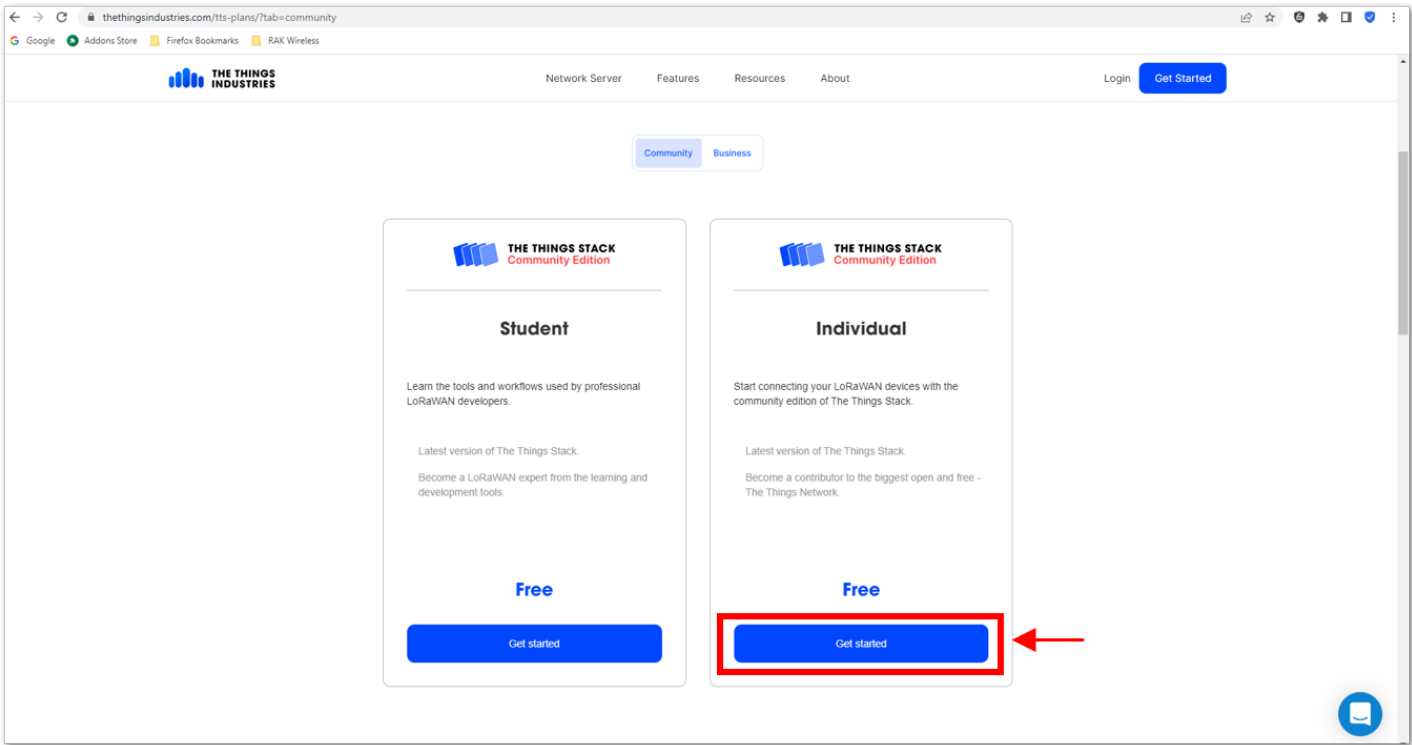


Figure 29: Signing up an account in TTN

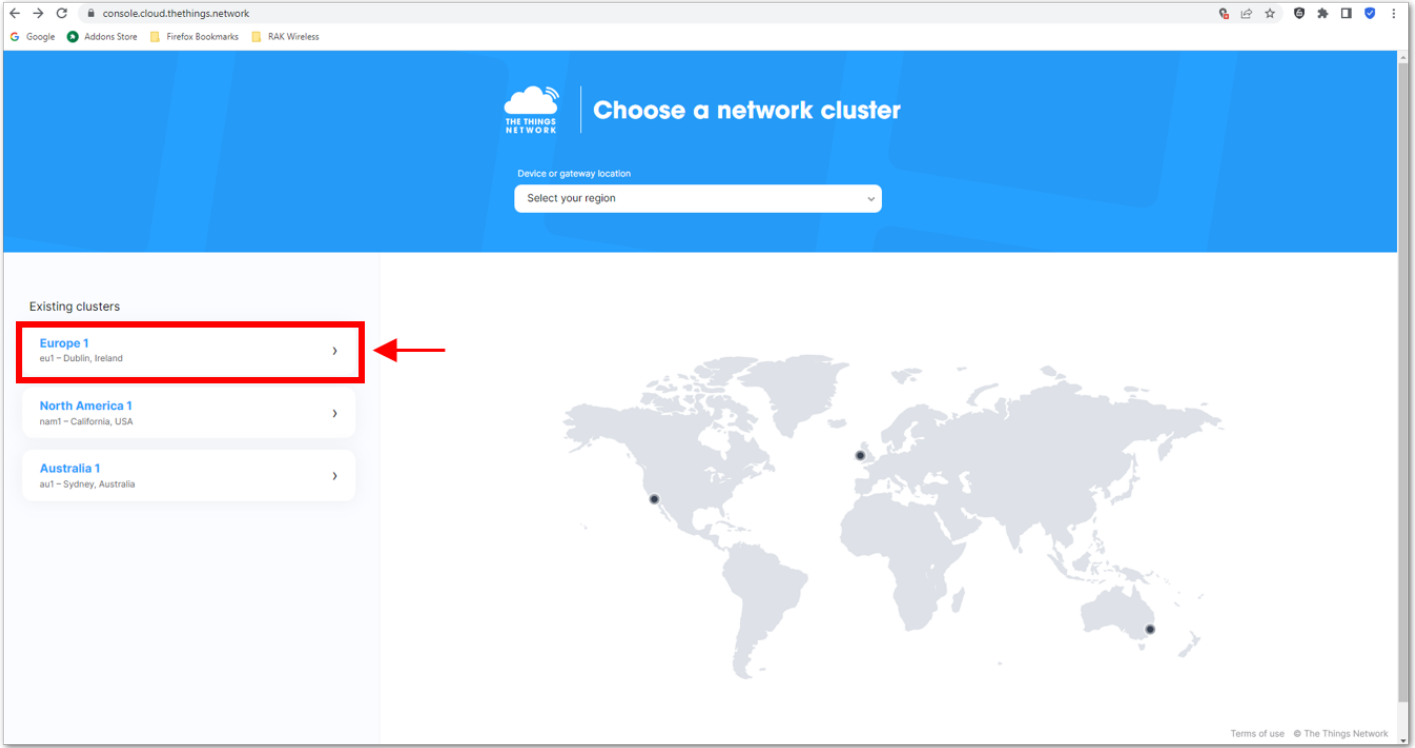


Figure 30: Selecting Cluster in TTN

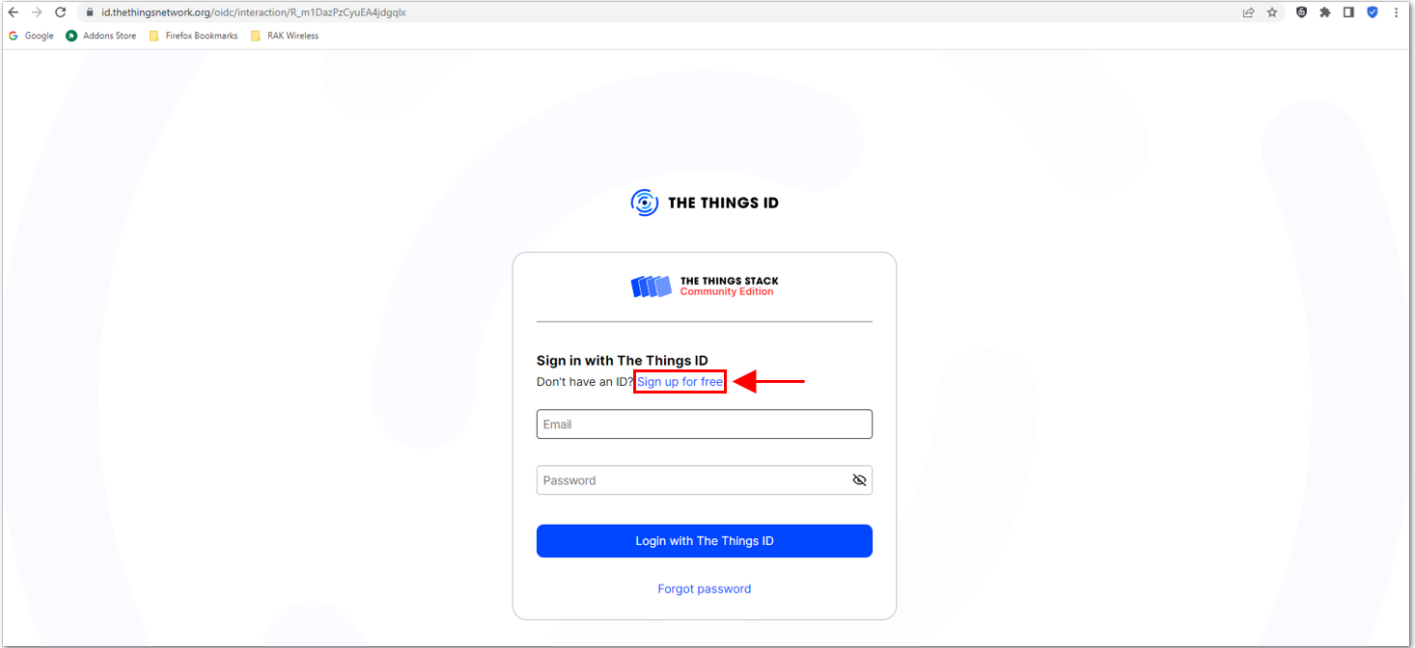


Figure 31: Signing up through the Things ID

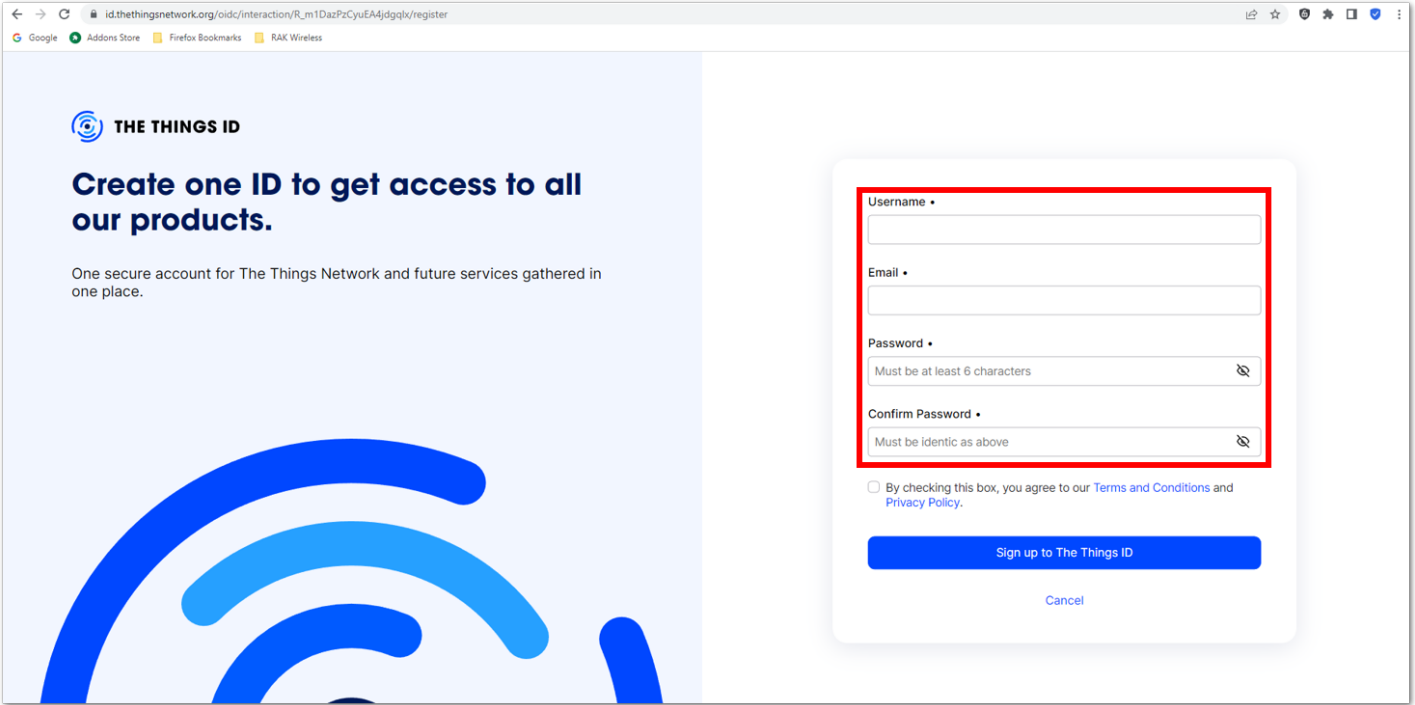


Figure 32: Creation of an account through the Things ID

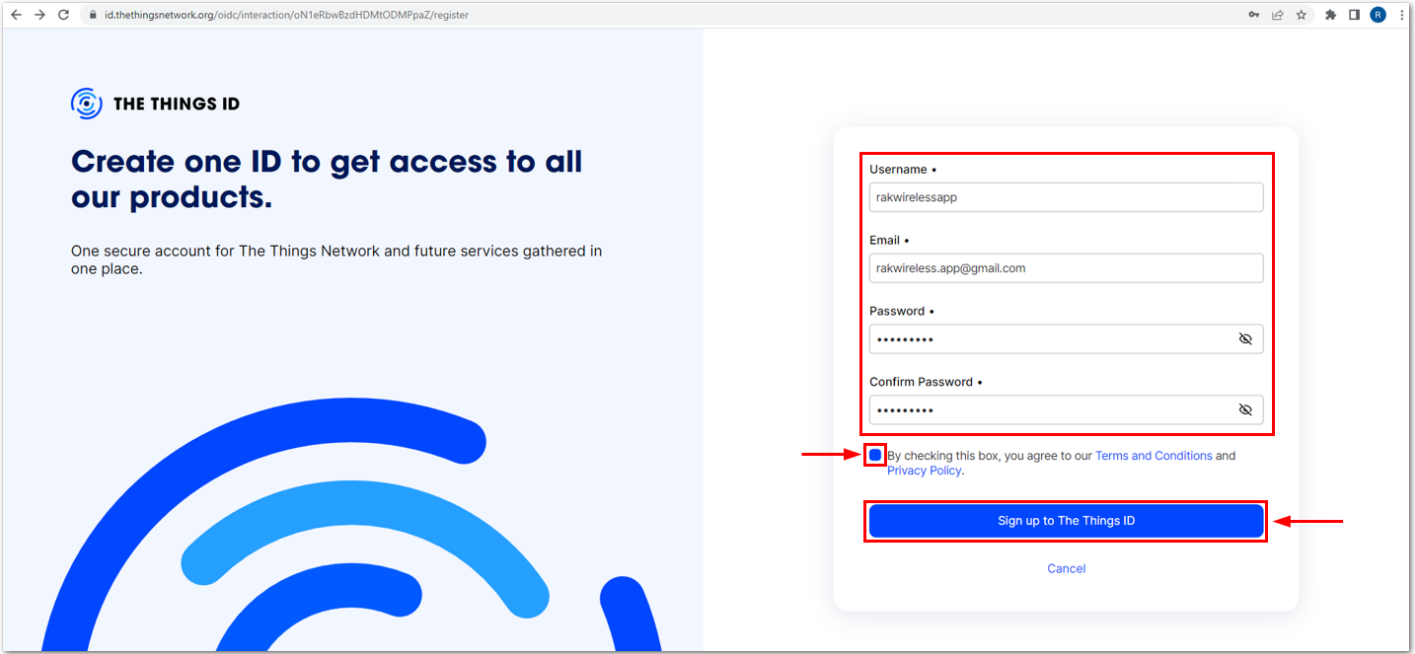


Figure 33: Creation of an account through the Things ID

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**.

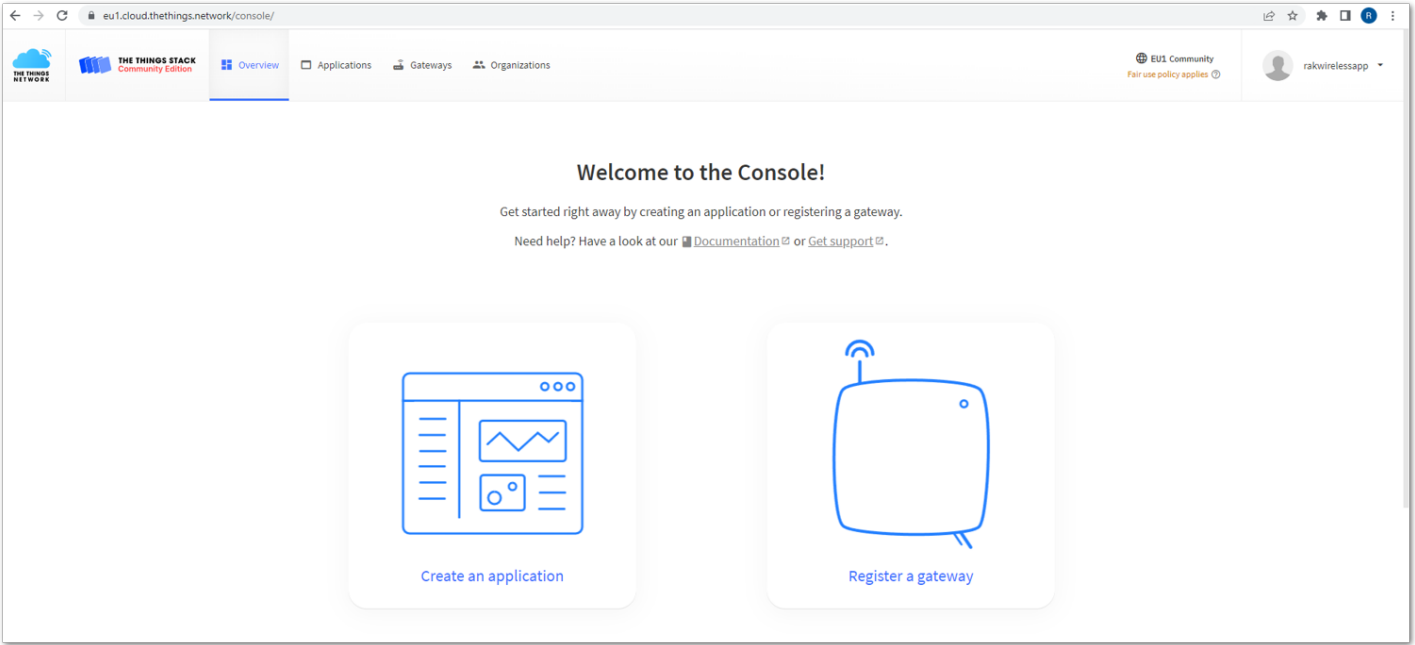


Figure 34: The Things Stack Platform

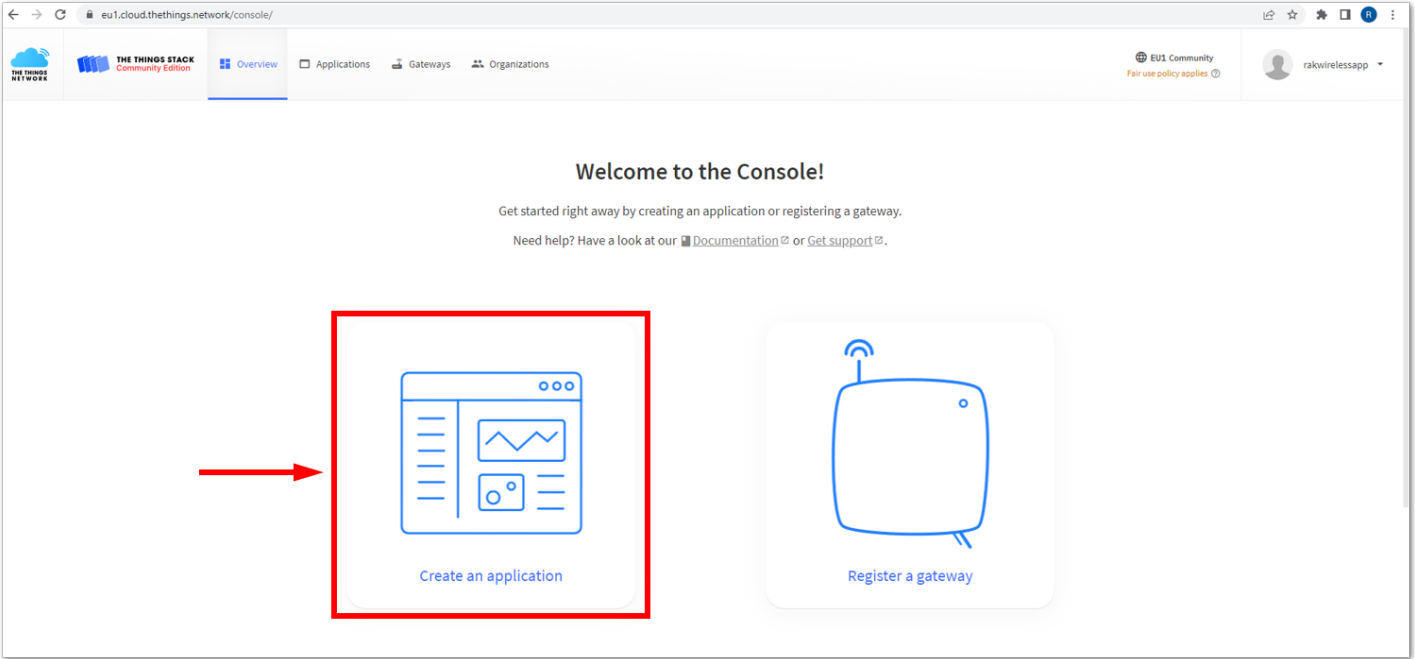


Figure 35: Creating TTN application for your LoRaWAN devices

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

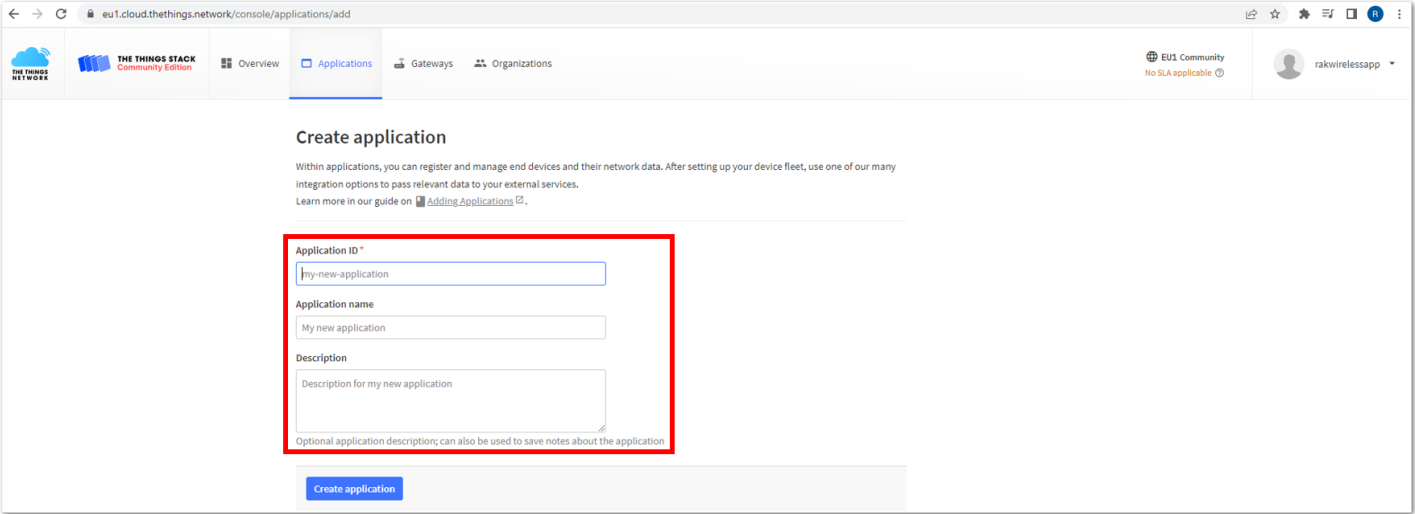


Figure 36: Details of the TTN application

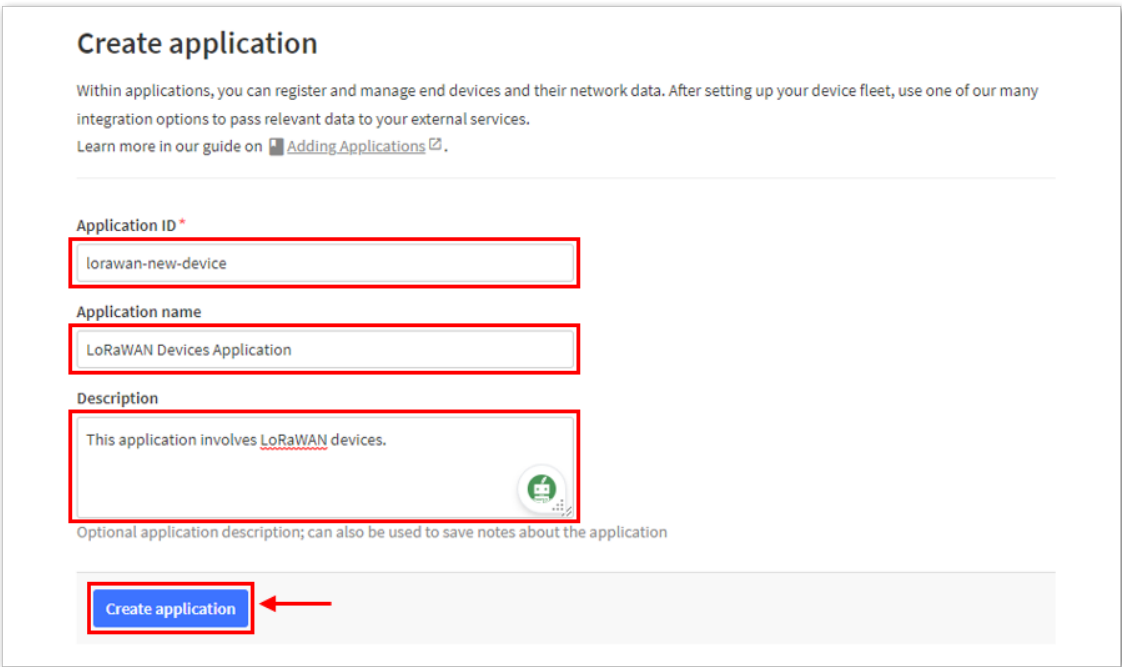


Figure 37: Details of the TTN application

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 38**.

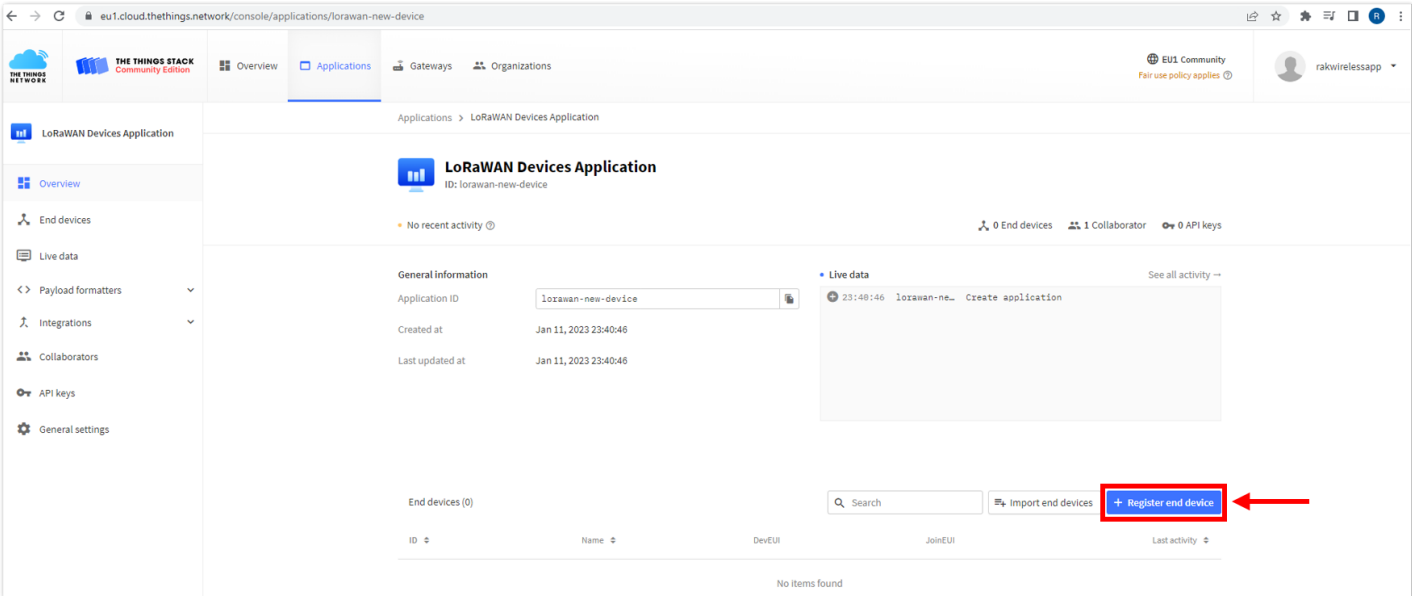


Figure 38: Register end device

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

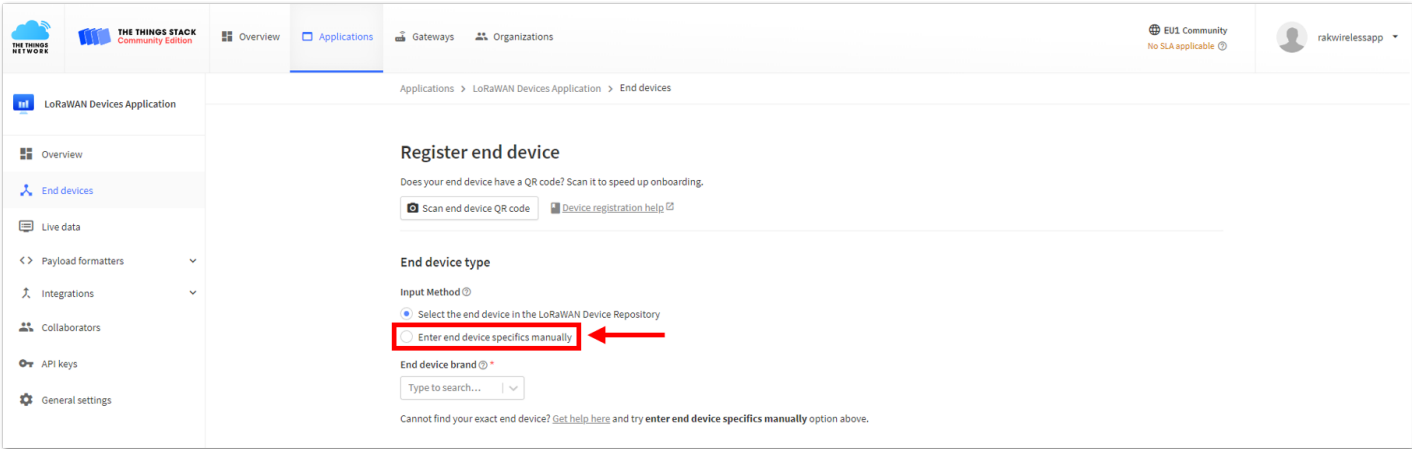


Figure 39: Enter end device specifics manually

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... | v

LoRaWAN version ⓘ *

Select... | v

Regional Parameters version ⓘ *

Select... | v

To continue, please enter versions and frequency plan information

Figure 40: 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) | v

LoRaWAN version ⓘ *

Select... | v

Regional Parameters version ⓘ *

Select... | v

To continue, please enter versions and frequency plan information

Figure 41: 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) | v

LoRaWAN version ⓘ *

LoRaWAN Specification 1.0.3 | v

Regional Parameters version ⓘ *

RP001 Regional Parameters 1.0.3 revision A | v

[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 42: 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 43: 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 44: 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 45: 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 **AppKey** 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 RAK3272S 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 00 00 00 00 00 00 00 00

Generate

0/50 used

AppKey ⓘ *

.. .. .

Generate

End device ID ⓘ *

eui-ac1f09ffe053776

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 46: Setting up for your device



Figure 47: Setting up for your device



Figure 48: 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 49**.

 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 36**.

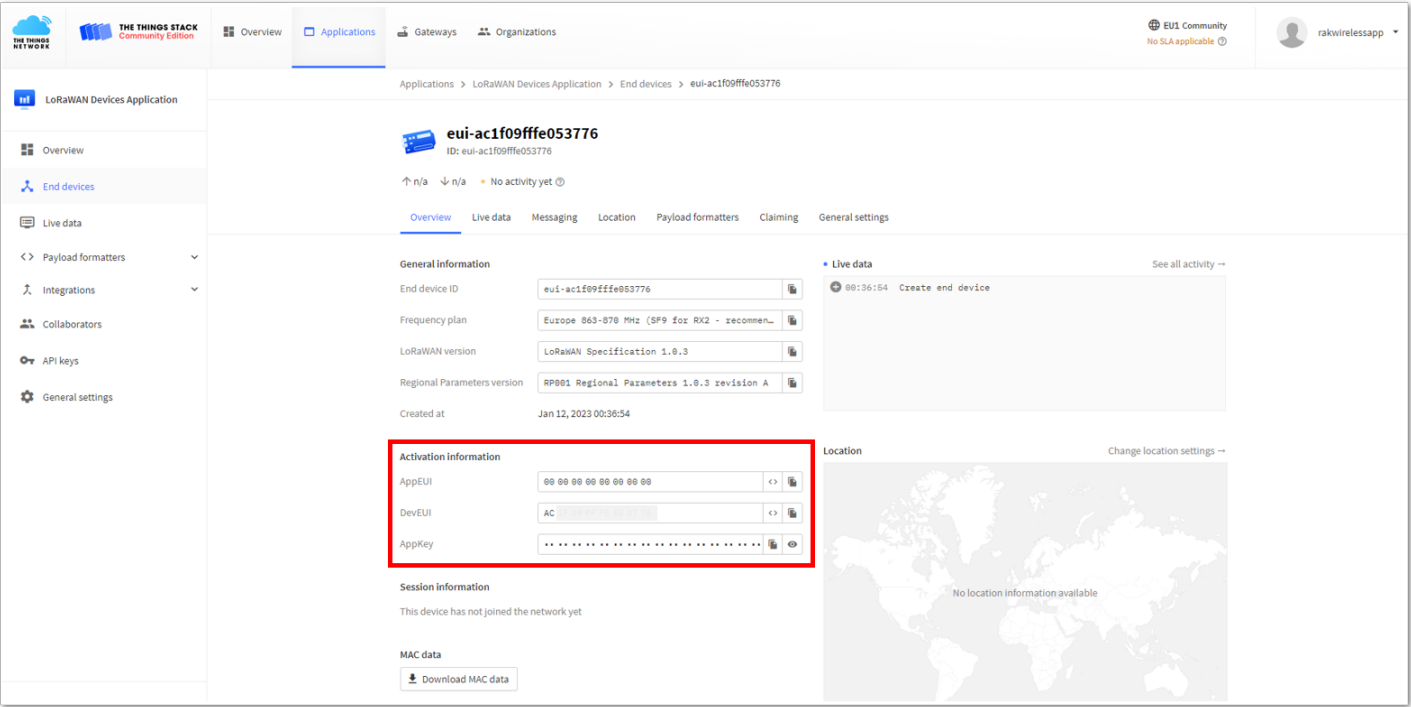


Figure 49: OTAA device successfully registered to TTN

OTAA Configuration for TTN

The RAK3272 Breakout Board which has a RAK3172 module in it can be configured using WisToolBox to do the OTAA configuration. **WisToolBox** is a software tool that supports **RAK3172** module. It automatically detects RAK3172 module once connected to PC. Below are the options in WisToolBox that 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 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.

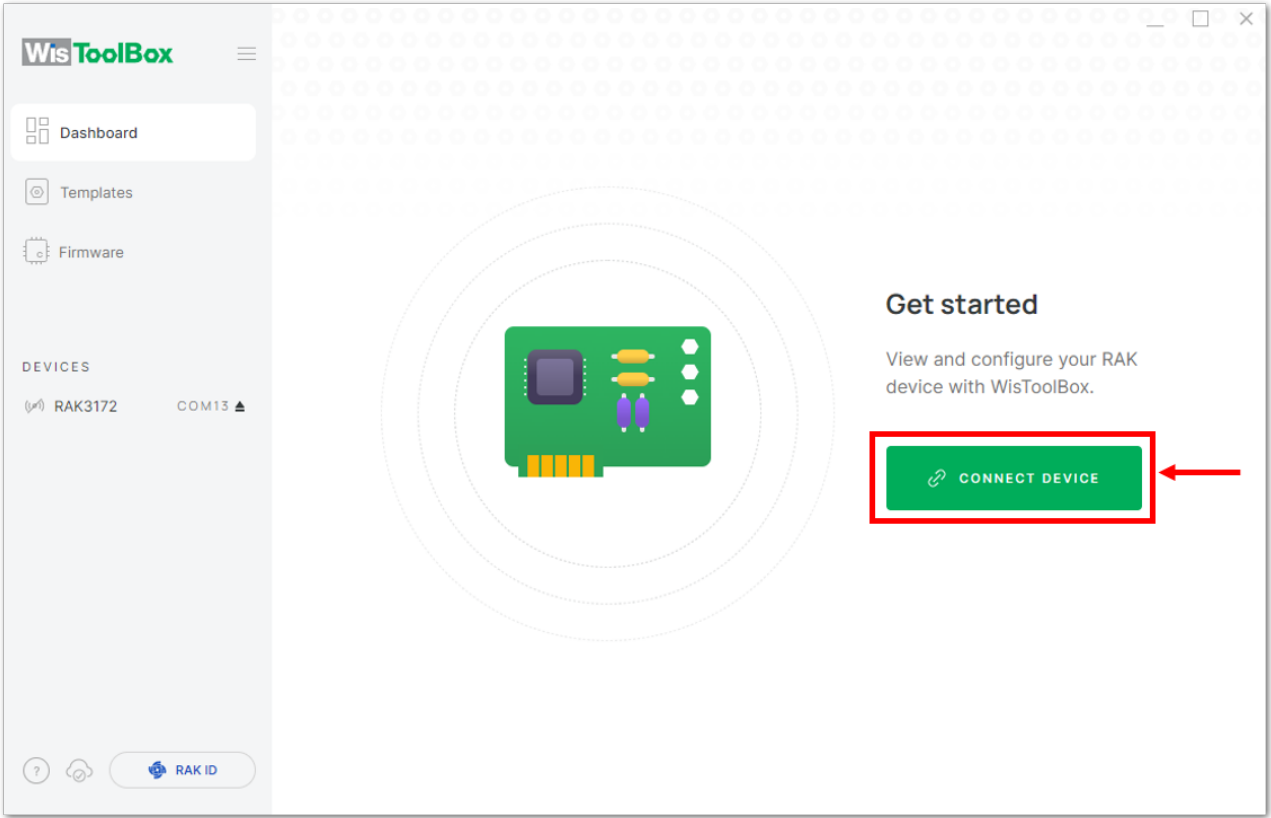


Figure 50: CONNECT DEVICE

3. Then select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT** as shown in **Figure 52**.

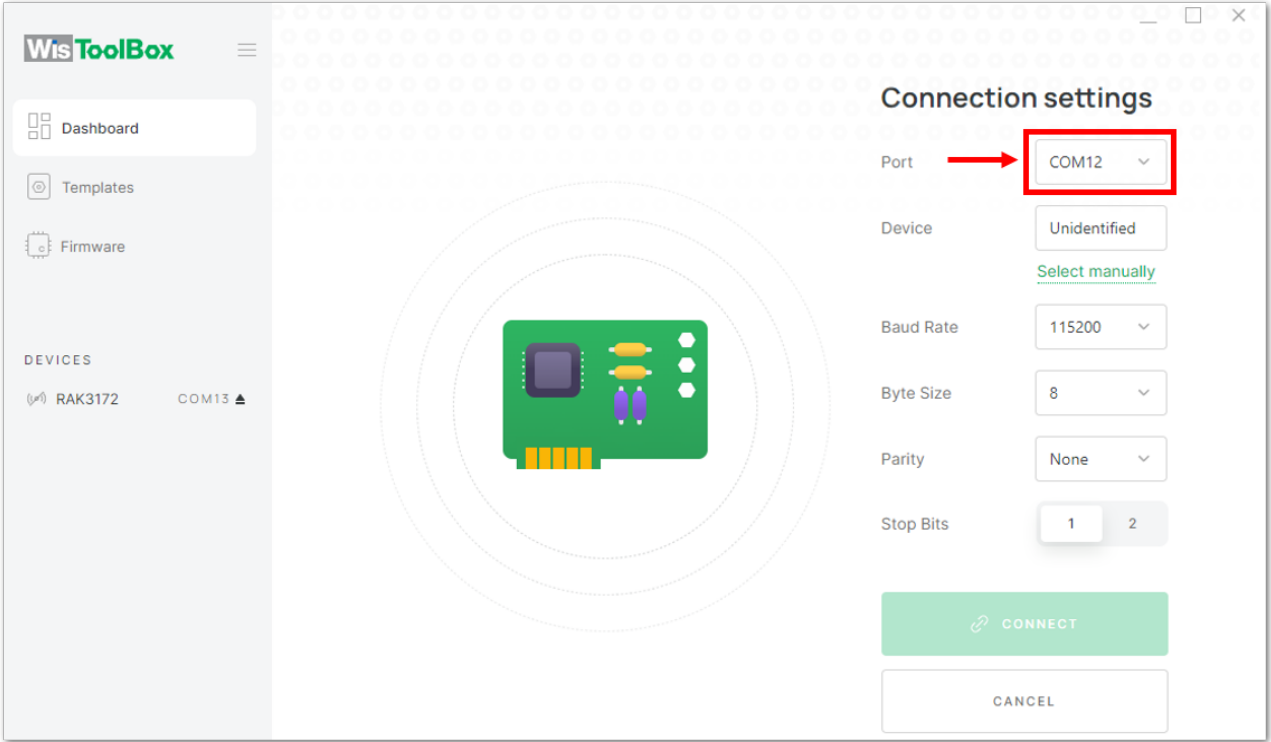


Figure 51: Setting up your device

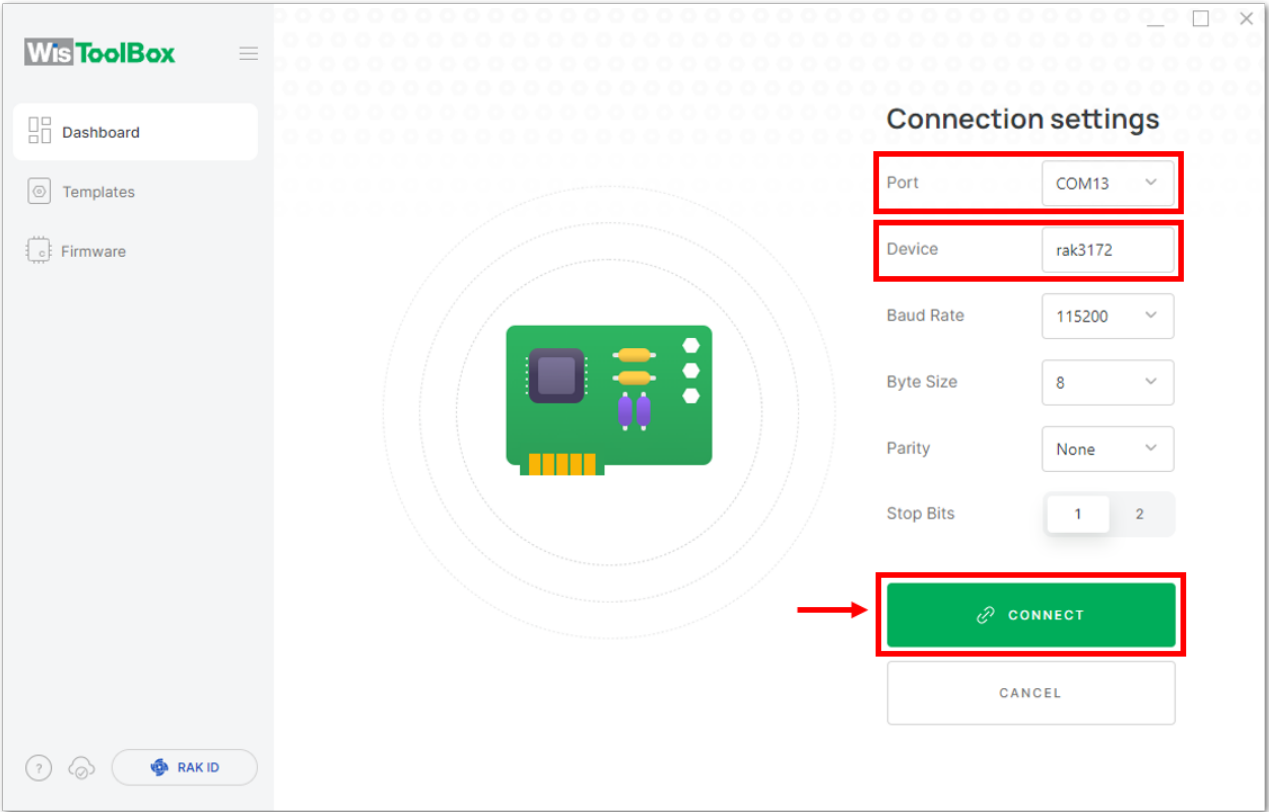


Figure 52: Setting up your device

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

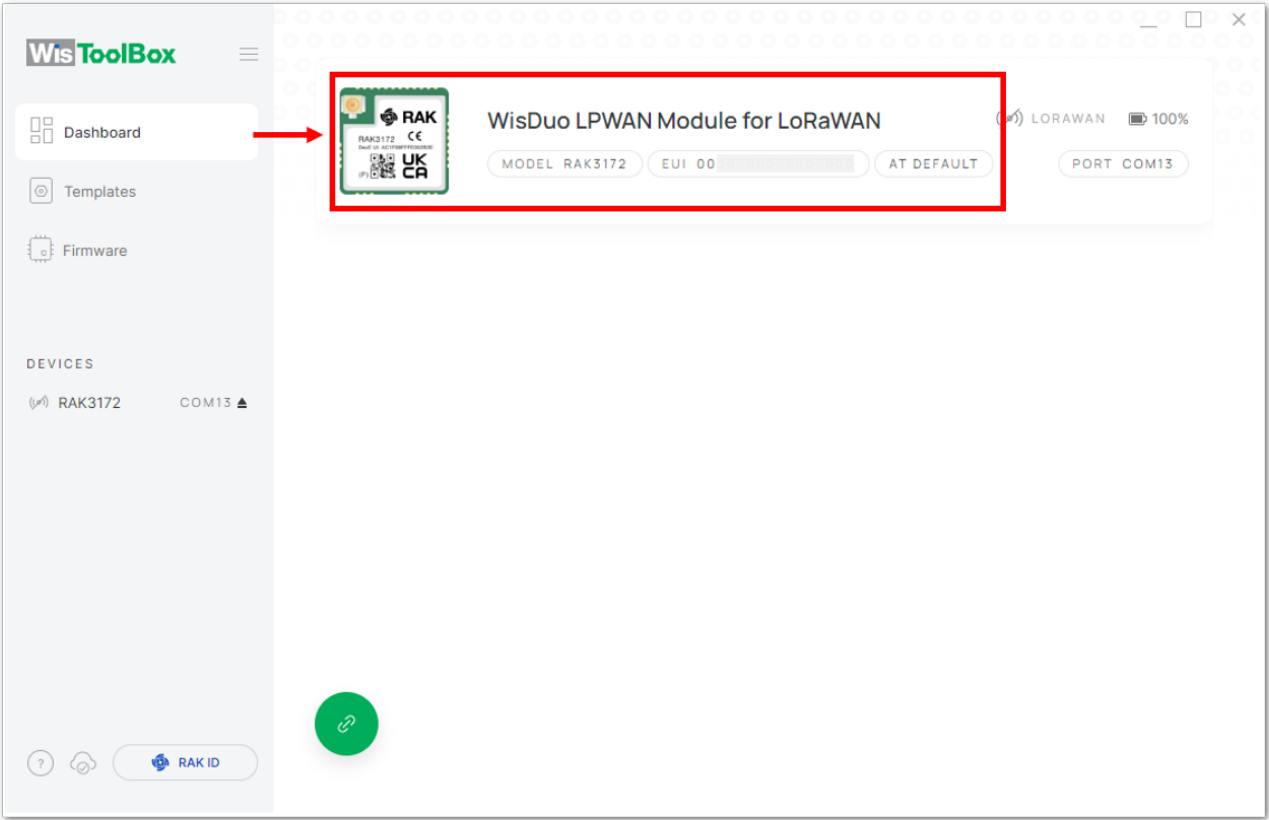


Figure 53: Device seen from WisToolBox dashboard

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

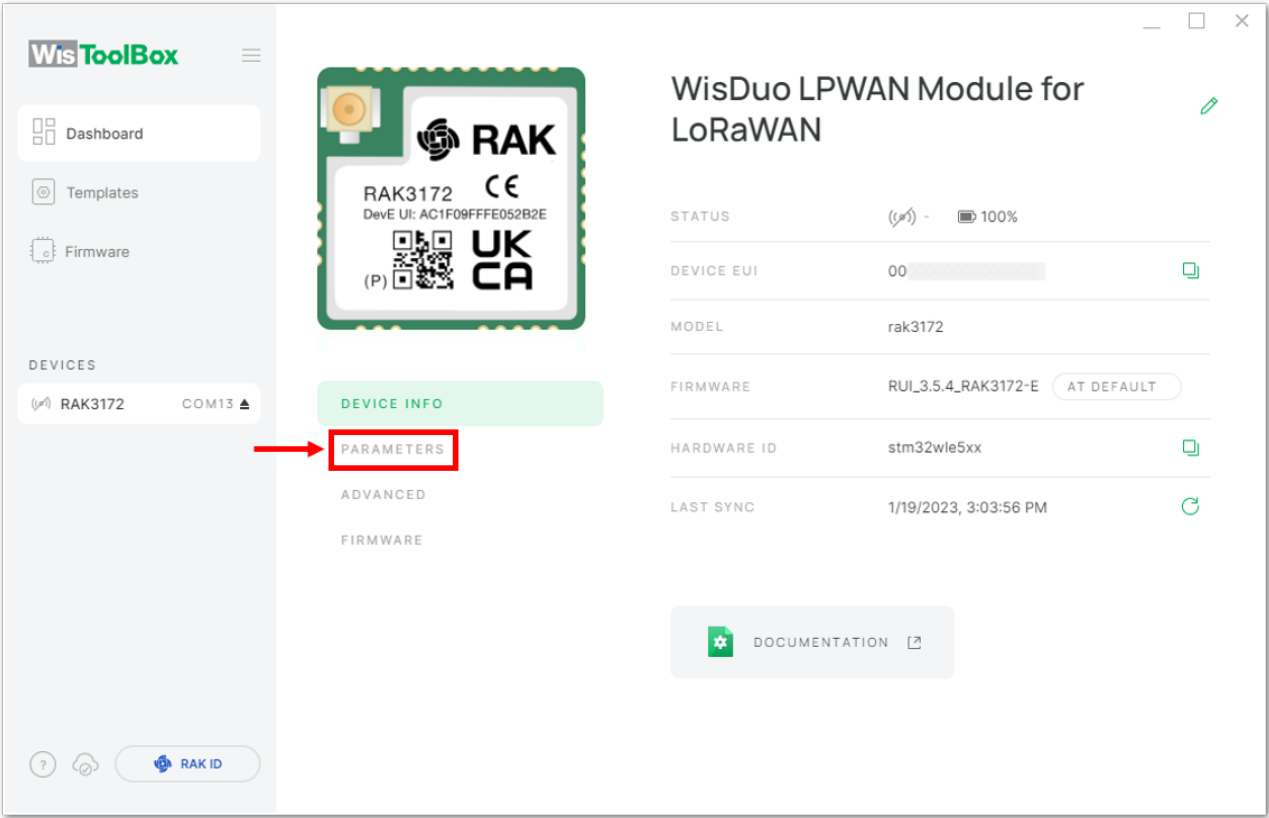


Figure 54: Setting up your device

6. Click **Global settings** to set the network mode into **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 band, you can choose among active regions based on your location.
- LoRa network mode: **LoRaWAN**
 - LoRaWAN join mode: **OTAA**
 - LoRaWAN region: **EU868**

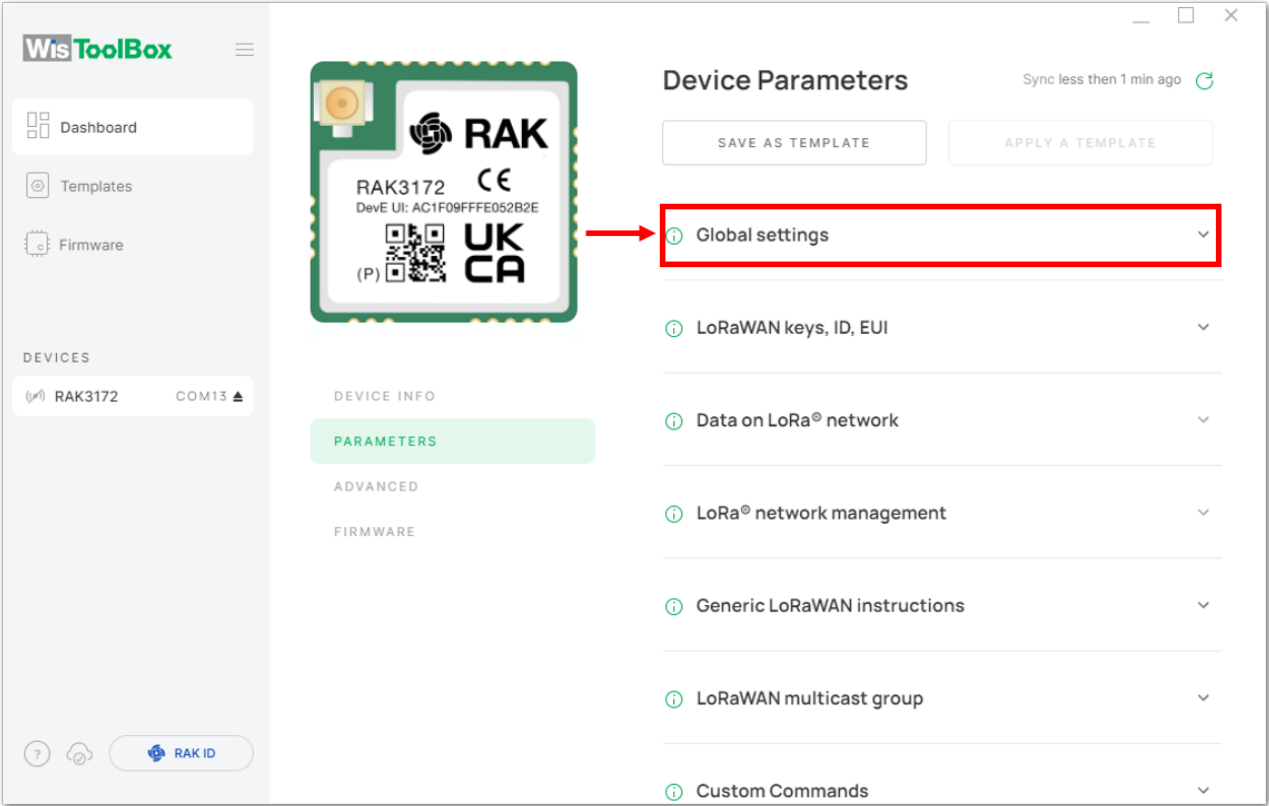


Figure 55: Global settings

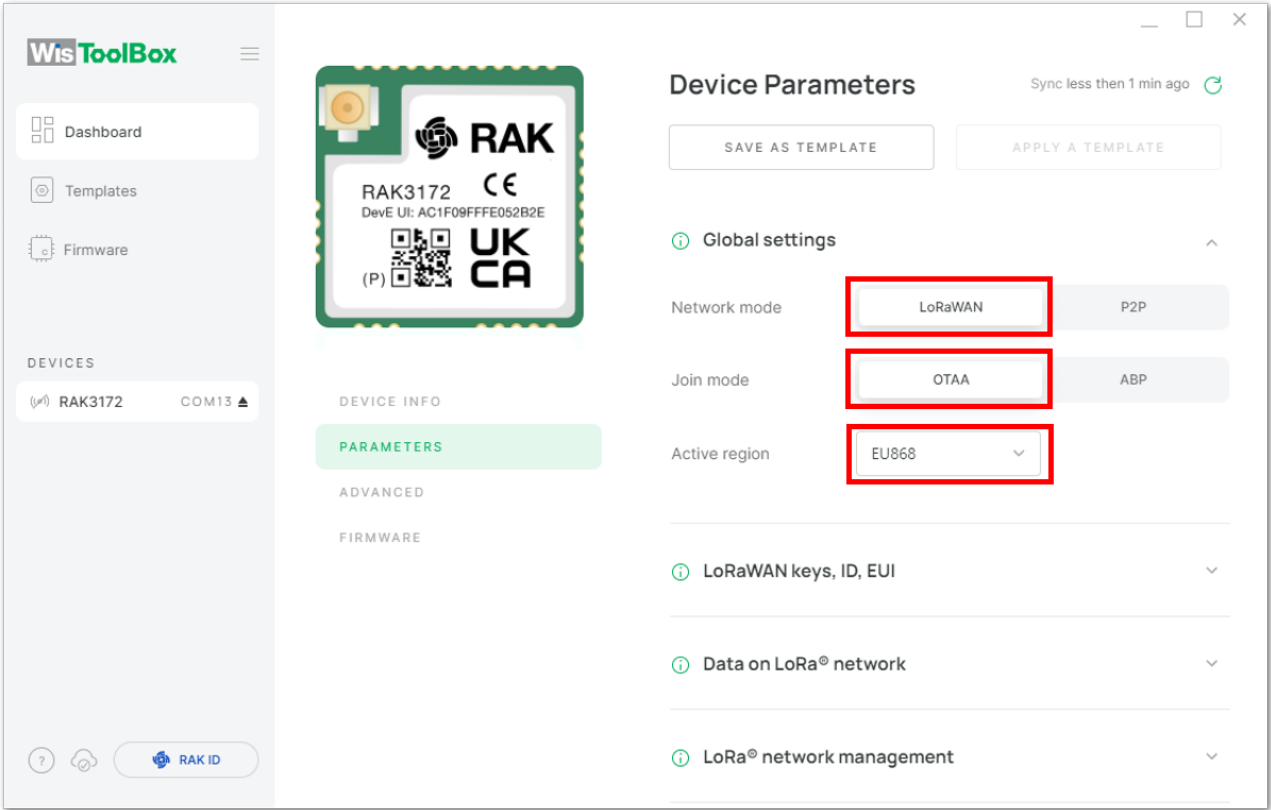


Figure 56: Global settings

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

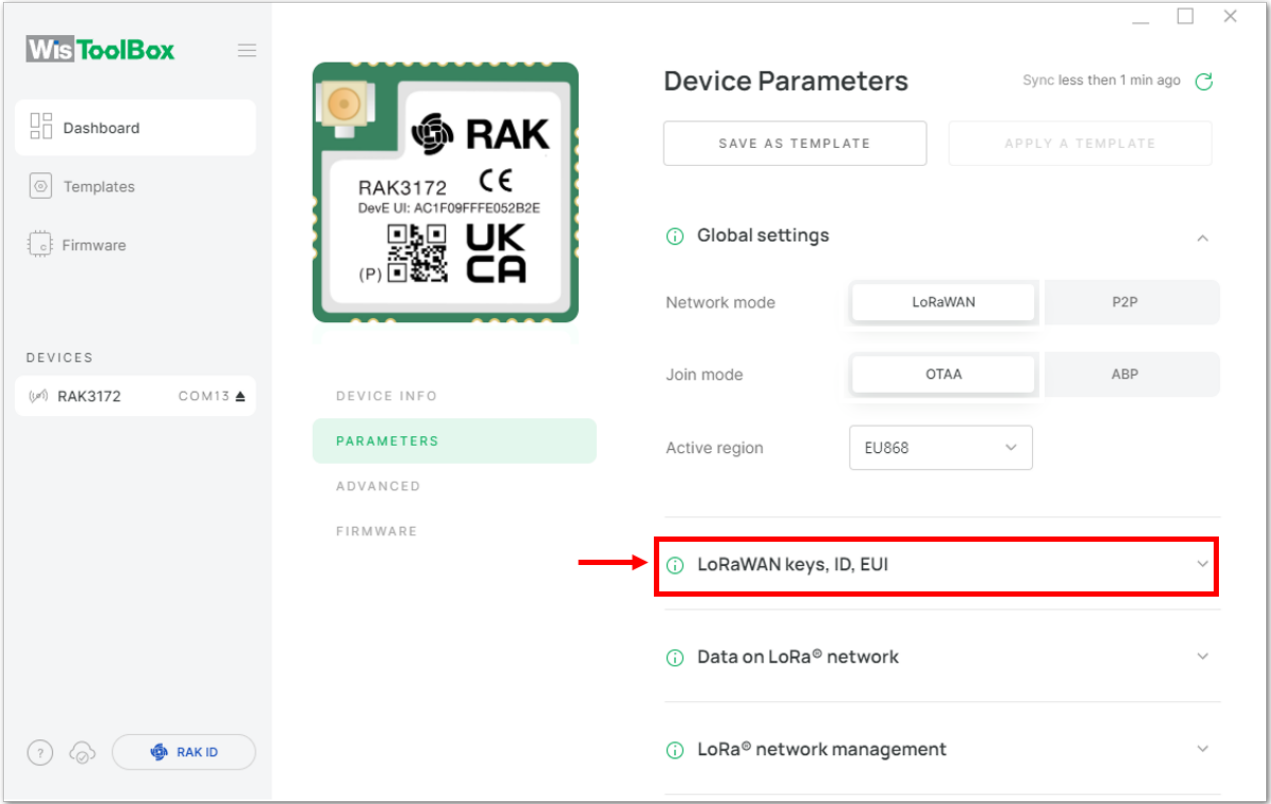


Figure 57: LoRaWAN keys, ID, EUI

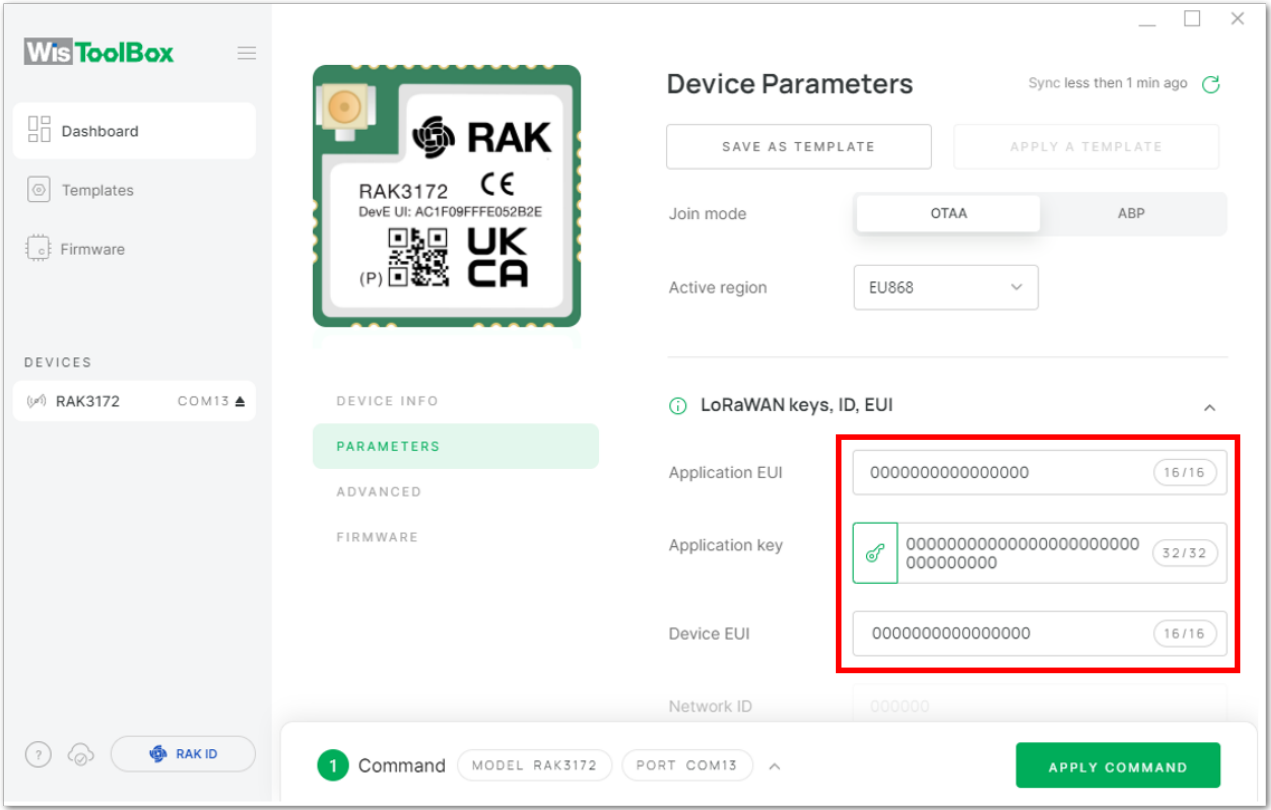


Figure 58: Setting up your device

8. Then go back to 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 COMMAND** to update your device as shown in **Figure 66**.

 **NOTE:**

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

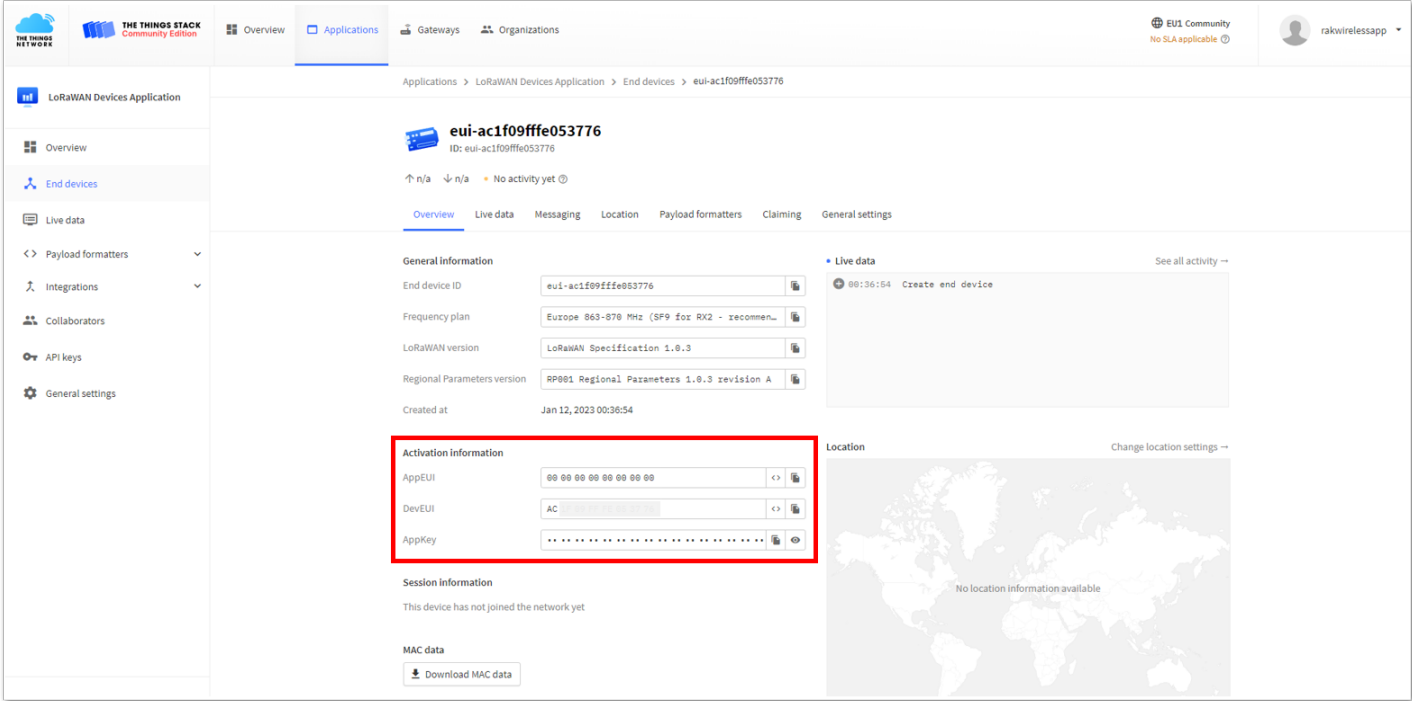


Figure 59: Your created OTAA device from your console

- For Application EUI (AppEUI)

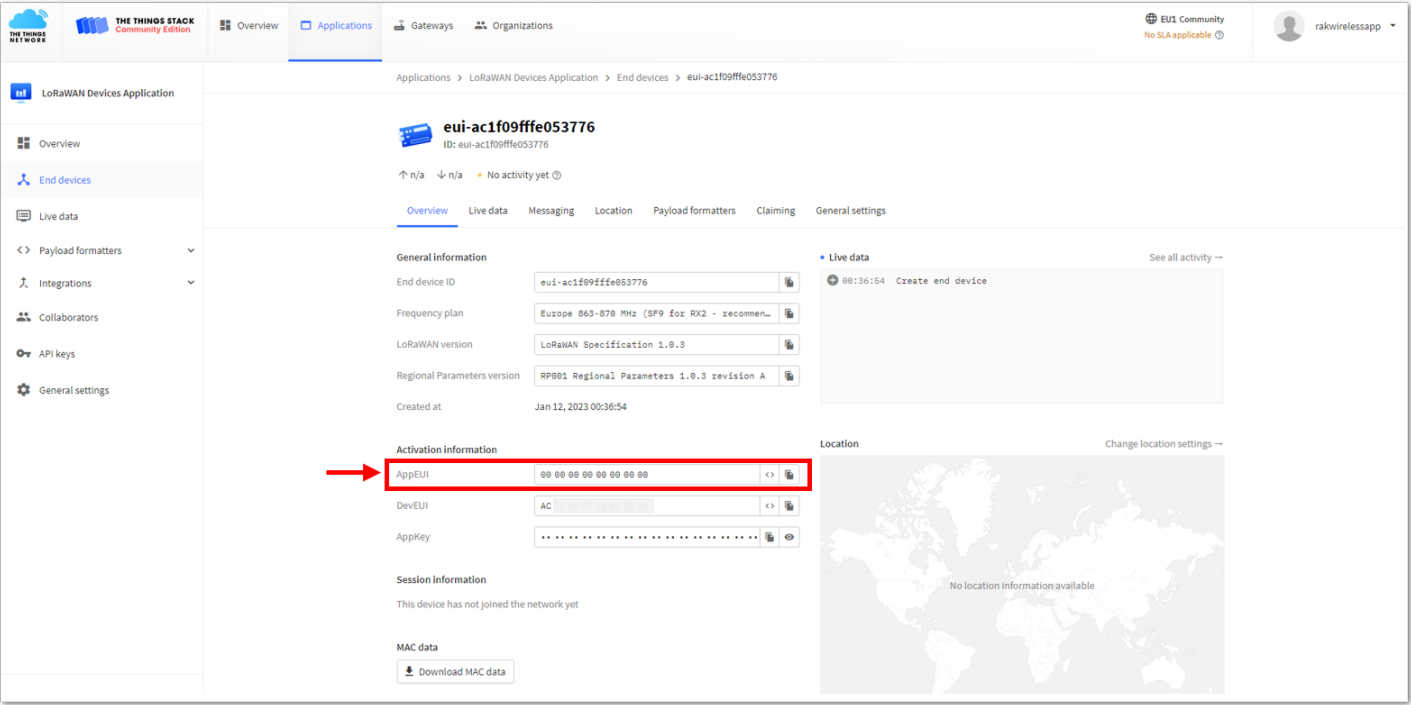


Figure 60: Copying the AppEUI credential from TTN to WisToolBox

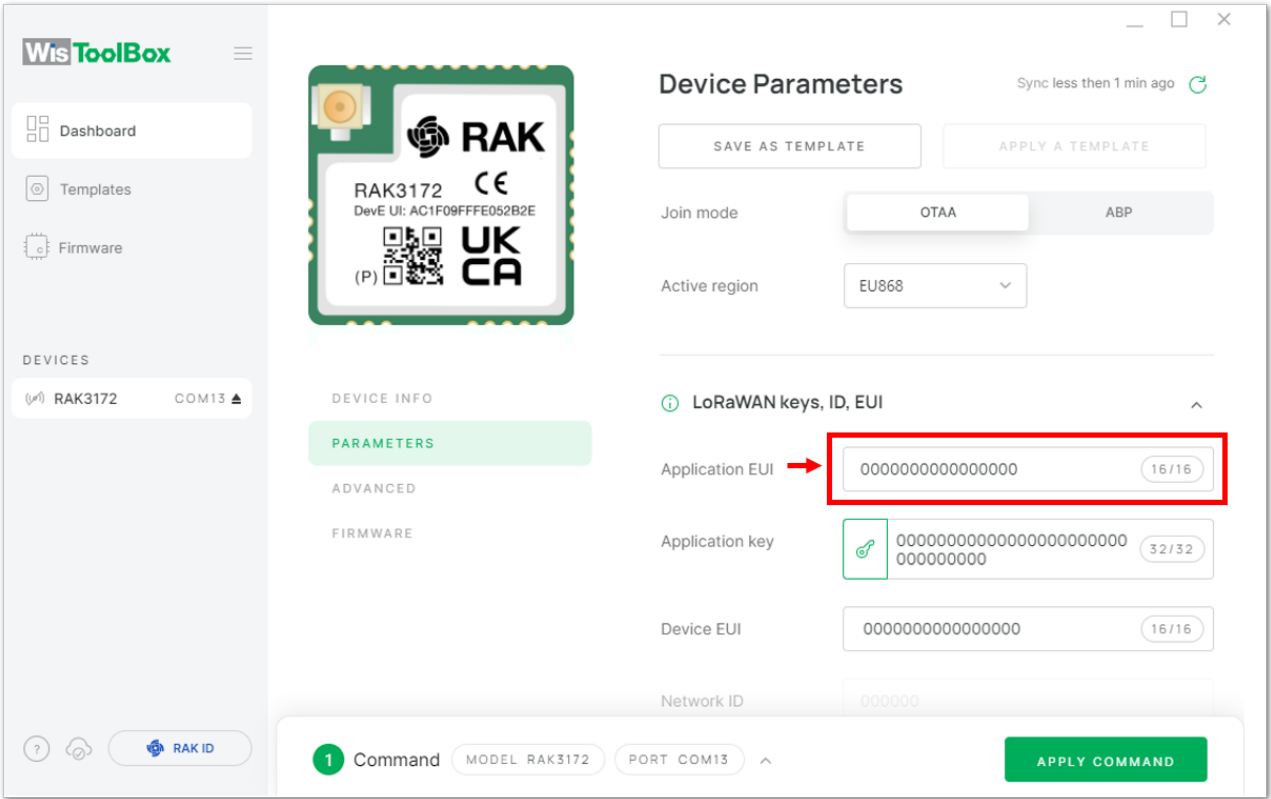


Figure 61: Copying the AppEUI credential from TTN to WisToolBox

- For Application key (AppKey)

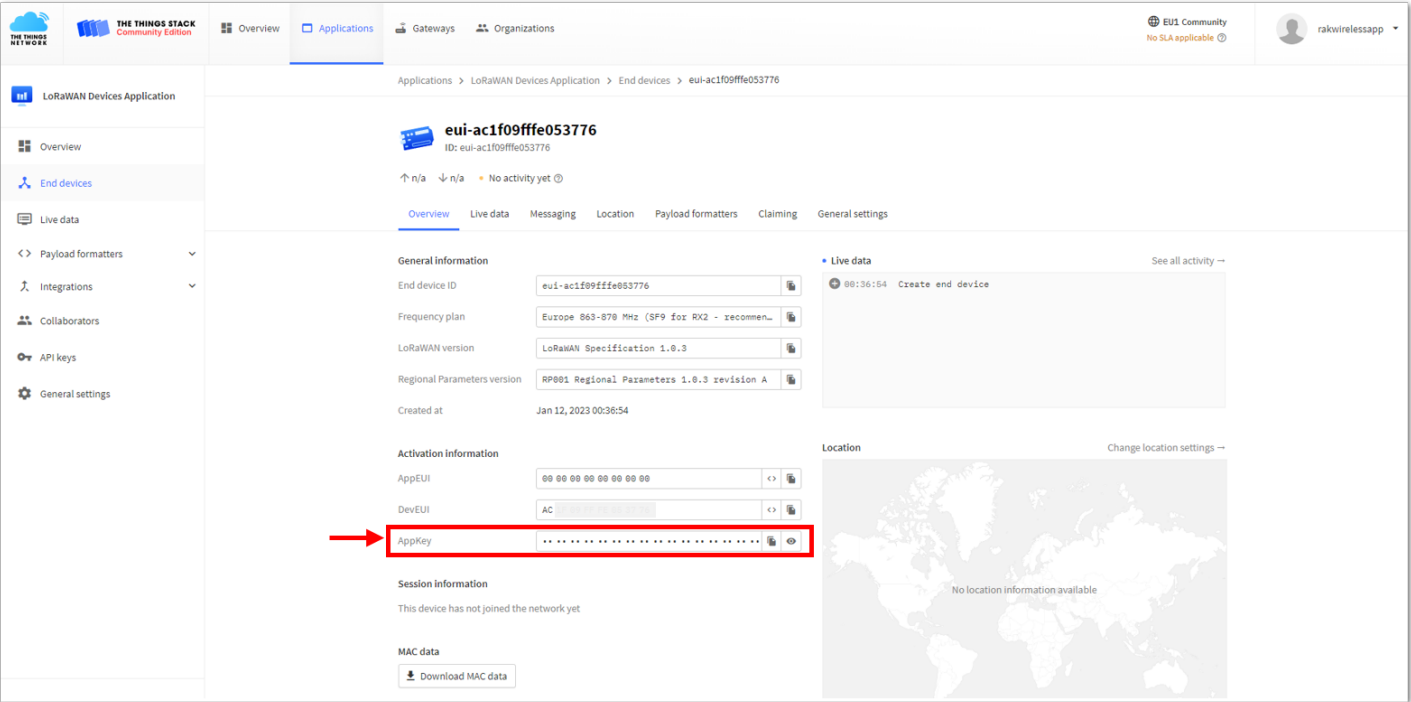


Figure 62: Copying the AppKey credential from TTN to WisToolBox

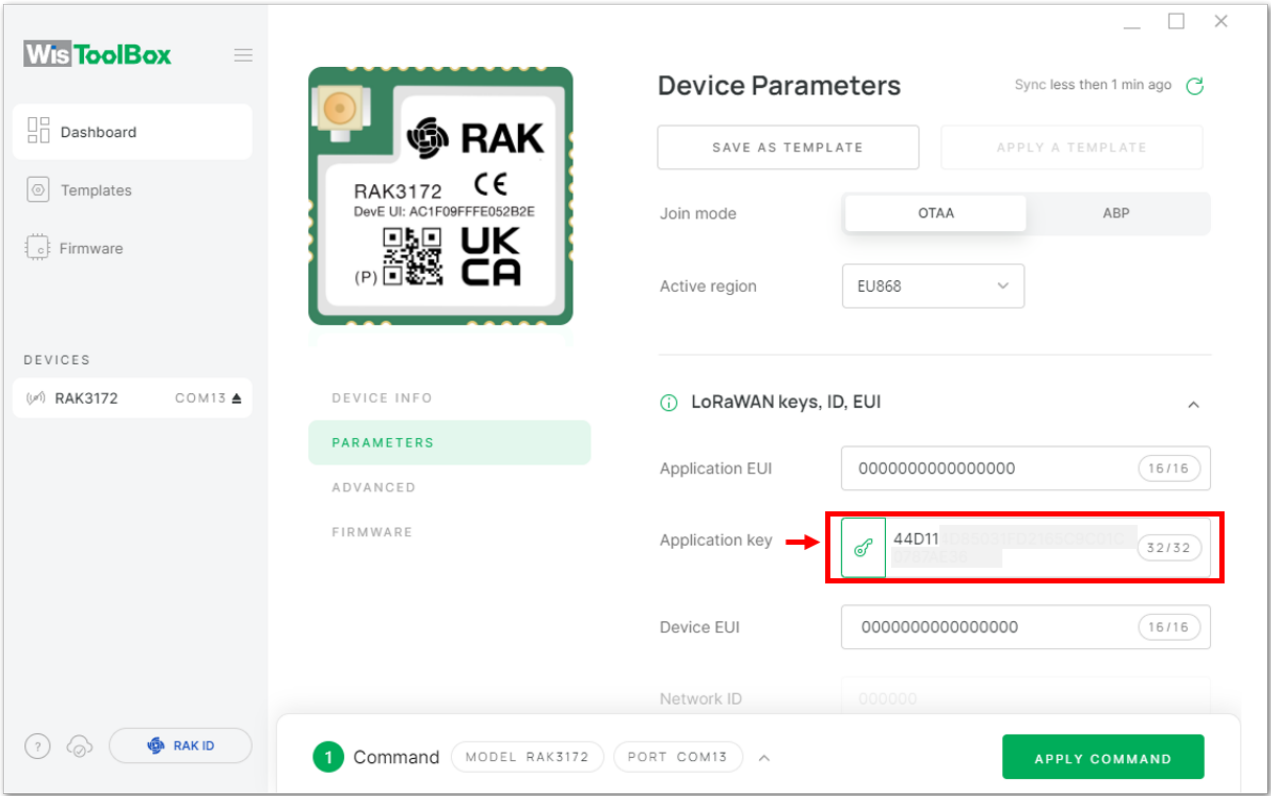


Figure 63: Copying the AppKey credential from TTN to WisToolBox

- For Device EUI (DevEUI)

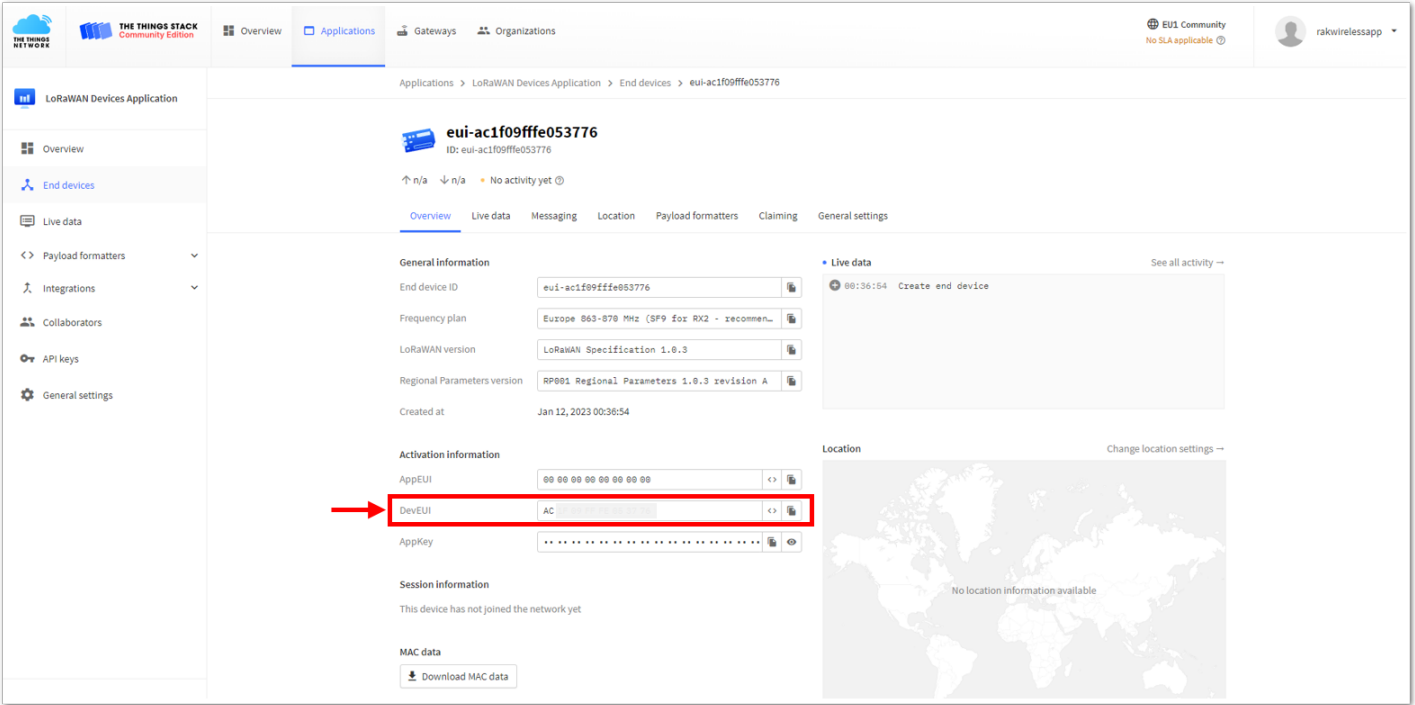


Figure 64: Copying the DevEUI credential from TTN to WisToolBox

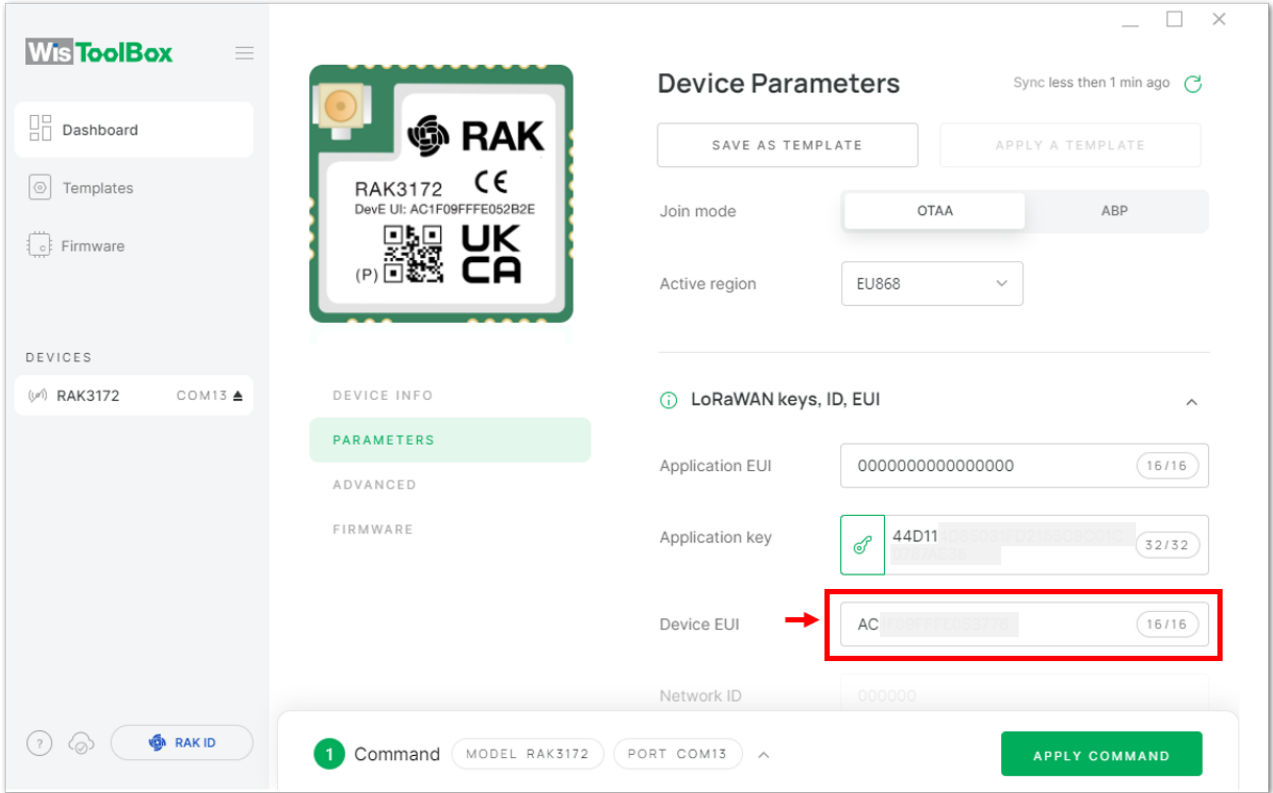


Figure 65: Copying the DevEUI credential from TTN to WisToolBox

• WisToolBox Dashboard

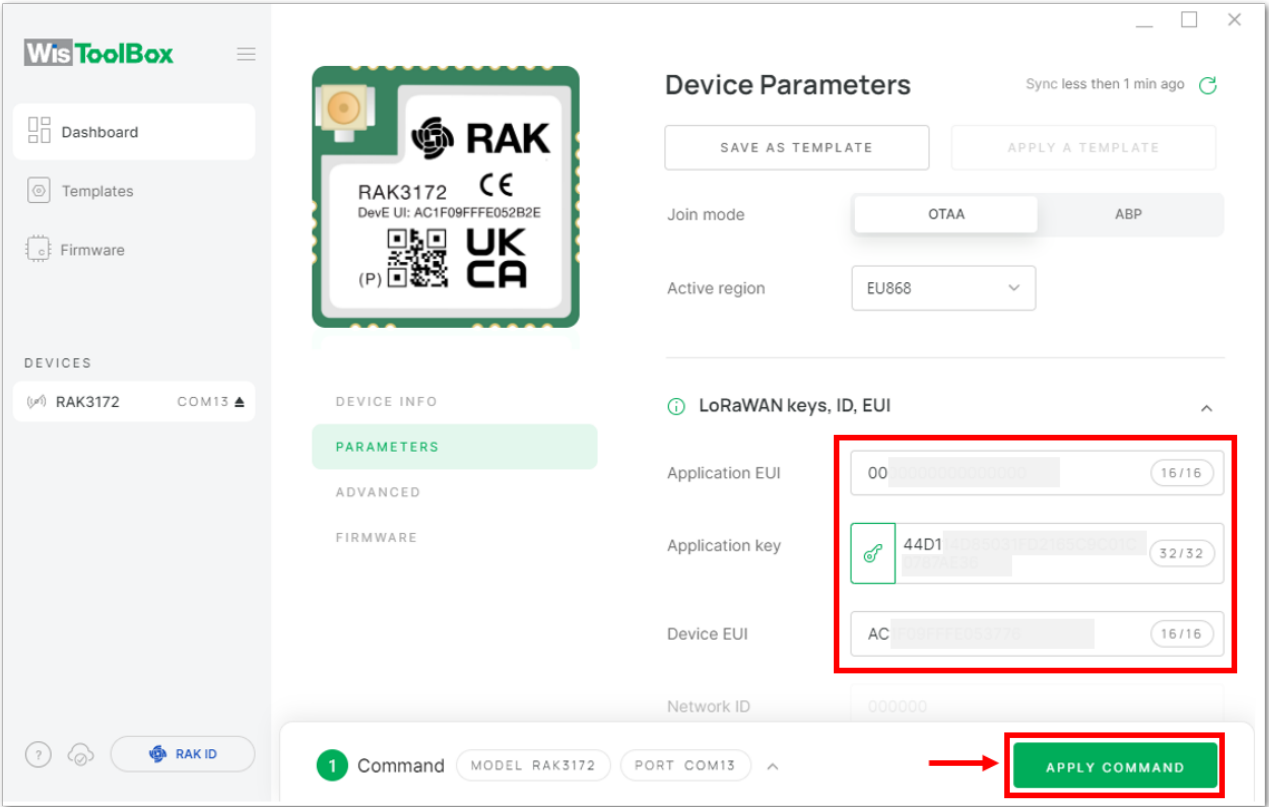


Figure 66: Used credentials from your console in WisToolBox dashboard

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

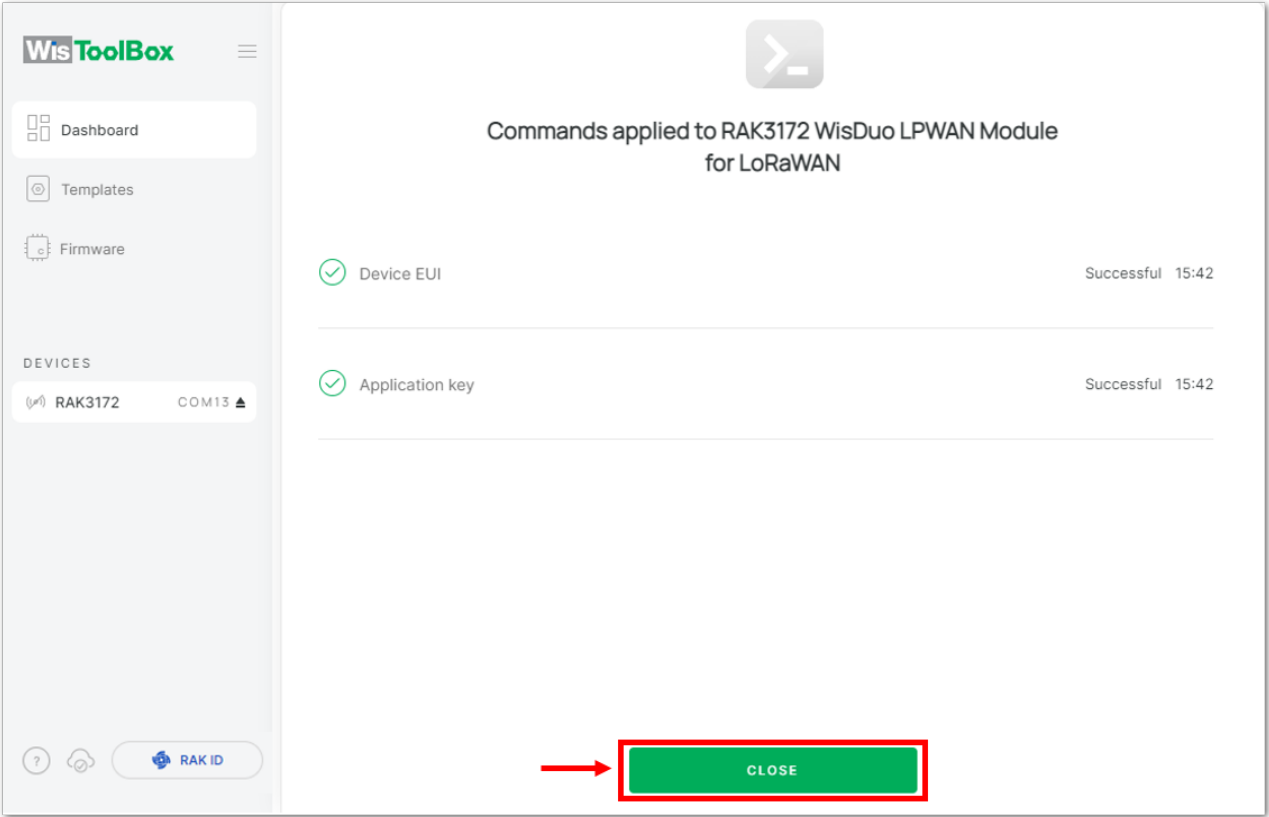


Figure 67: Summary of commands

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

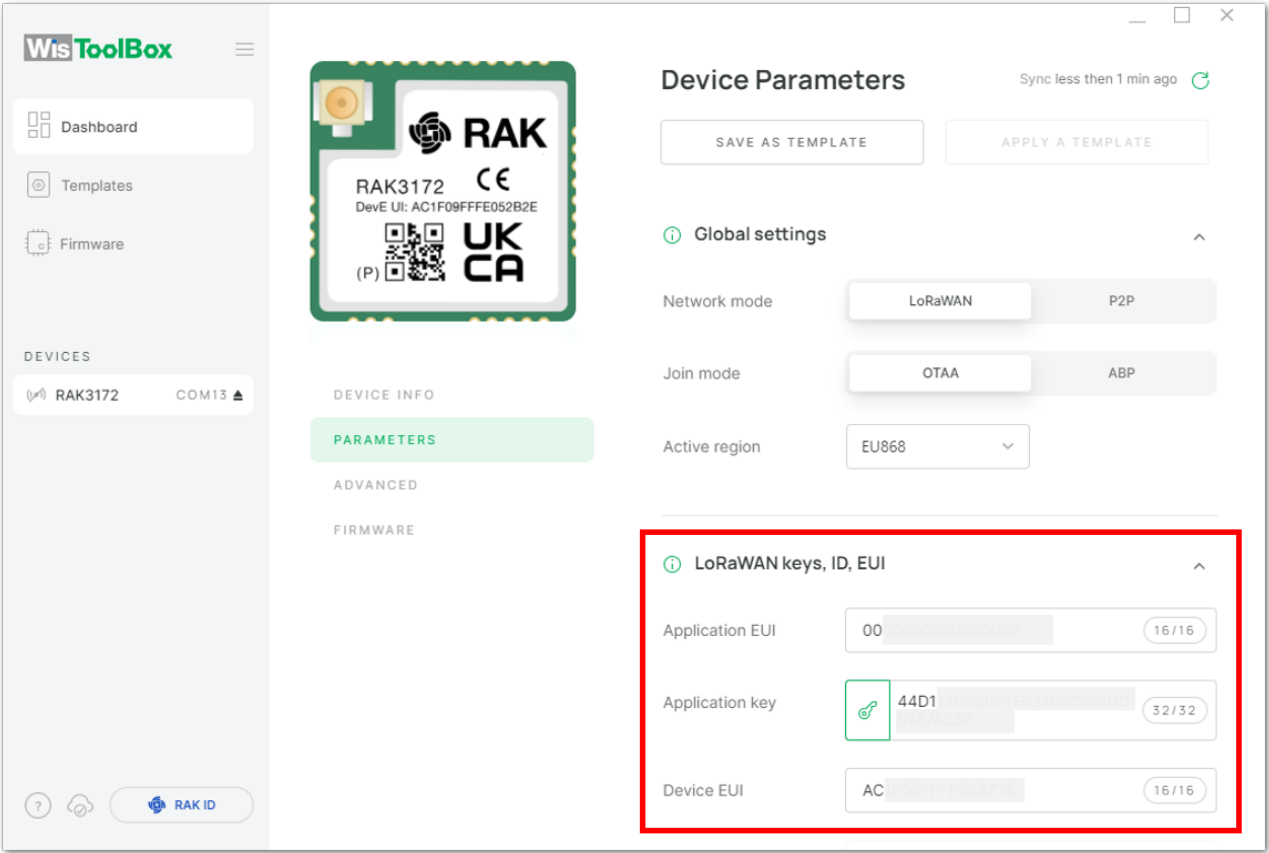


Figure 68: Successfully configured OTAA device via WisToolBox dashboard

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

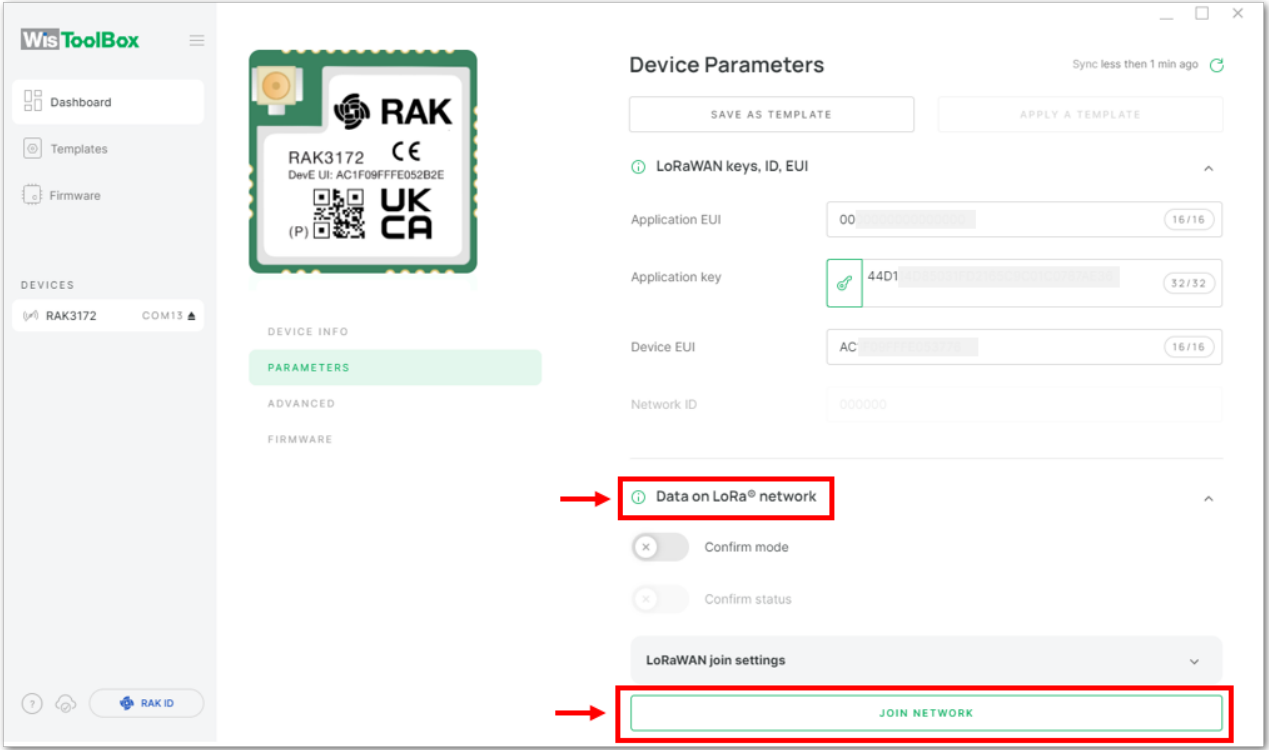


Figure 69: Joining mode of your OTAA device

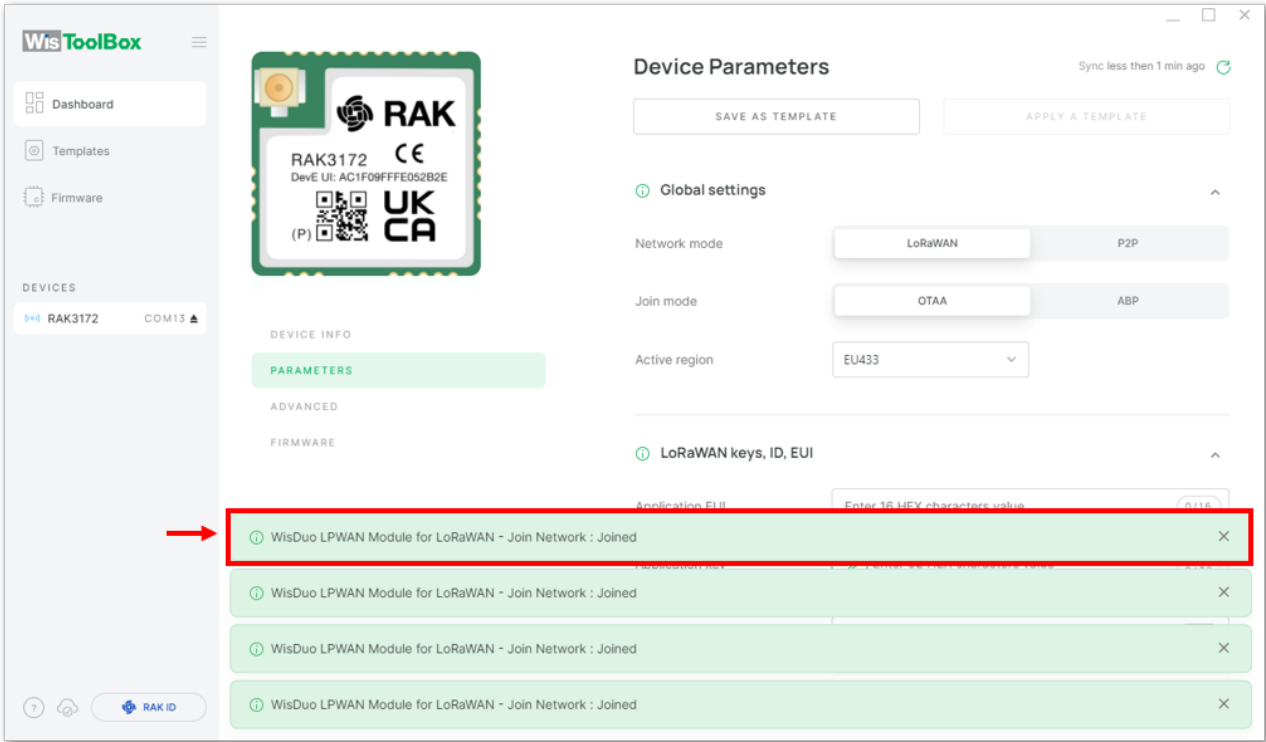


Figure 70: OTAA device successfully joined the TTN server

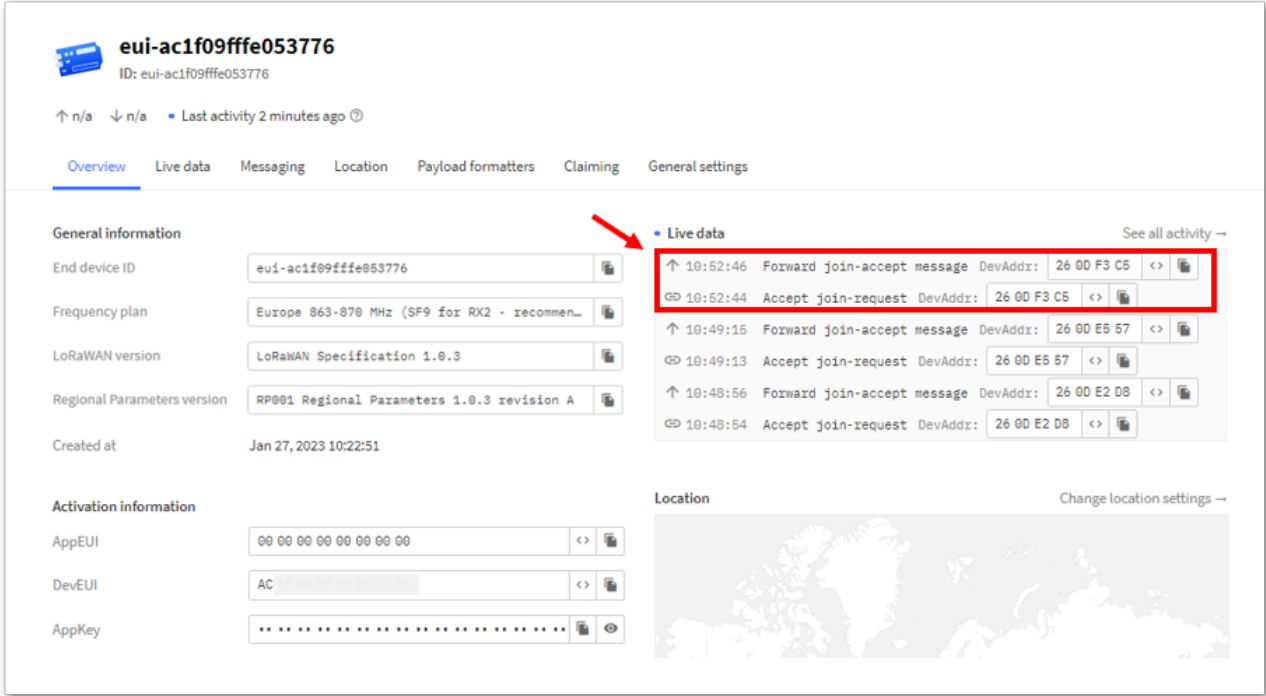


Figure 71: OTAA device successfully joined the TTN server

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.

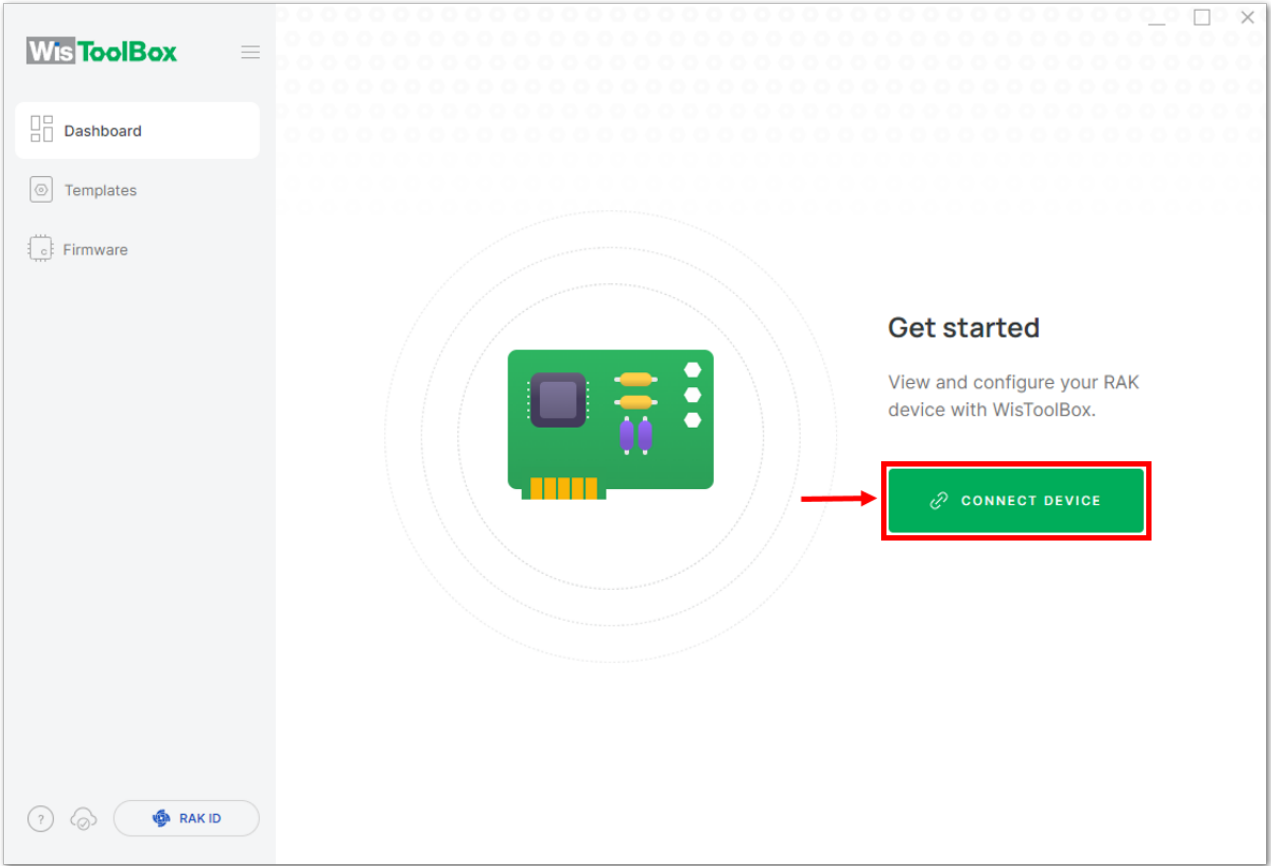


Figure 72: CONNECT DEVICE

3. Then select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT** as shown in **Figure 74**.

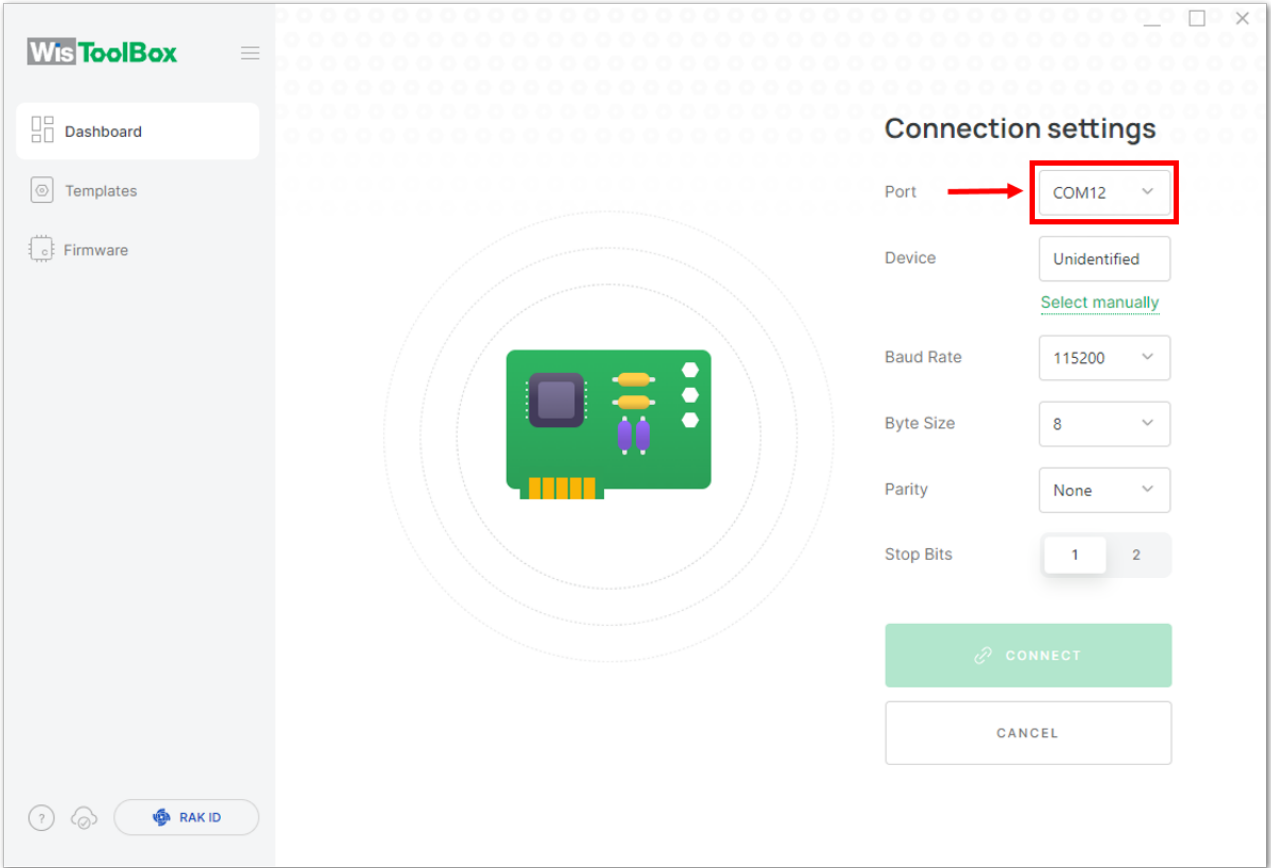


Figure 73: Setting up your device

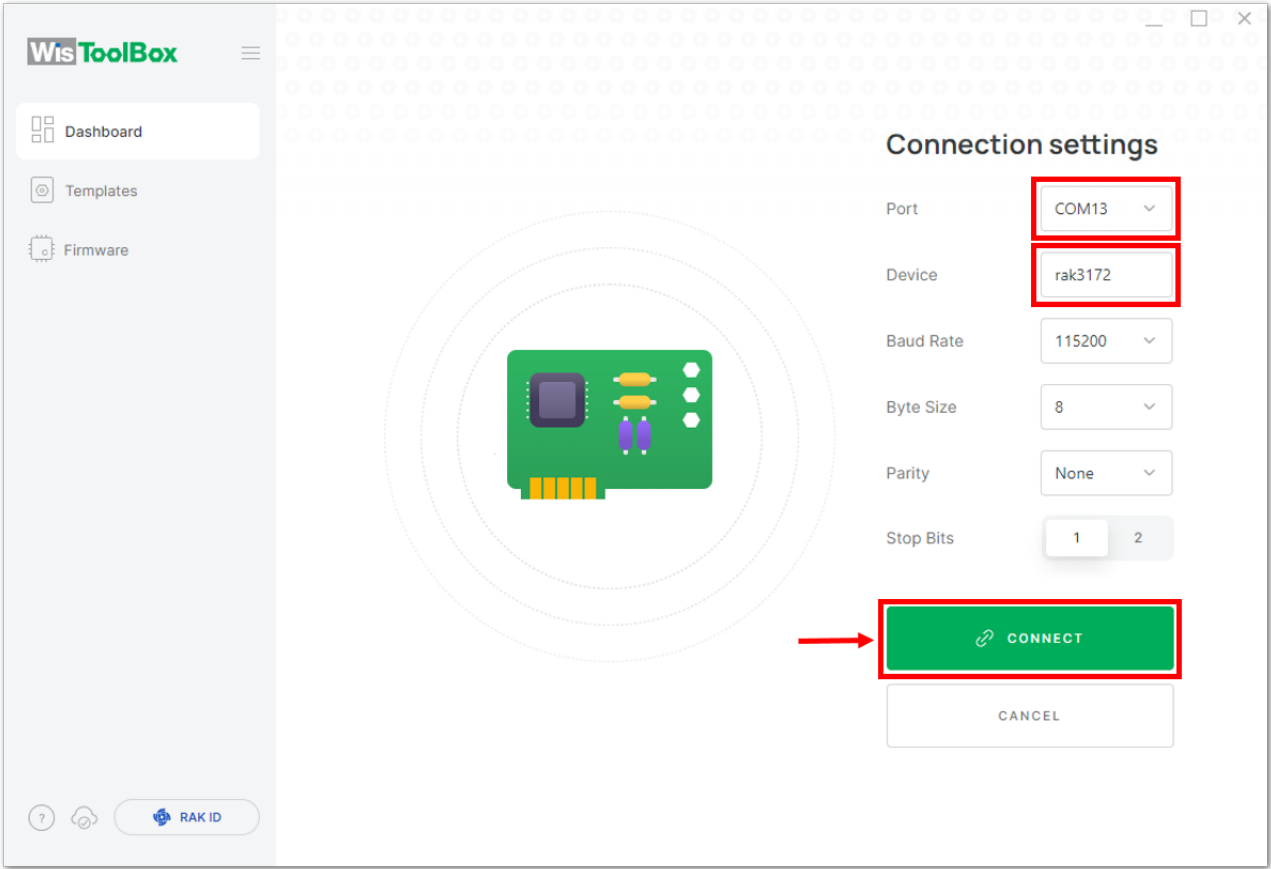


Figure 74: Setting up your device

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

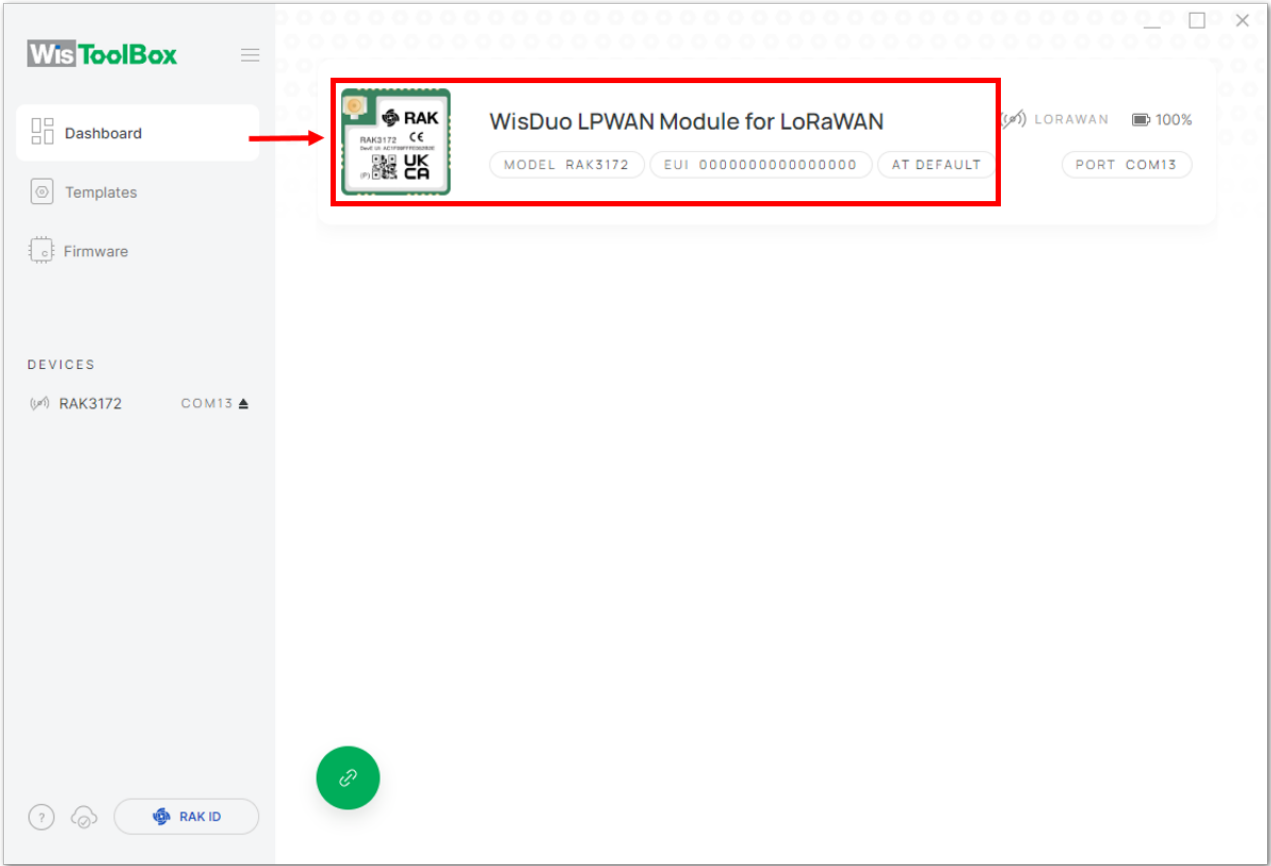


Figure 75: Device seen from WisToolBox dashboard

5. Then click **ADVANCED**.

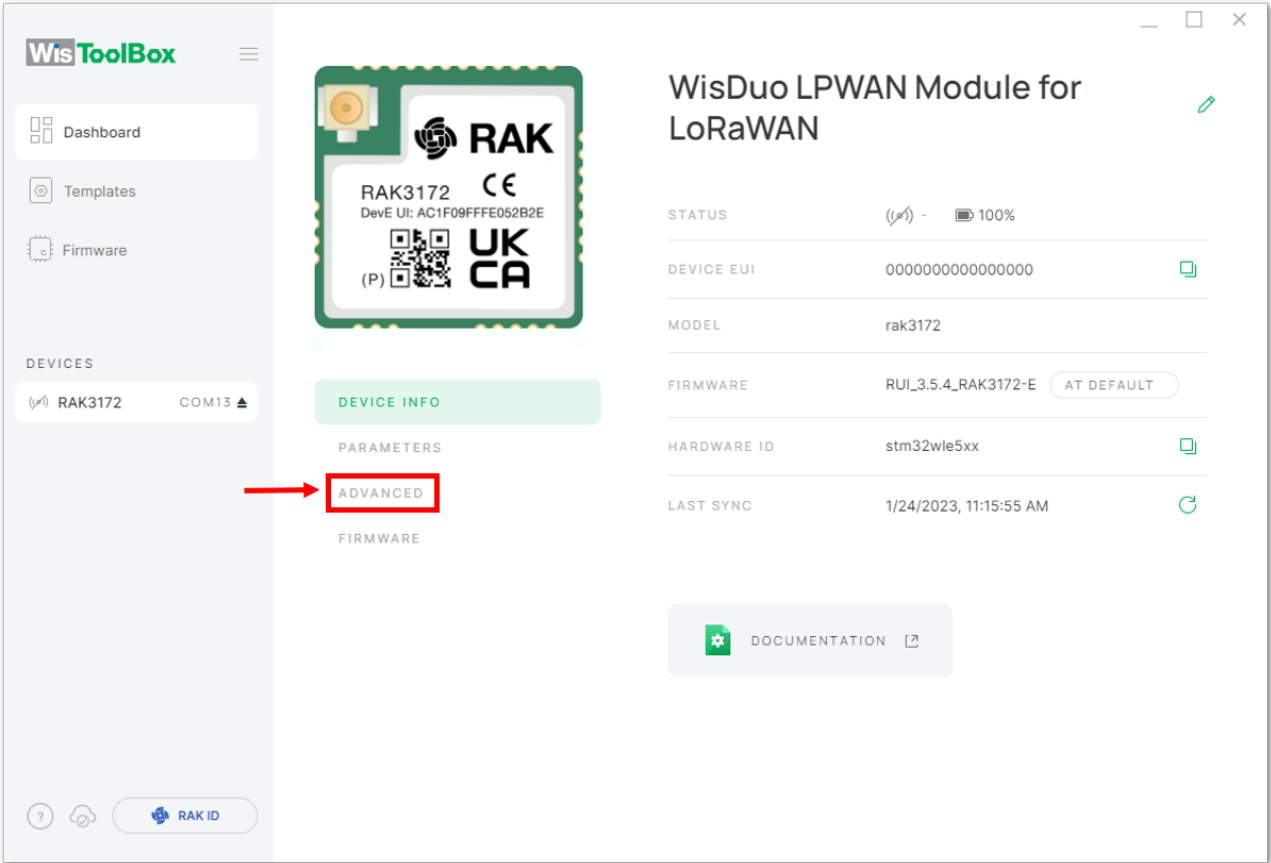


Figure 76: Setting up your device

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

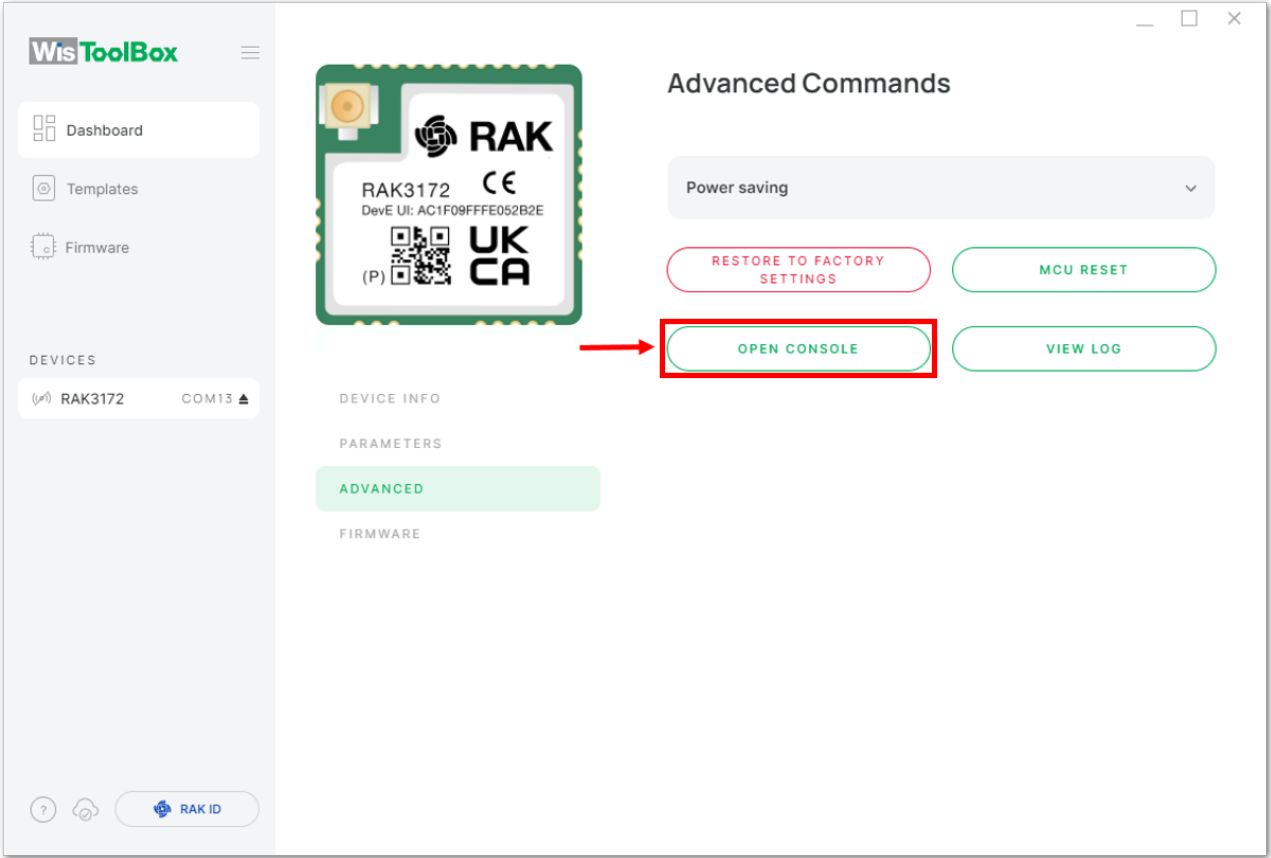


Figure 77: OPEN CONSOLE

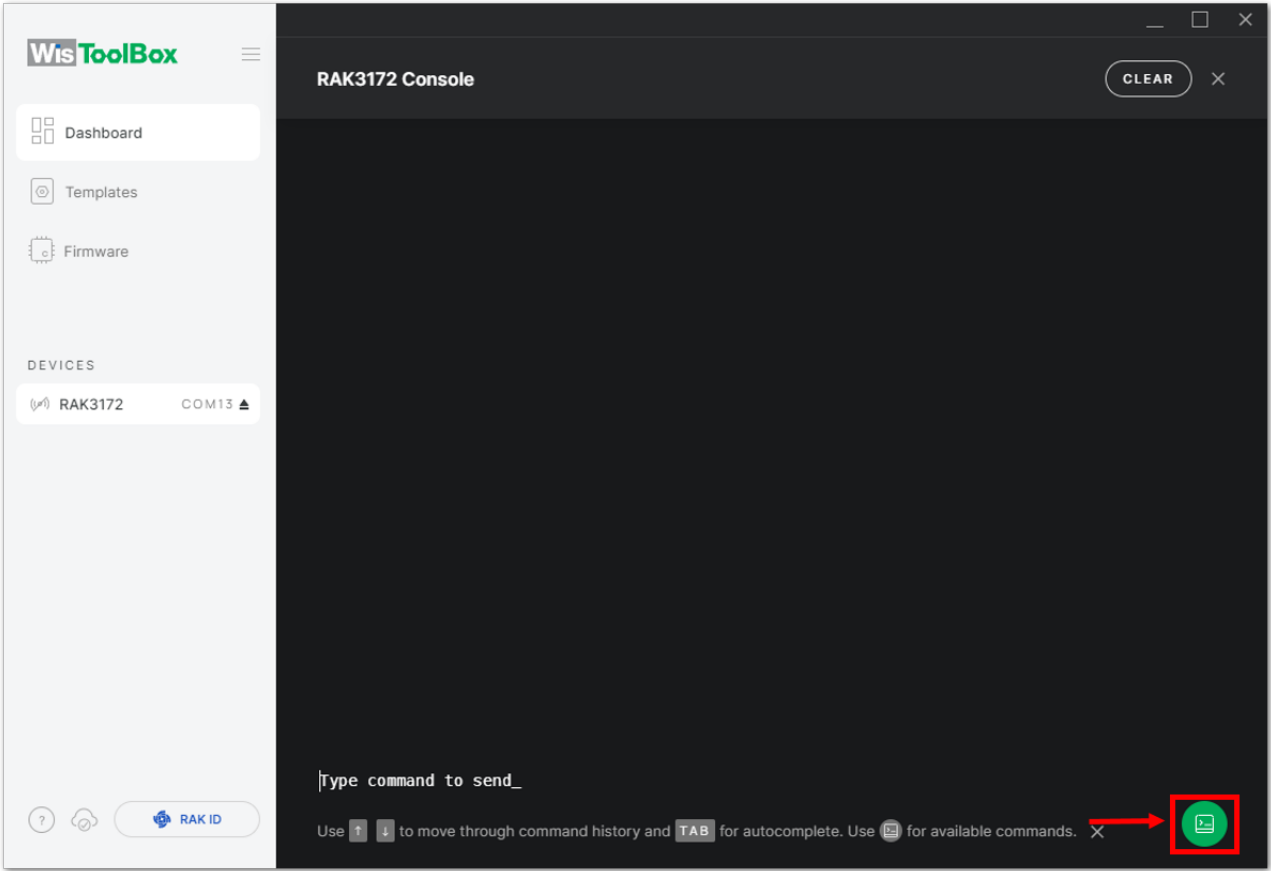


Figure 78: Opening the Console terminal of WisToolBox

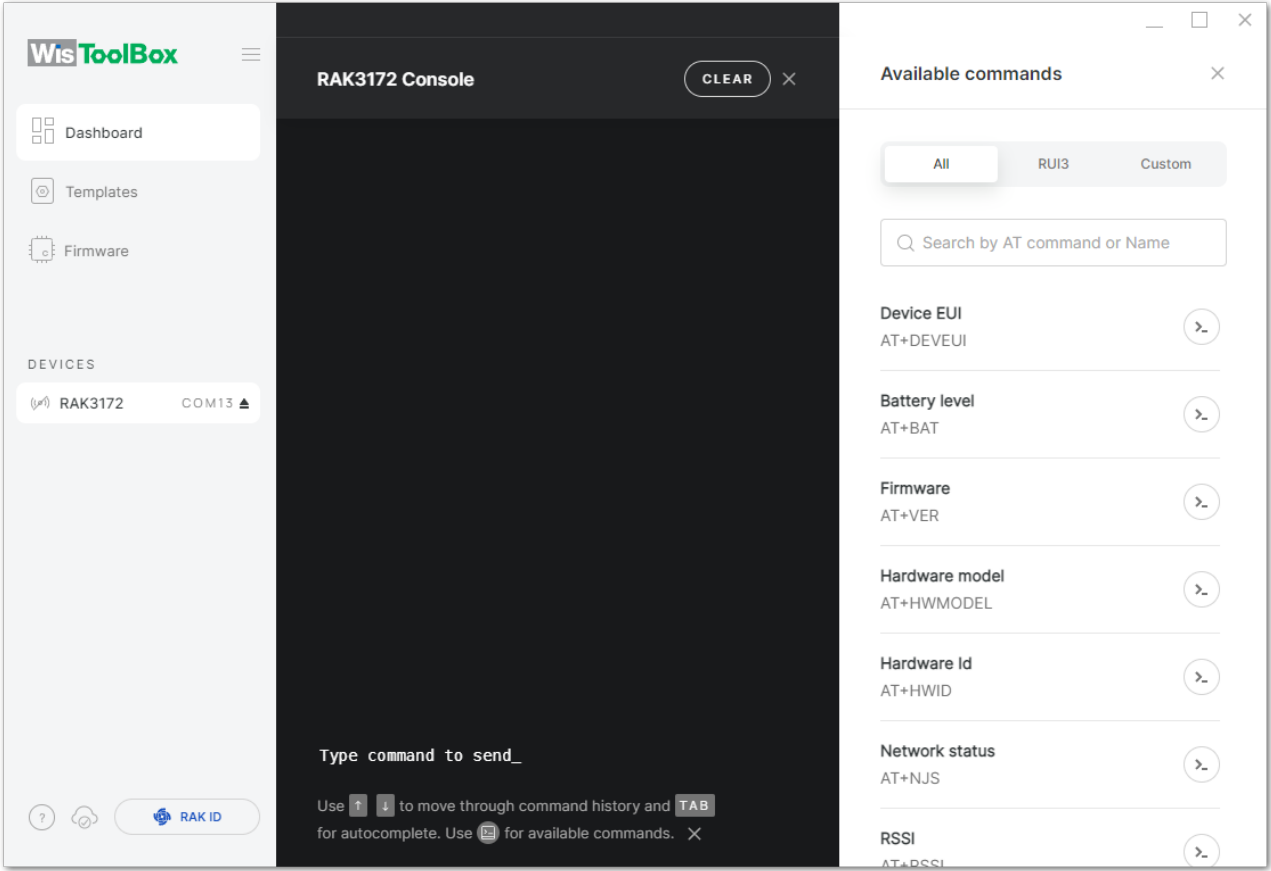


Figure 79: Opening the Console terminal of WisToolBox

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.

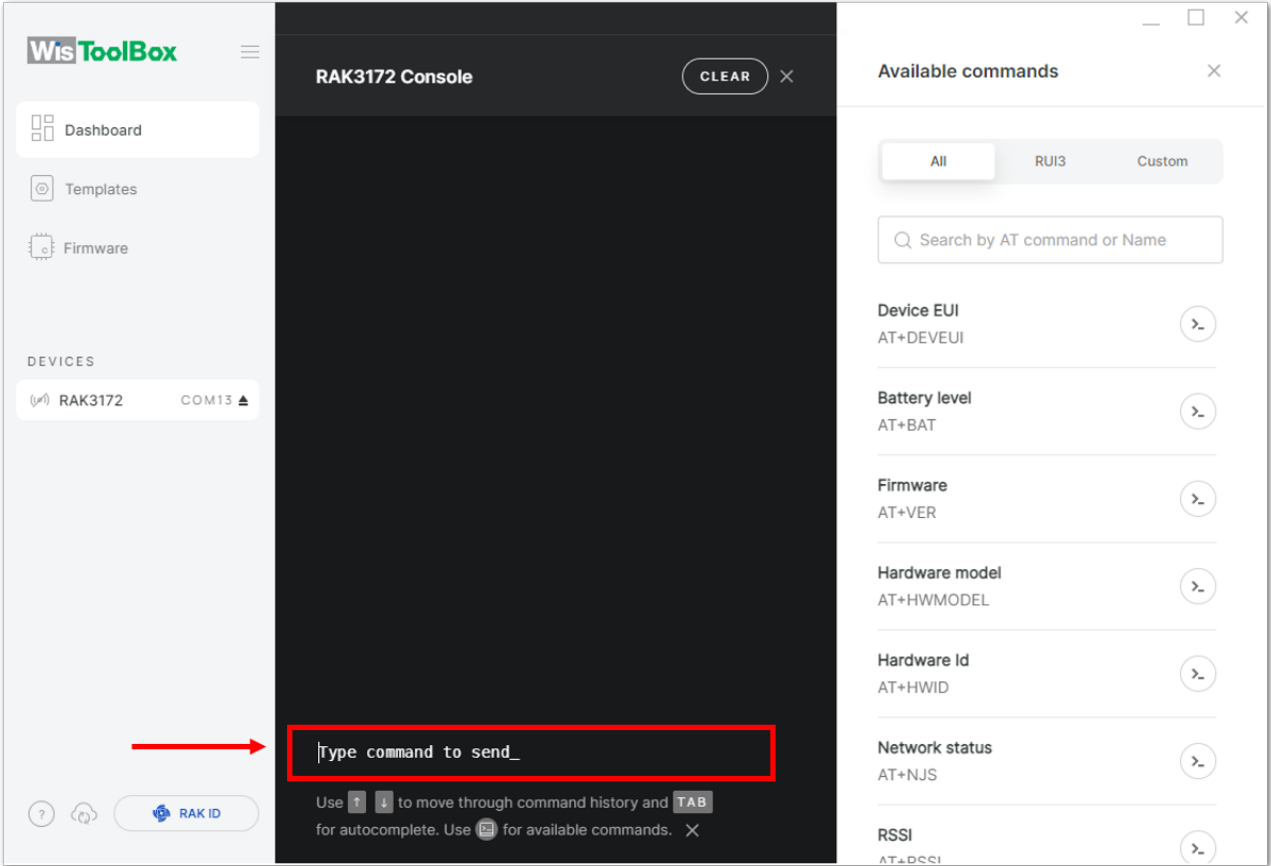


Figure 80: Setting up your Console

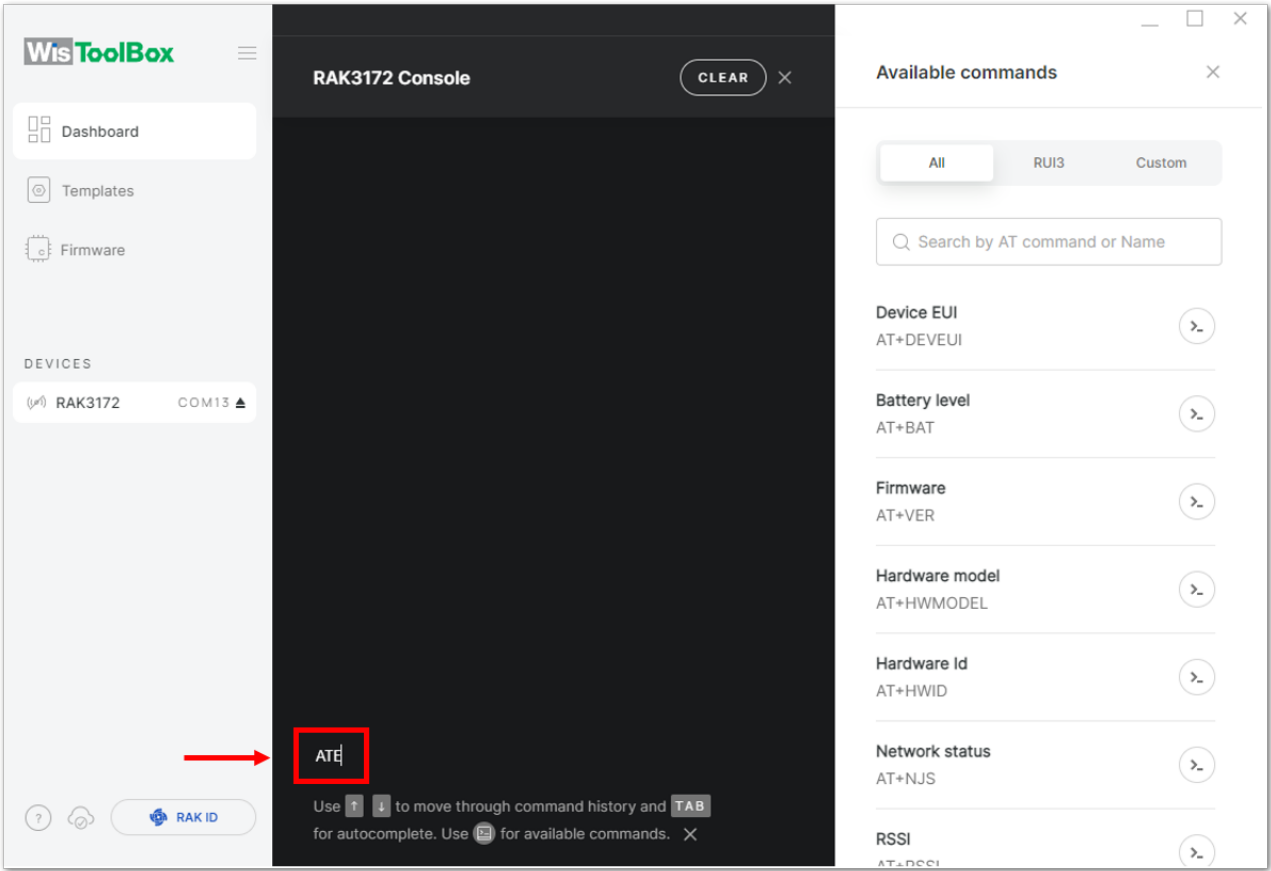


Figure 81: Setting up your Console

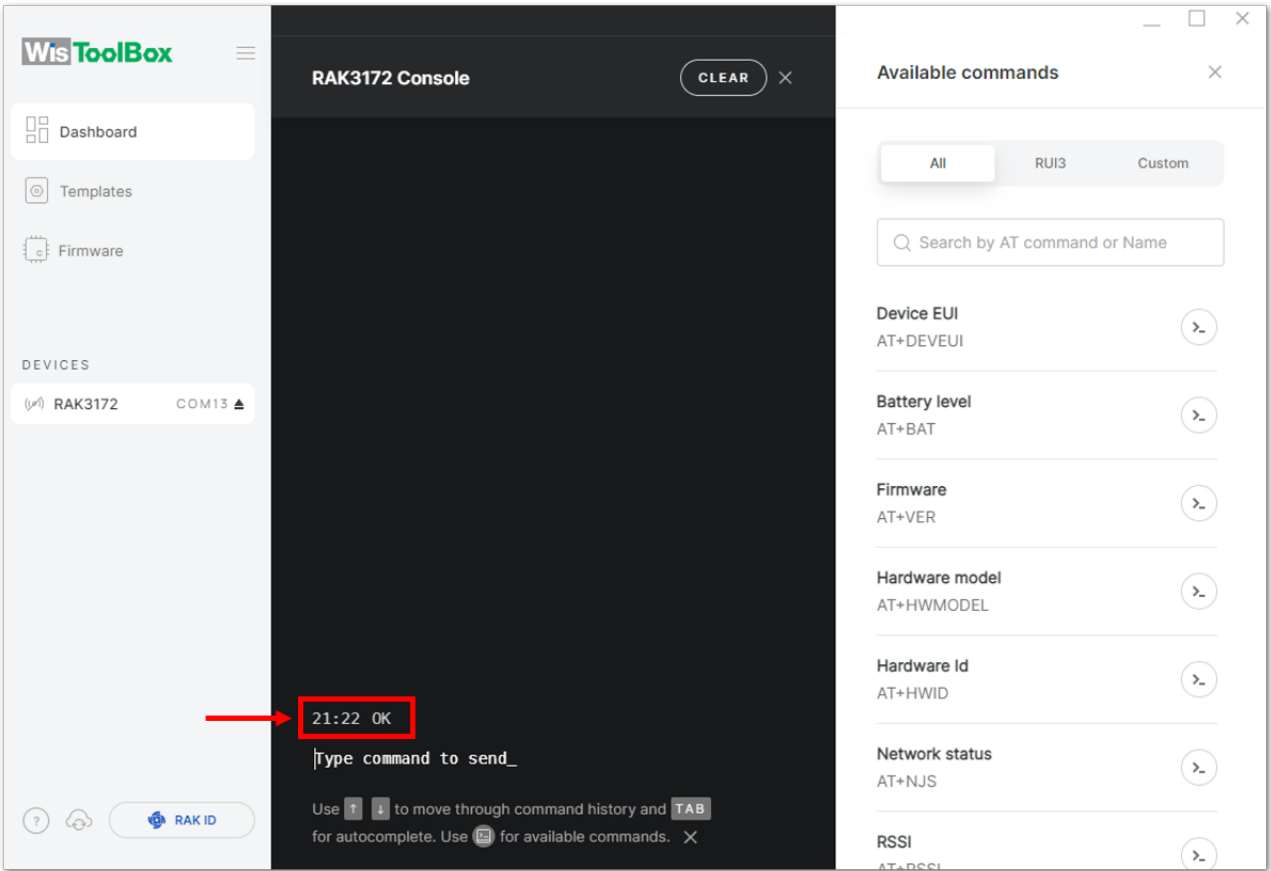


Figure 82: Setting up your Console

8. Then configure the LoRaWAN join mode to **OTAA**. You can check what parameter you will input by typing **AT+NJM?** then **Enter** into the console terminal. For **OTAA**, you should input **AT+NJM=1** then press **Enter** as shown in **Figure 85**.

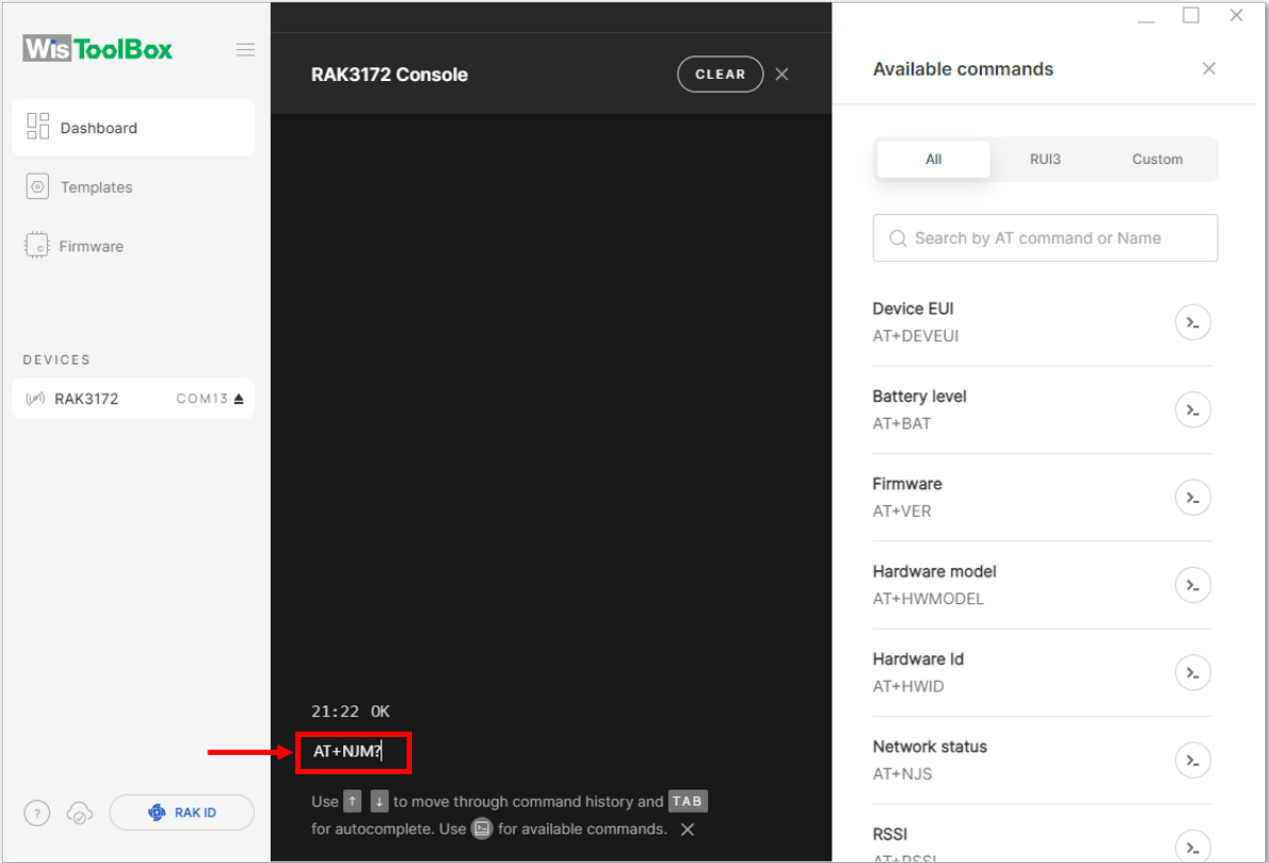


Figure 83: Setting up your Console

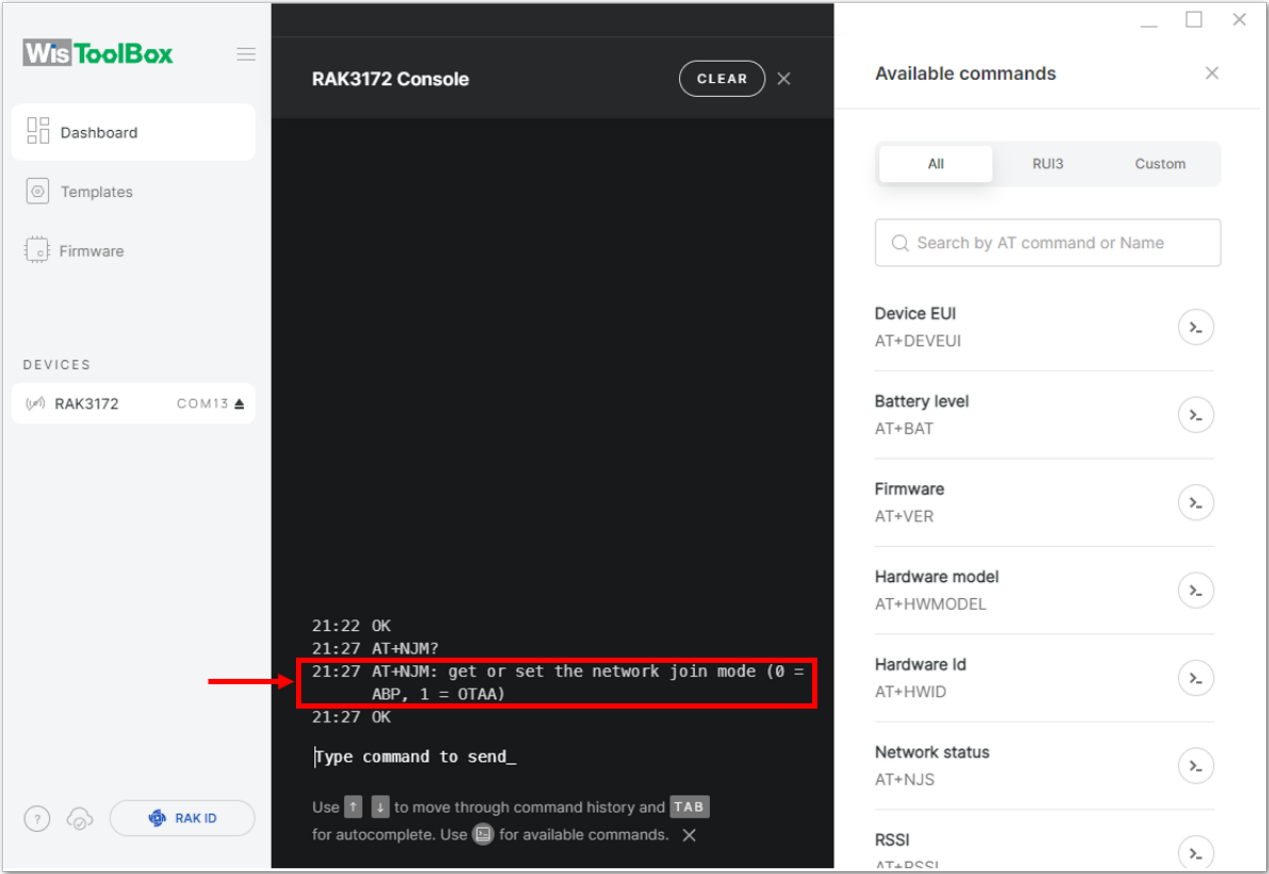


Figure 84: Setting up your Console

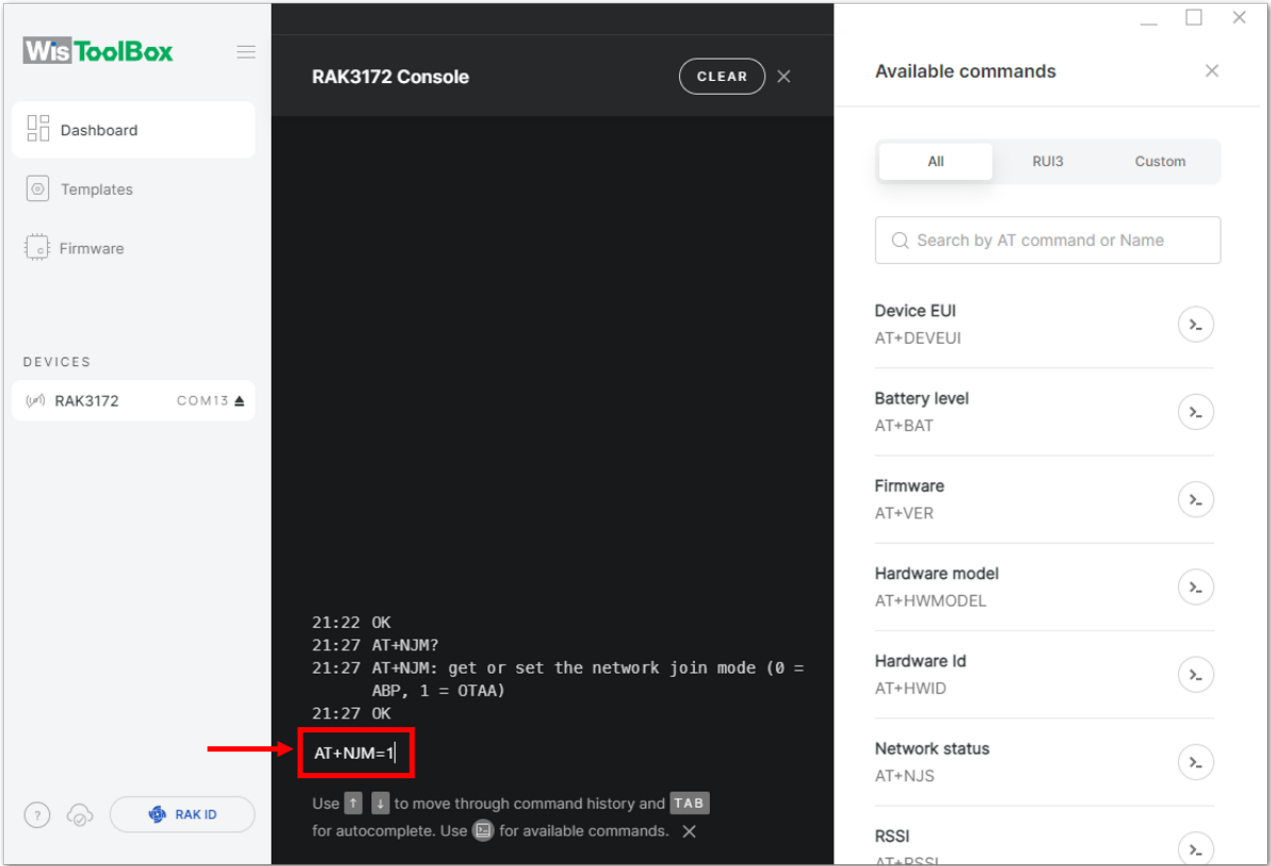


Figure 85: Setting up your Console

9. Once done, set-up your LoRaWAN region to EU868. You can check what parameter you will input by typing **AT+BAND?** 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 band, 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

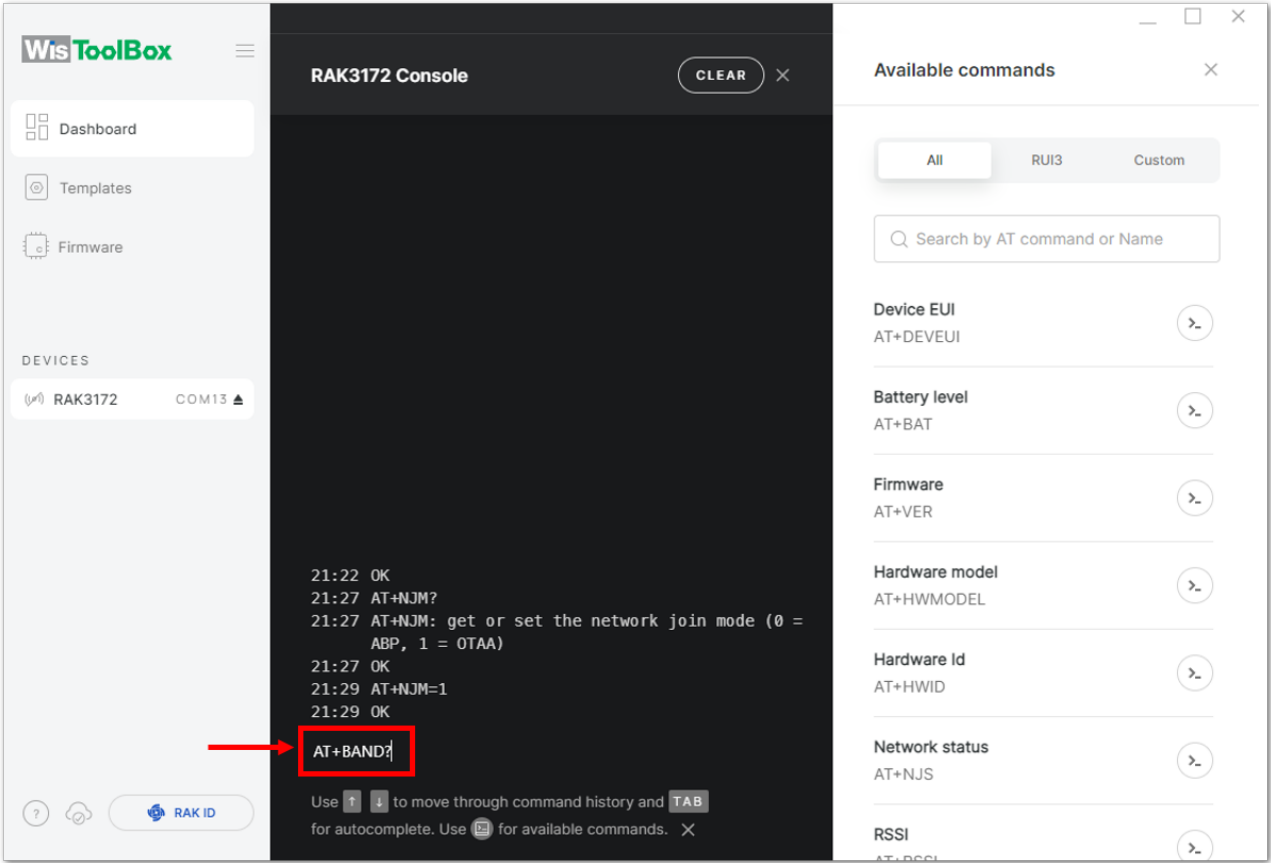


Figure 86: Setting up your Console

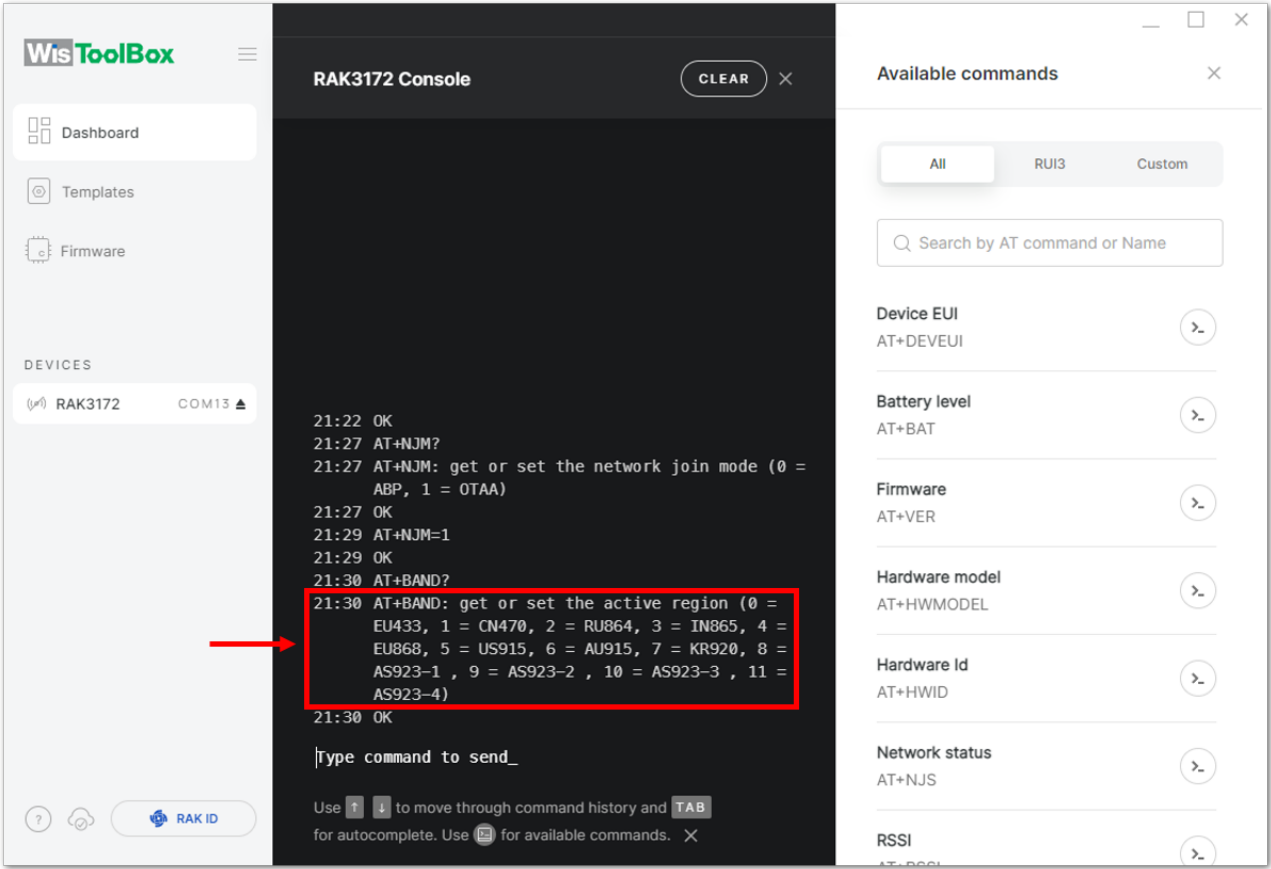


Figure 87: Setting up your Console

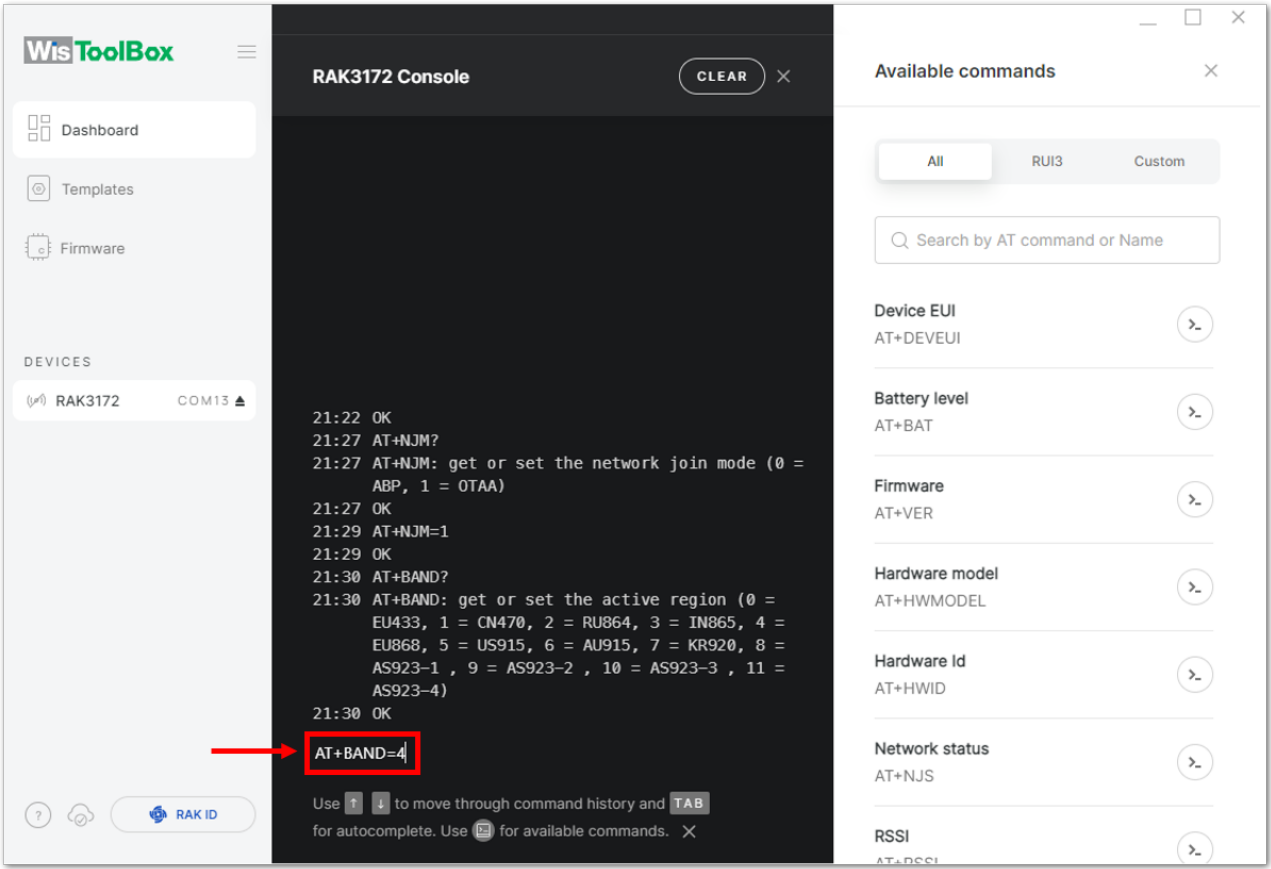


Figure 88: Setting up your Console

10. Then next to this will be updating the OTAA credentials of your device. First to this 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 then press **Enter**.

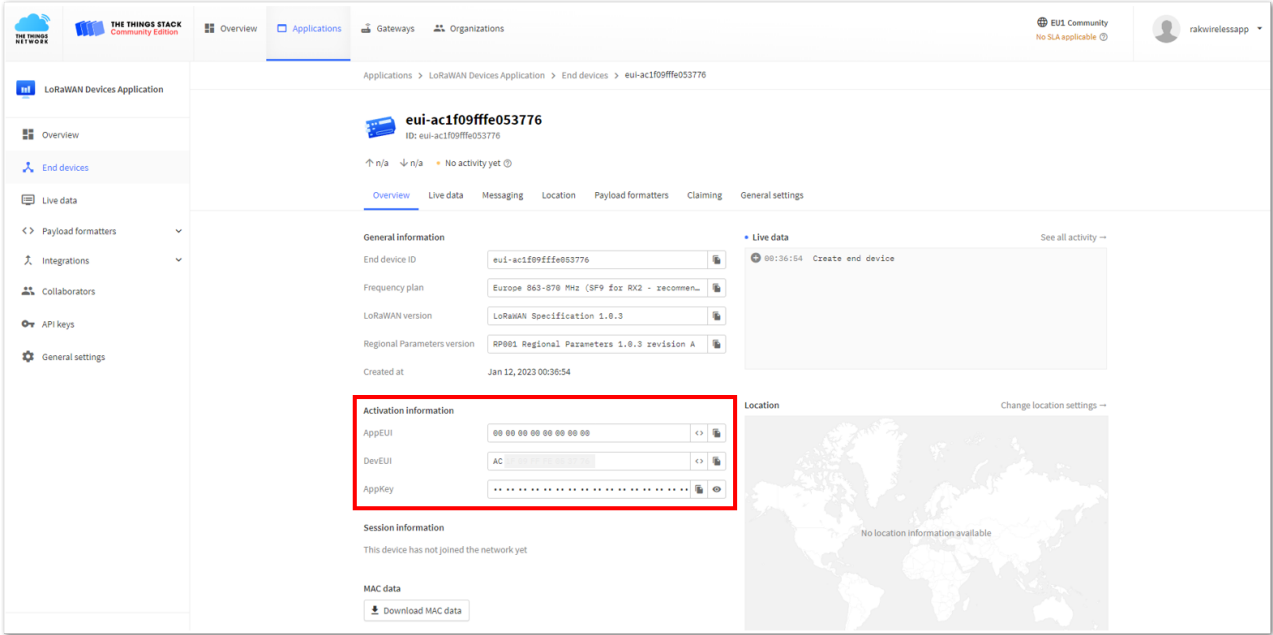


Figure 89: Your created OTAA device from your TTN console

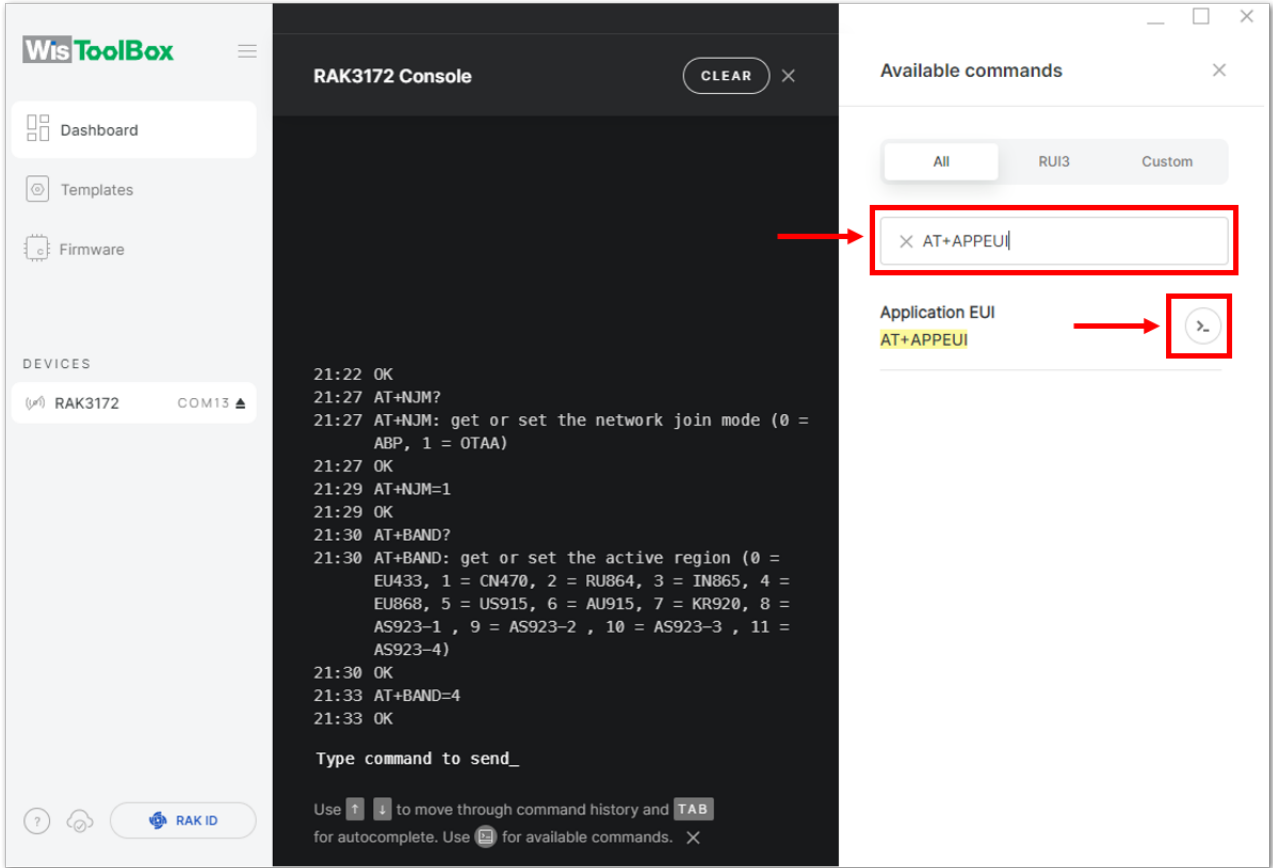


Figure 90: Setting up your Console

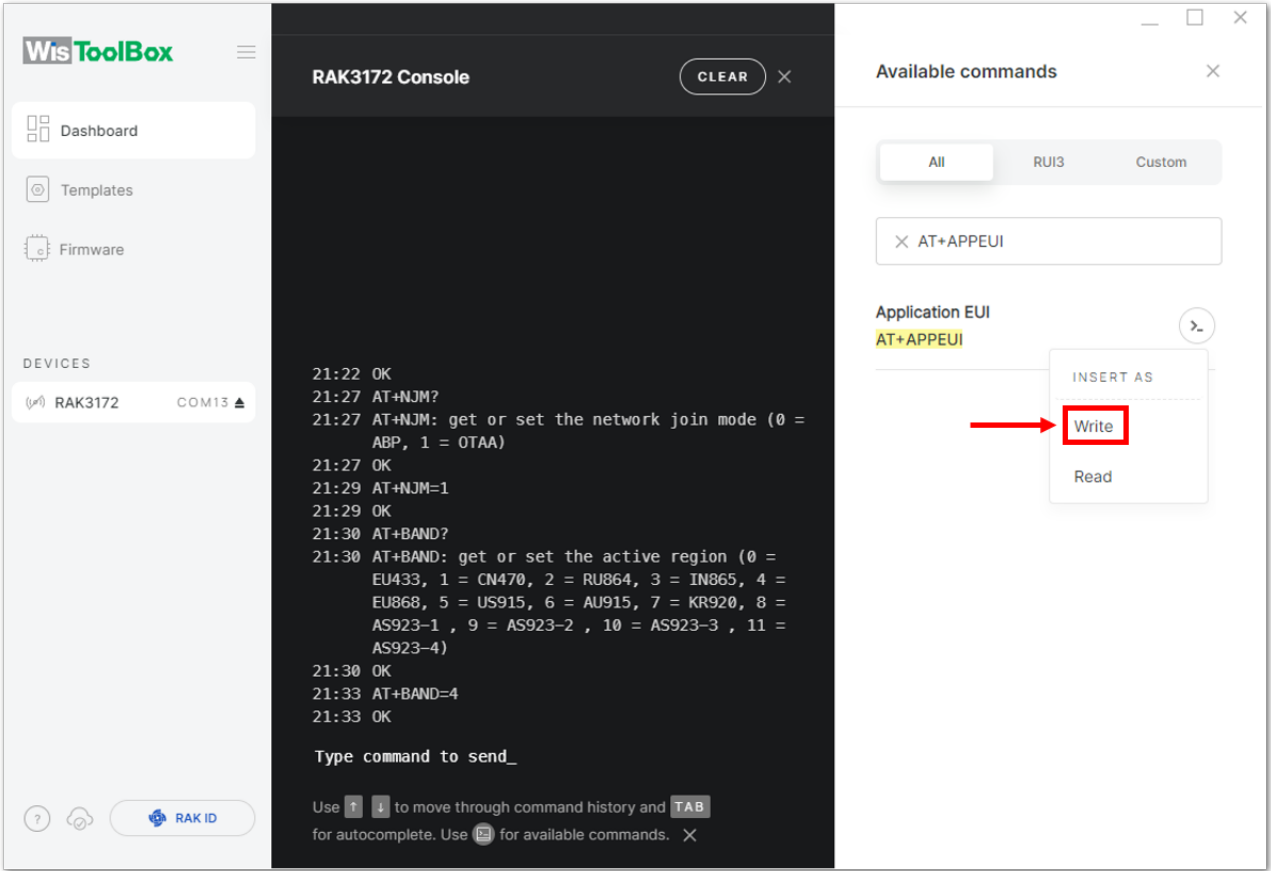


Figure 91: Setting up your Console

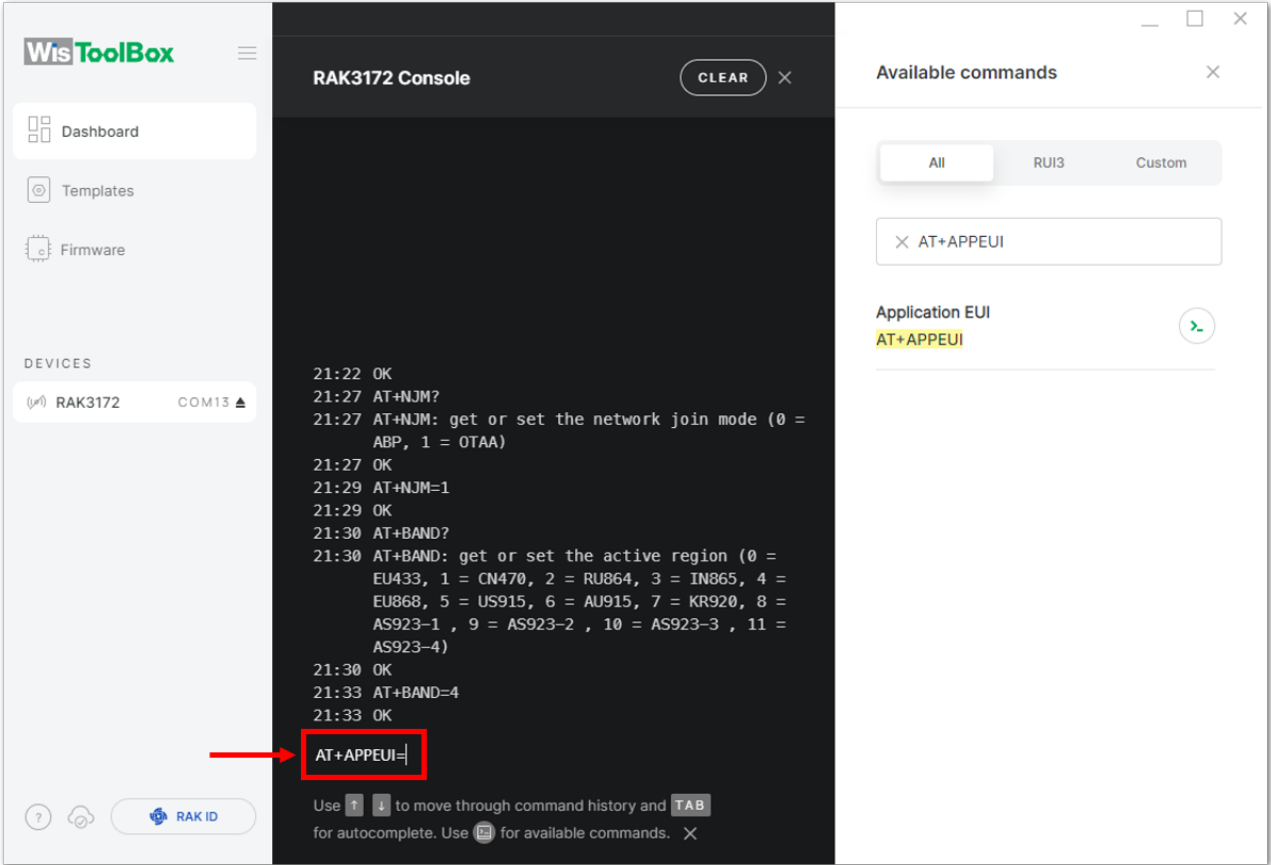


Figure 92: Setting up your Console

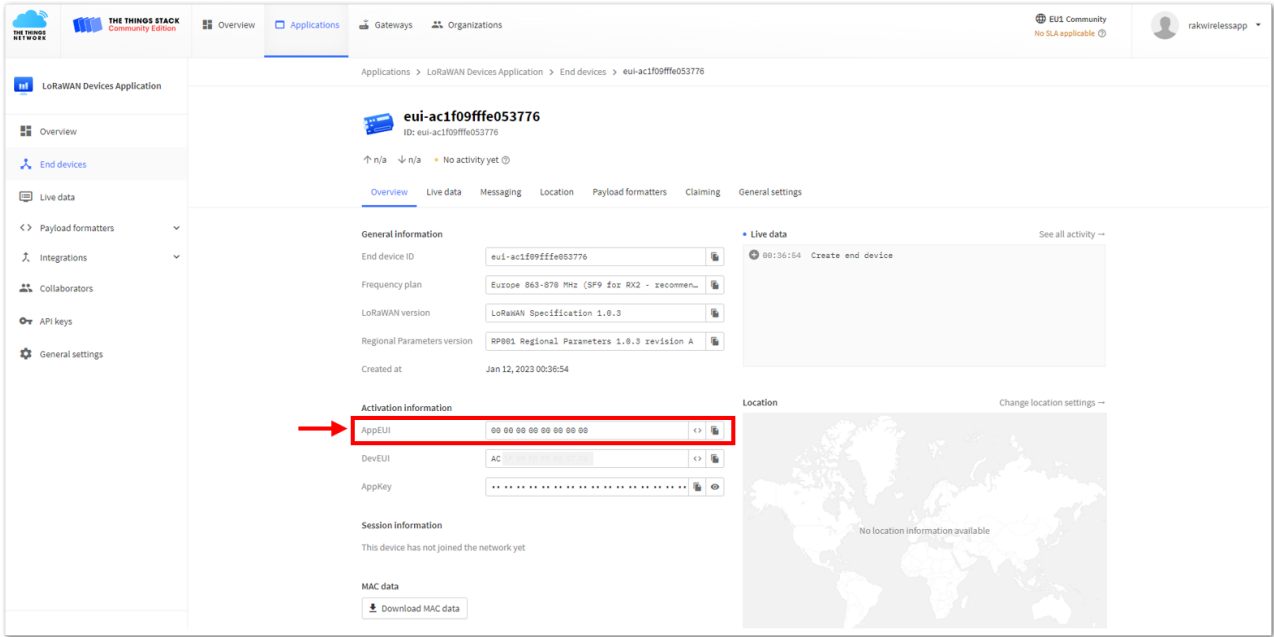


Figure 93: Copying the AppEUI credential from TTN to WisToolBox

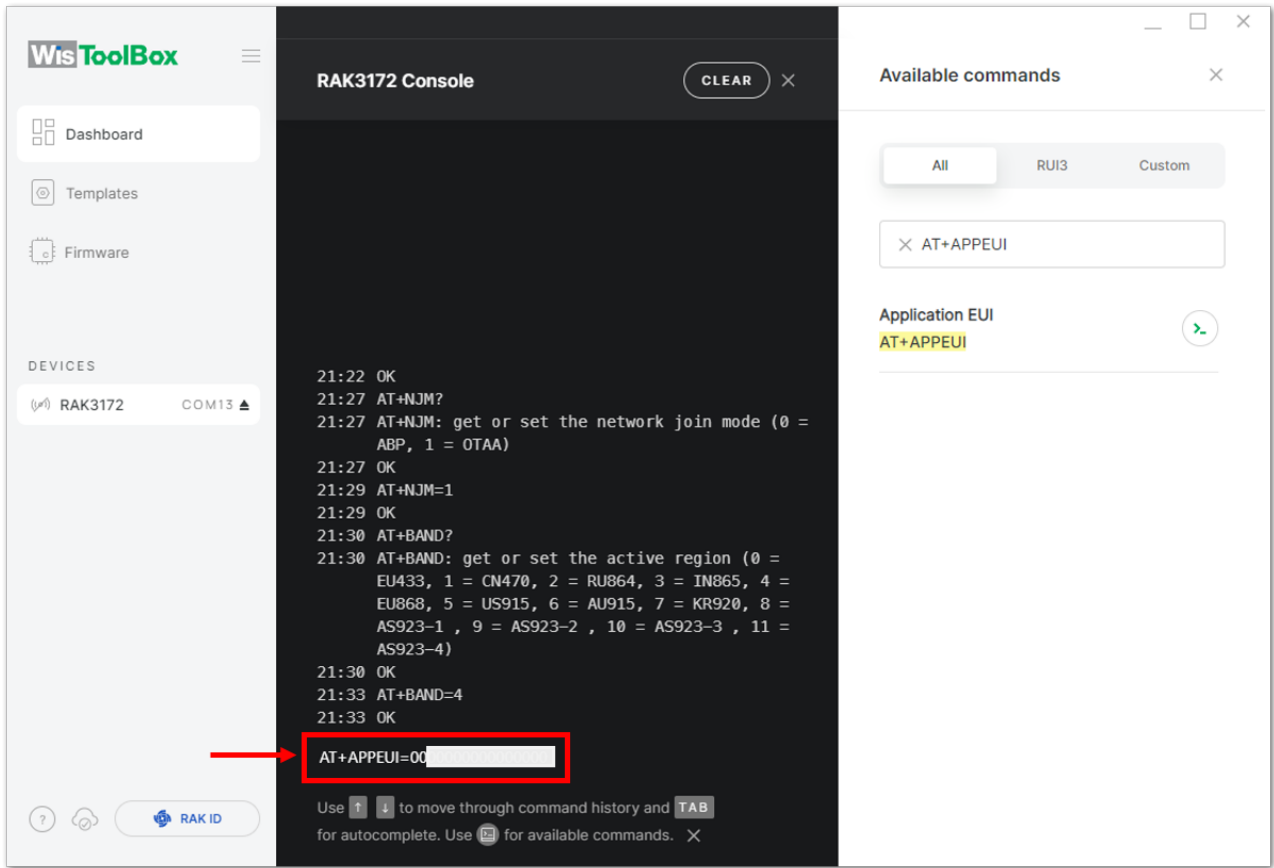


Figure 94: Setting up your Console

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

- **For Application key (AppKey)**

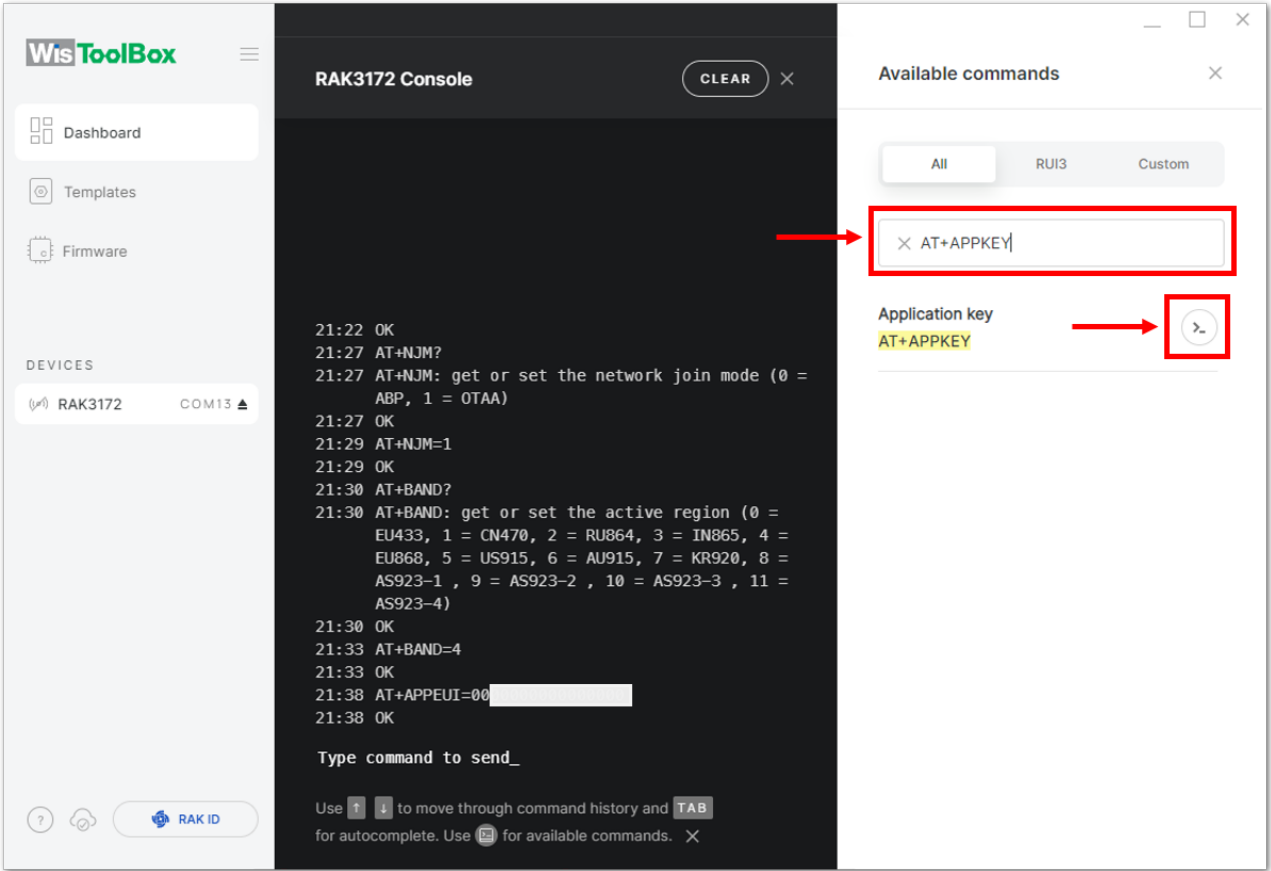


Figure 95: Setting up your Console

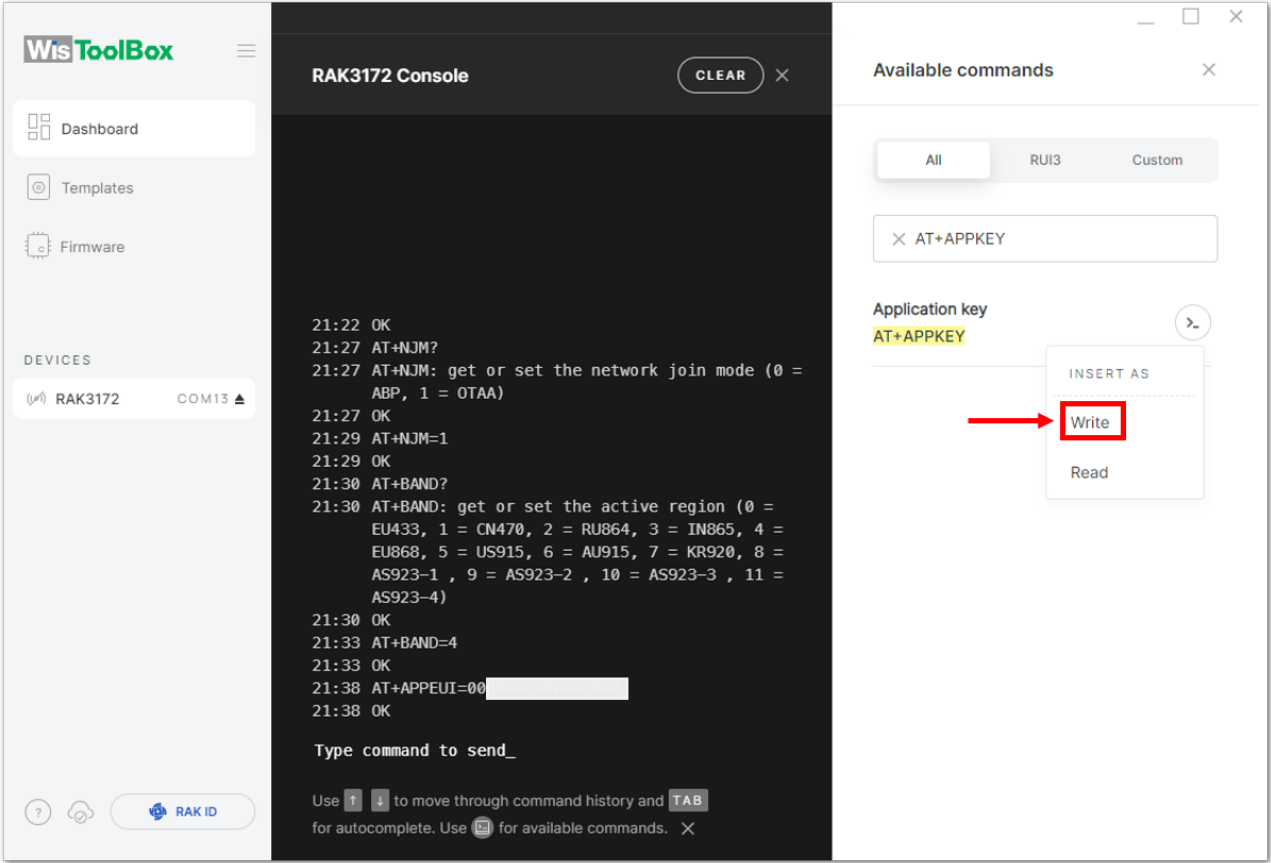


Figure 96: Setting up your Console

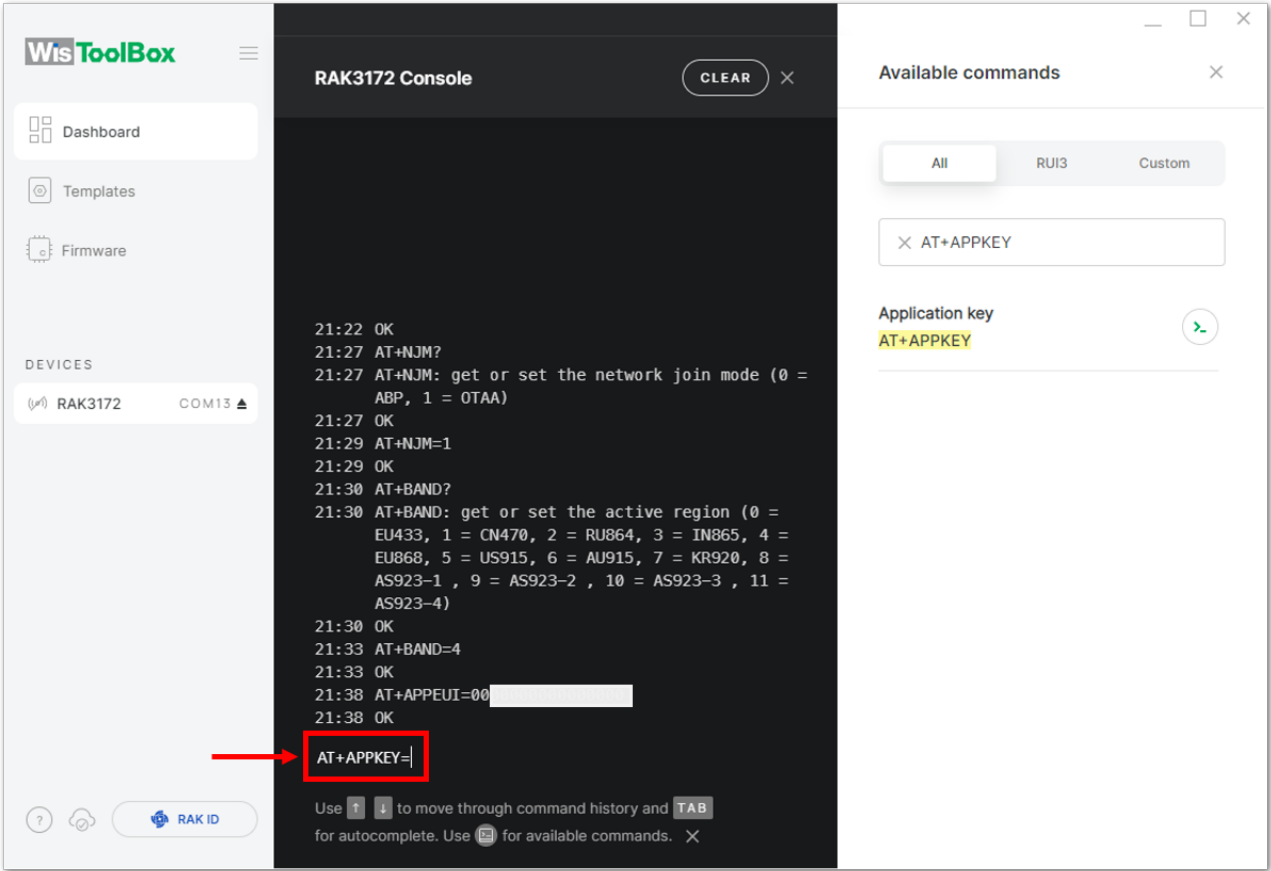


Figure 97: Setting up your Console

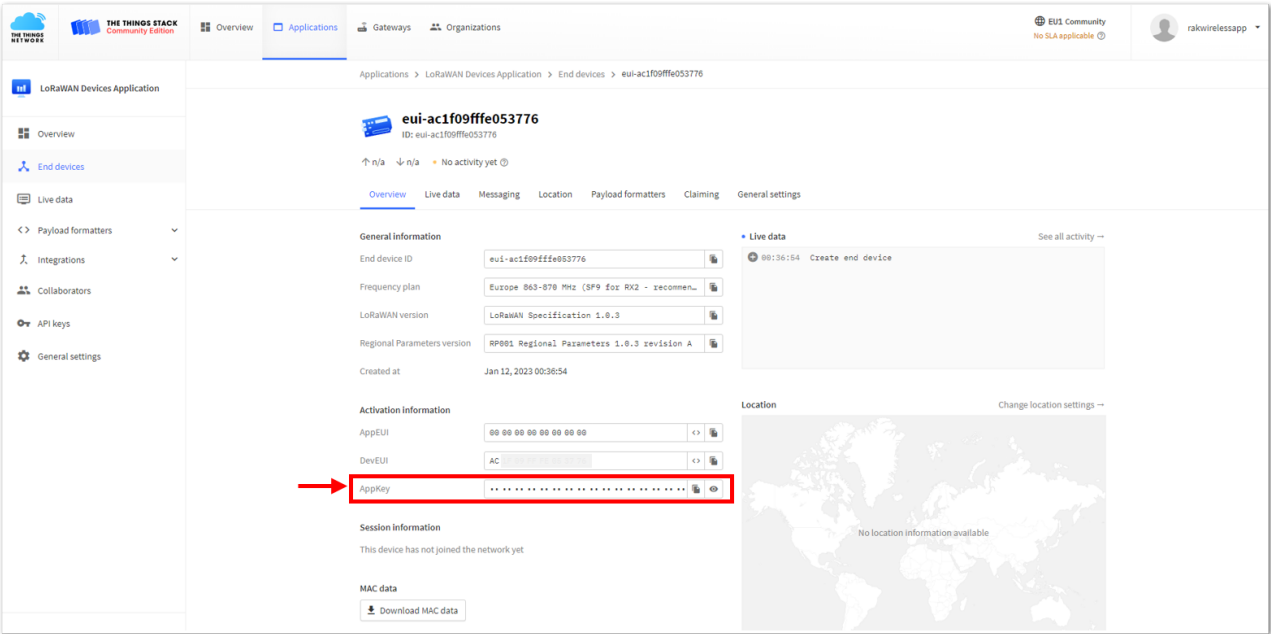


Figure 98: Copying the AppKey credential from TTN to WisToolBox

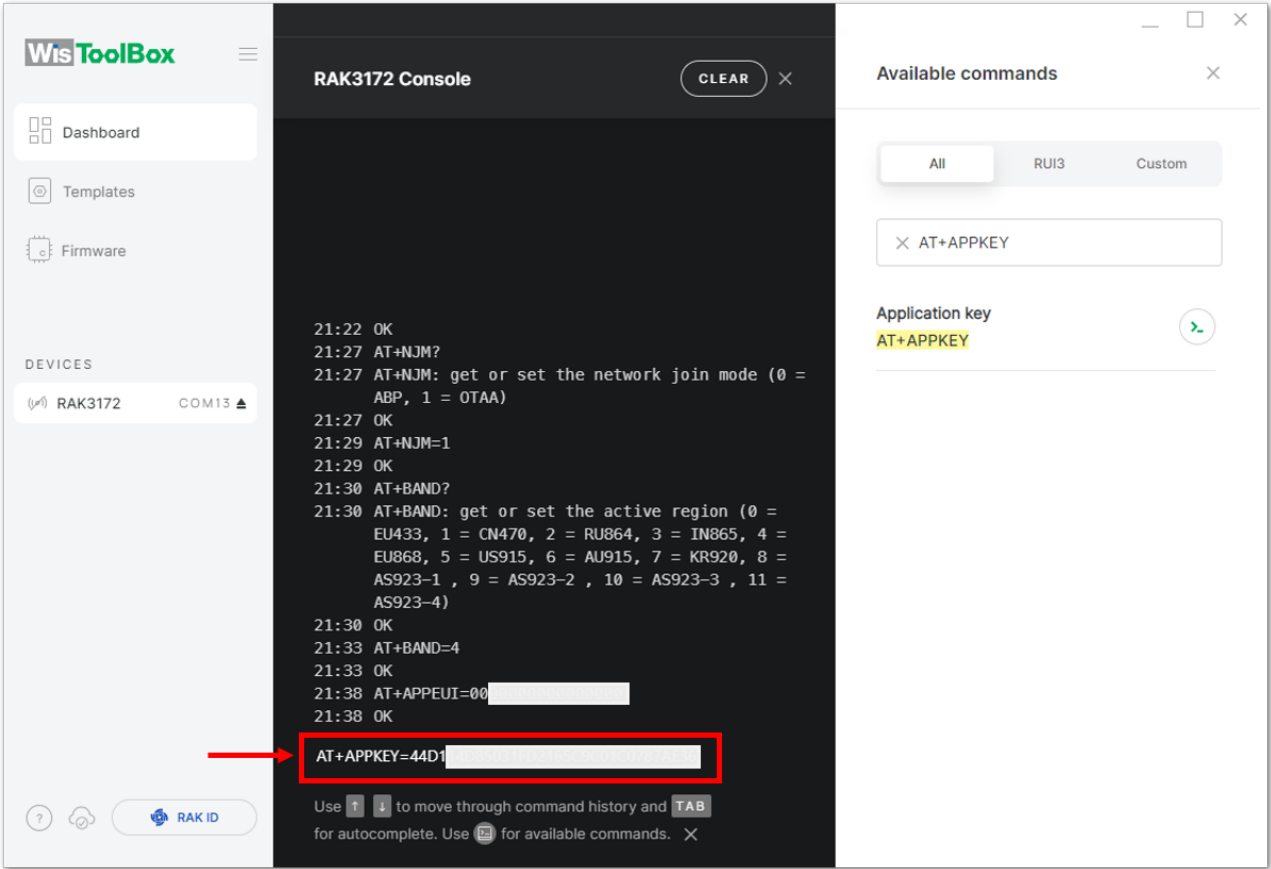


Figure 99: Setting up your Console

- For Device EUI (DevEUI)

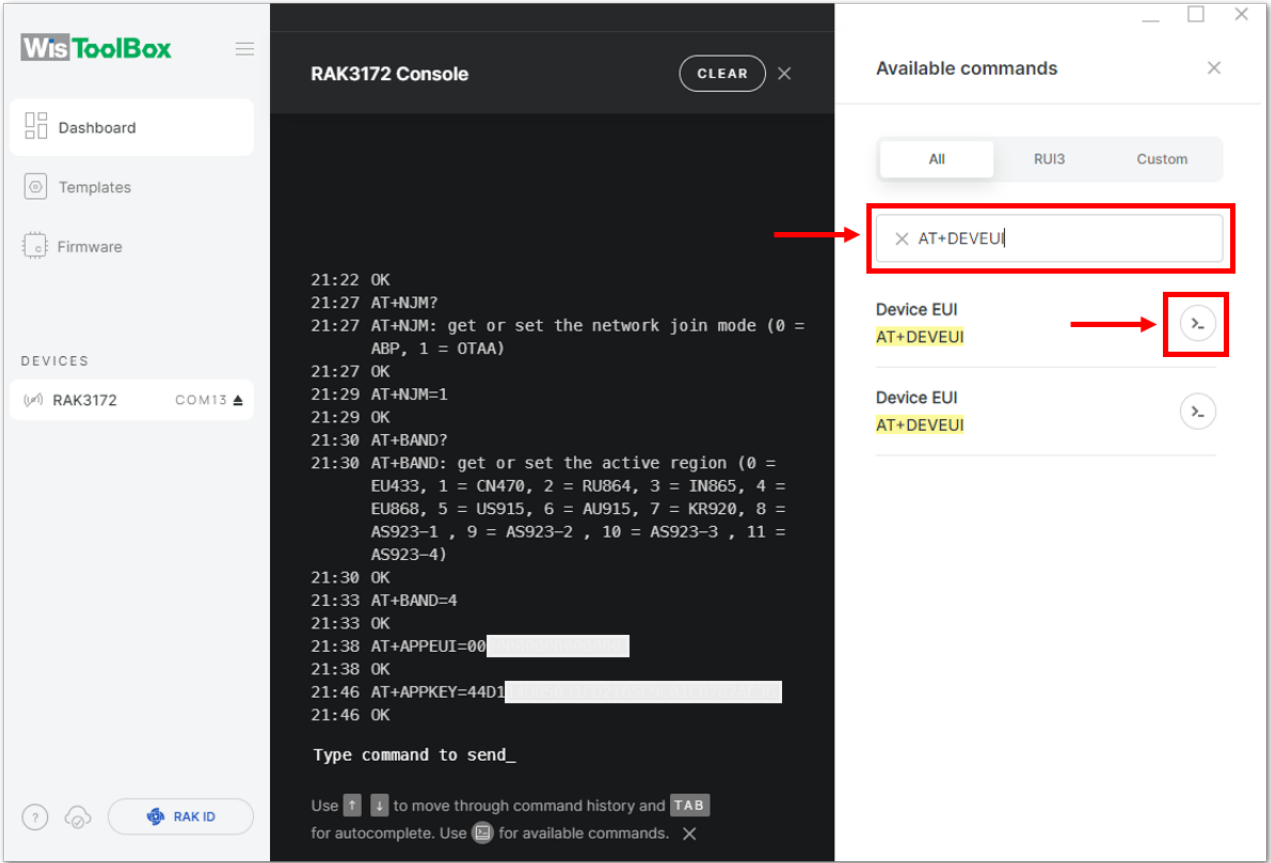


Figure 100: Setting up your Console

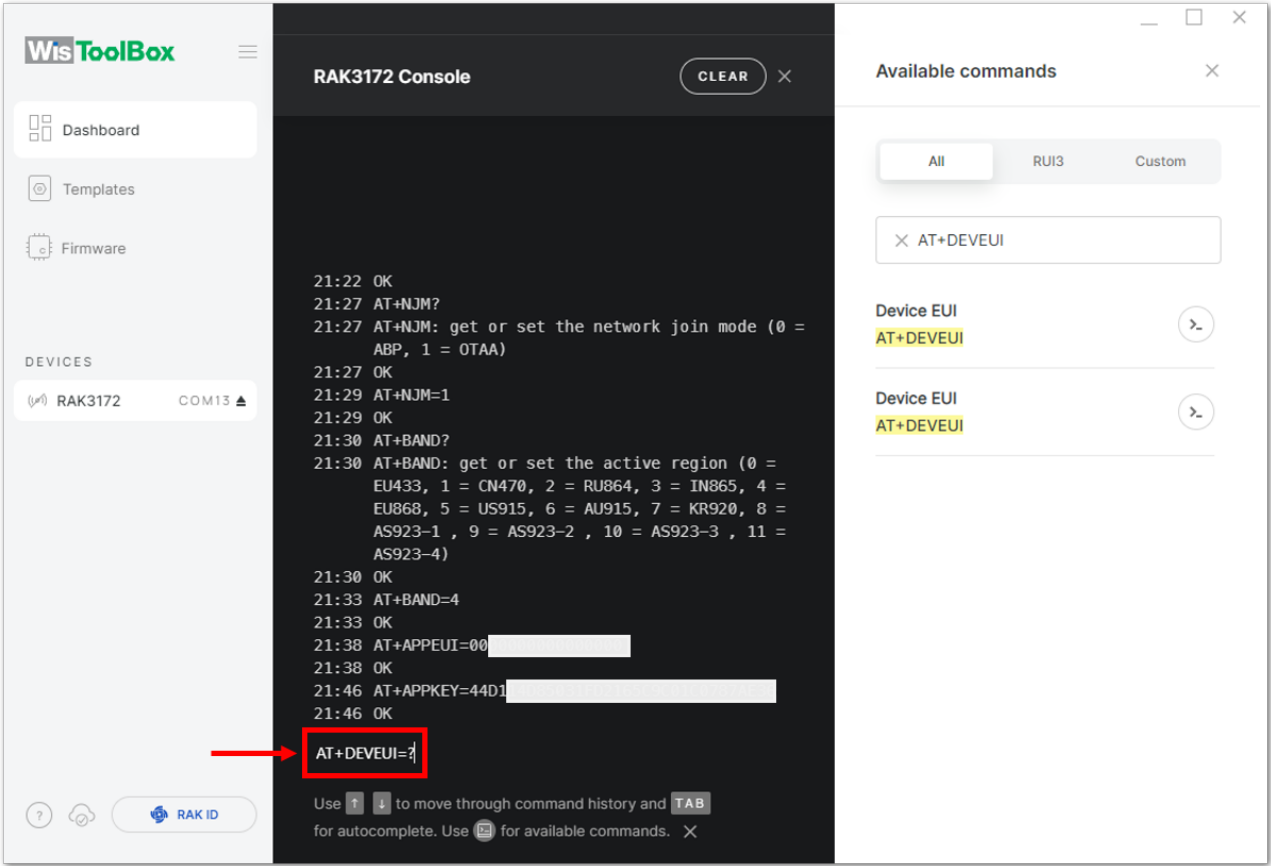


Figure 101: Setting up your Console

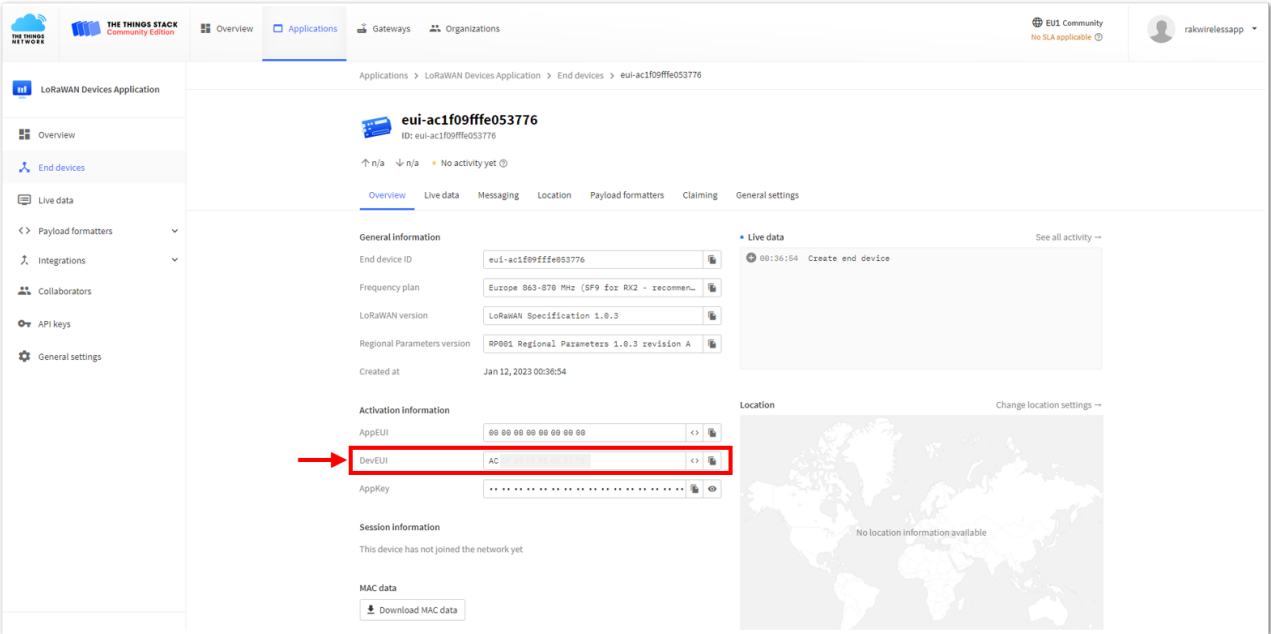


Figure 102: Copying the DevEUI credential from TTN to WisToolBox

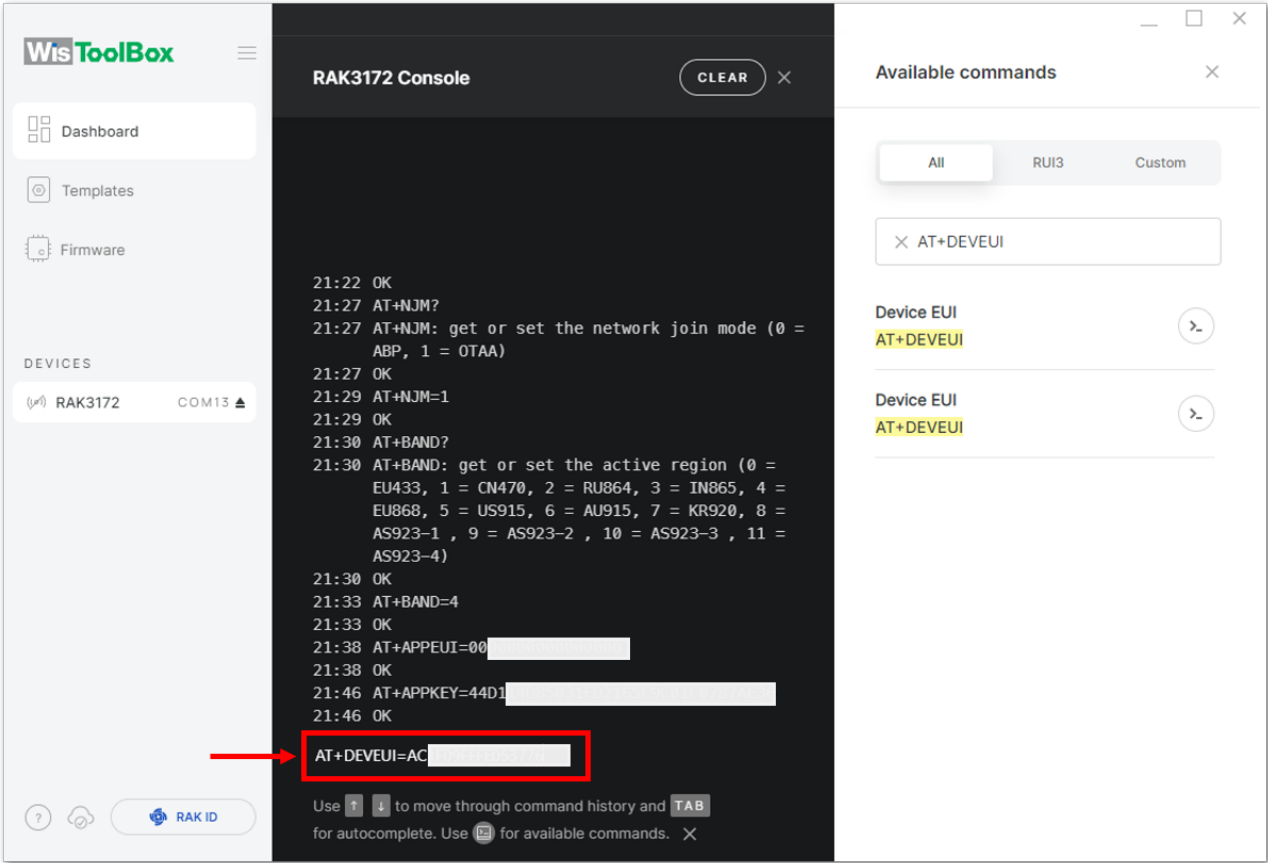


Figure 103: Setting up your Console

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.

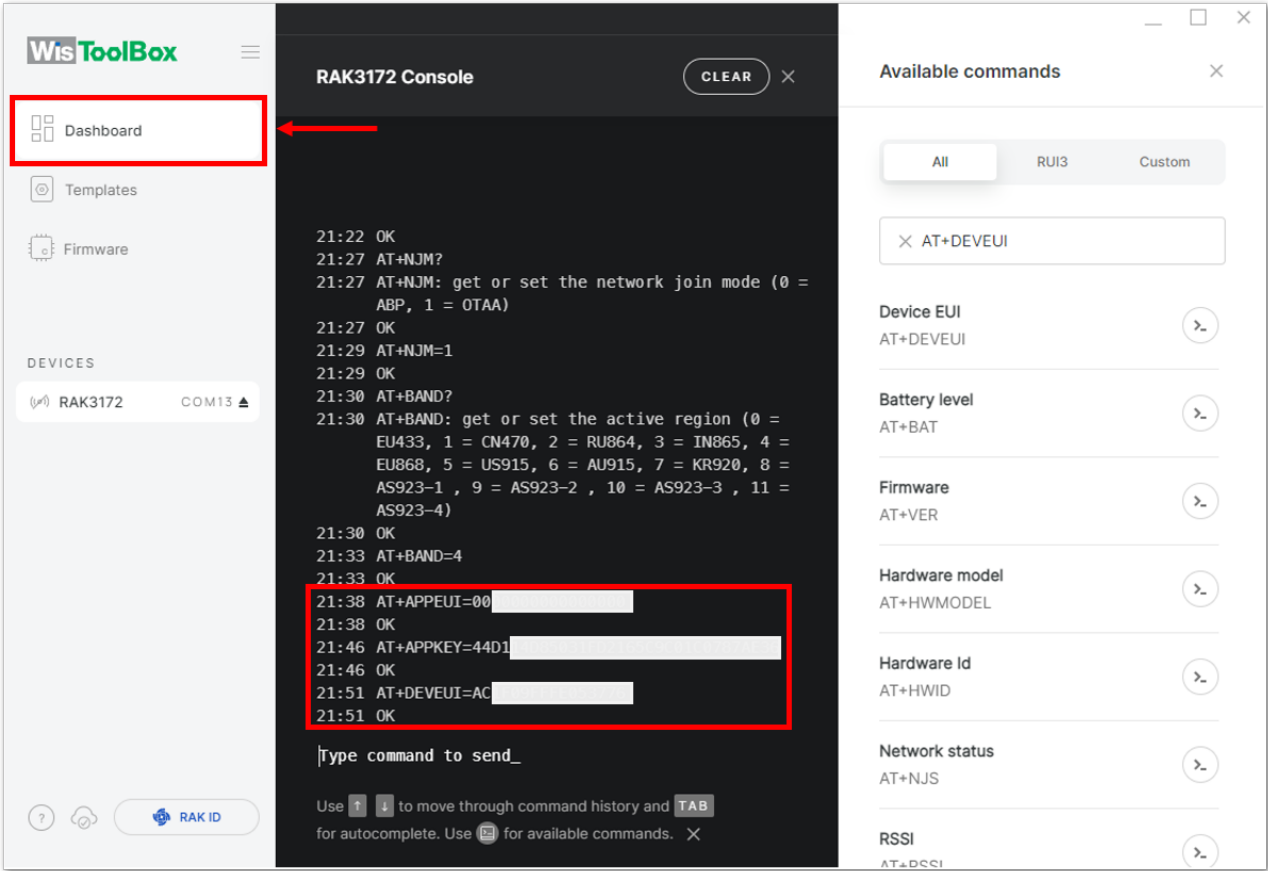


Figure 104: Setting up your Console

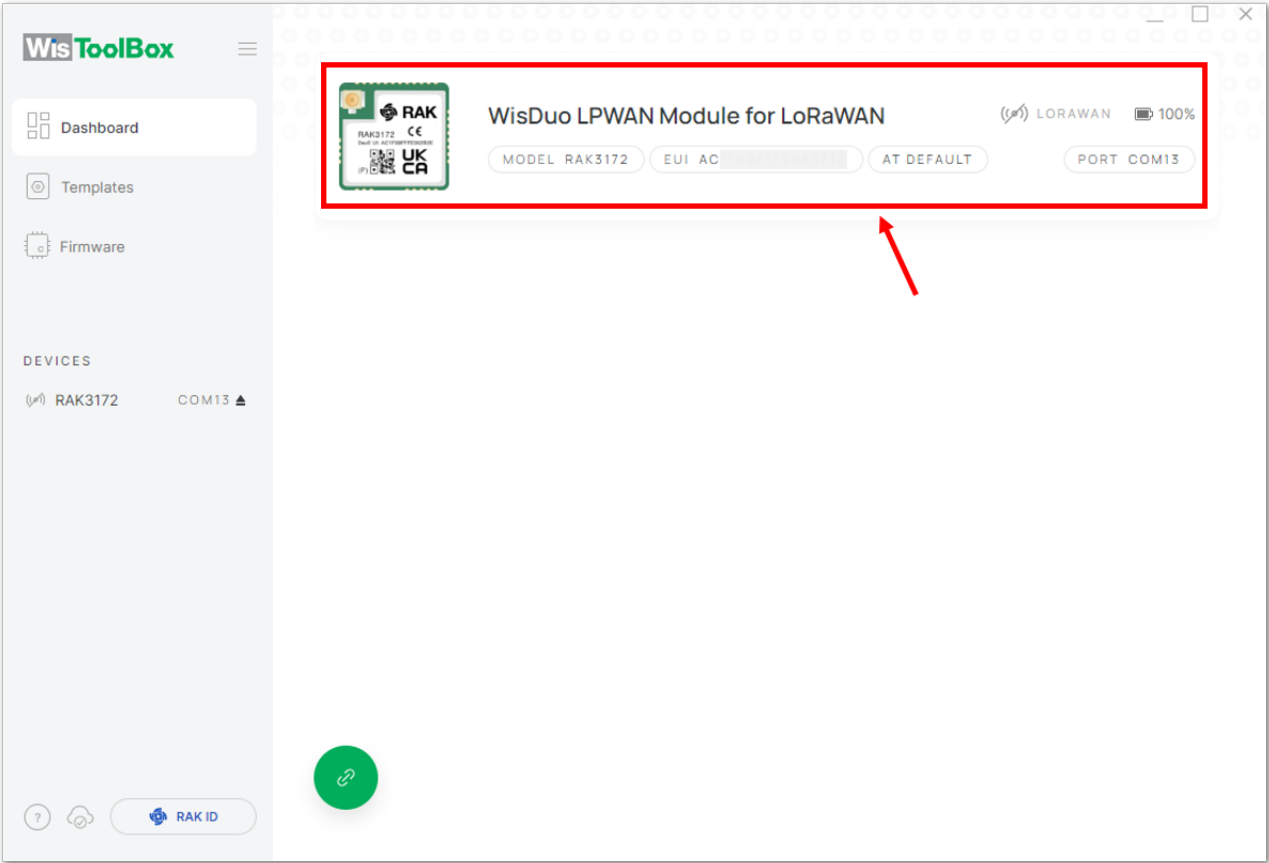


Figure 105: Setting up your Console

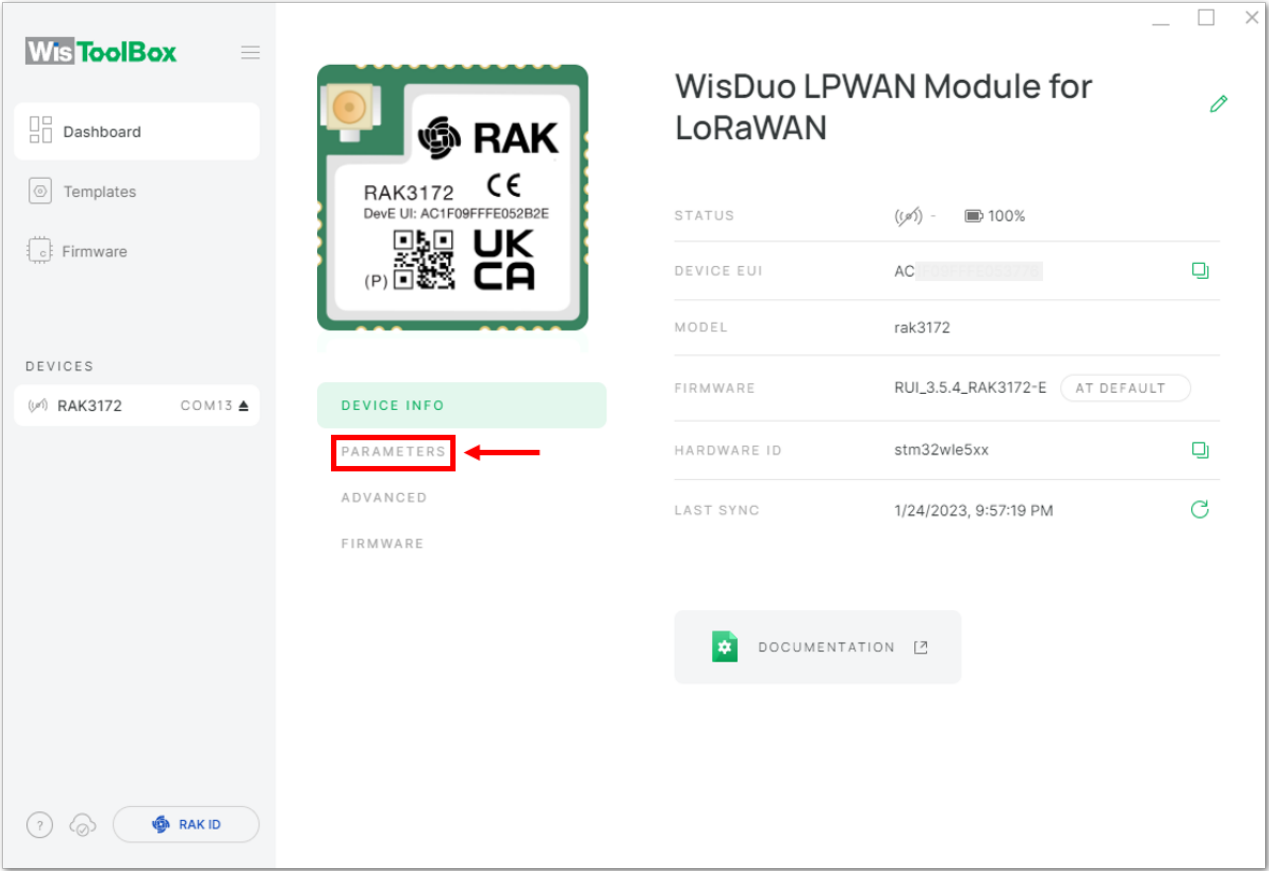


Figure 106: PARAMETERS

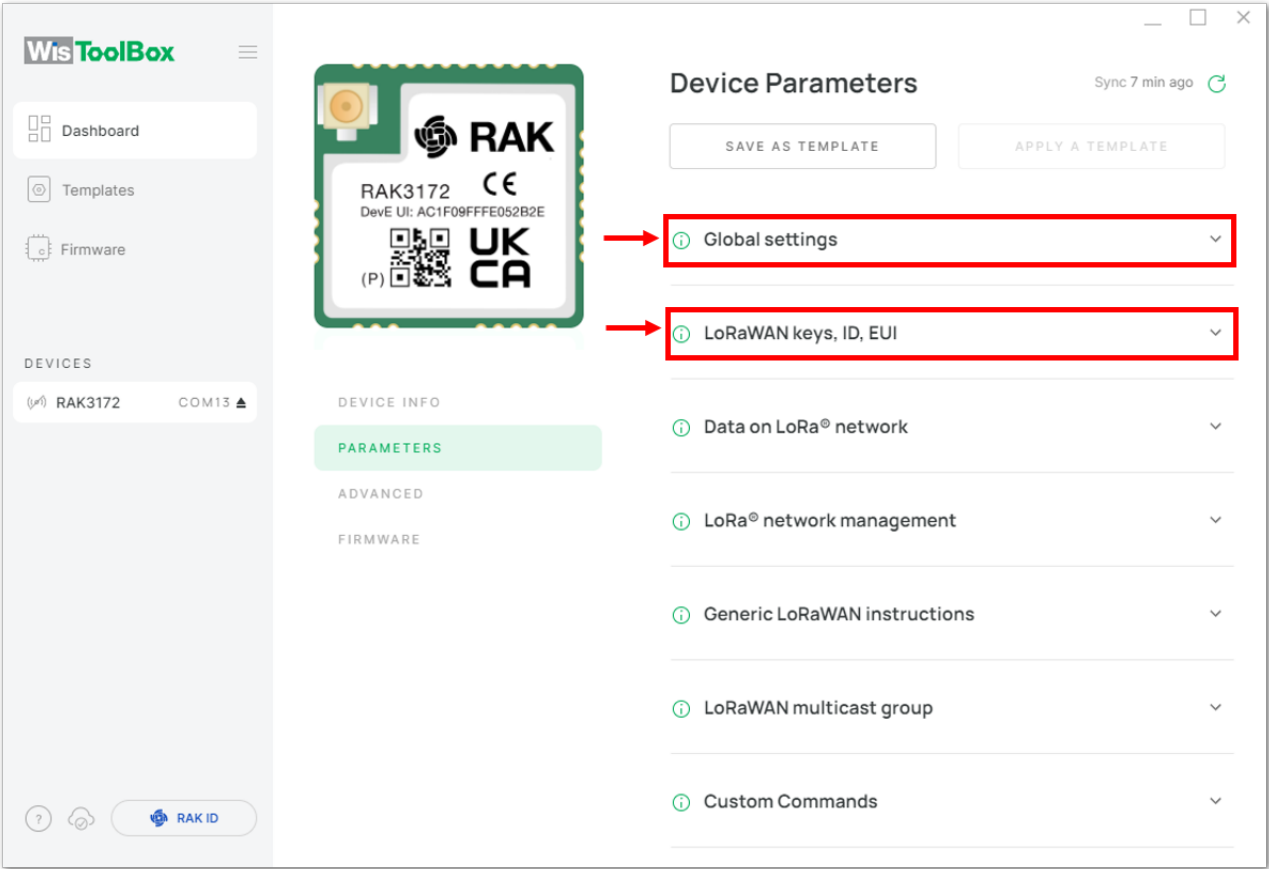


Figure 107: Global settings and LoRaWAN keys, ID, EUI

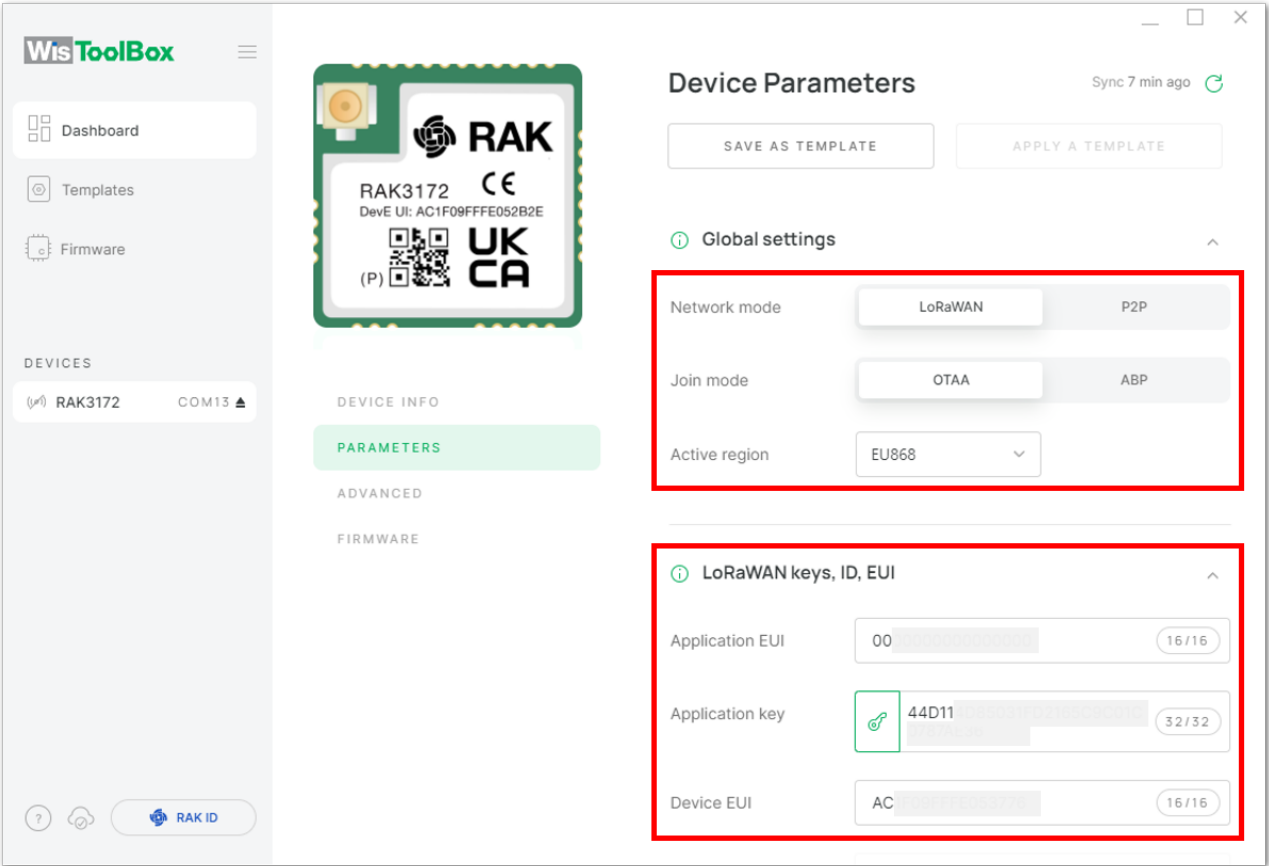


Figure 108: Global settings and LoRaWAN keys, ID, EUI details

- 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 again to WisToolBox console and type **AT+JOIN**. Then edit it to **AT+JOIN=1** then 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.

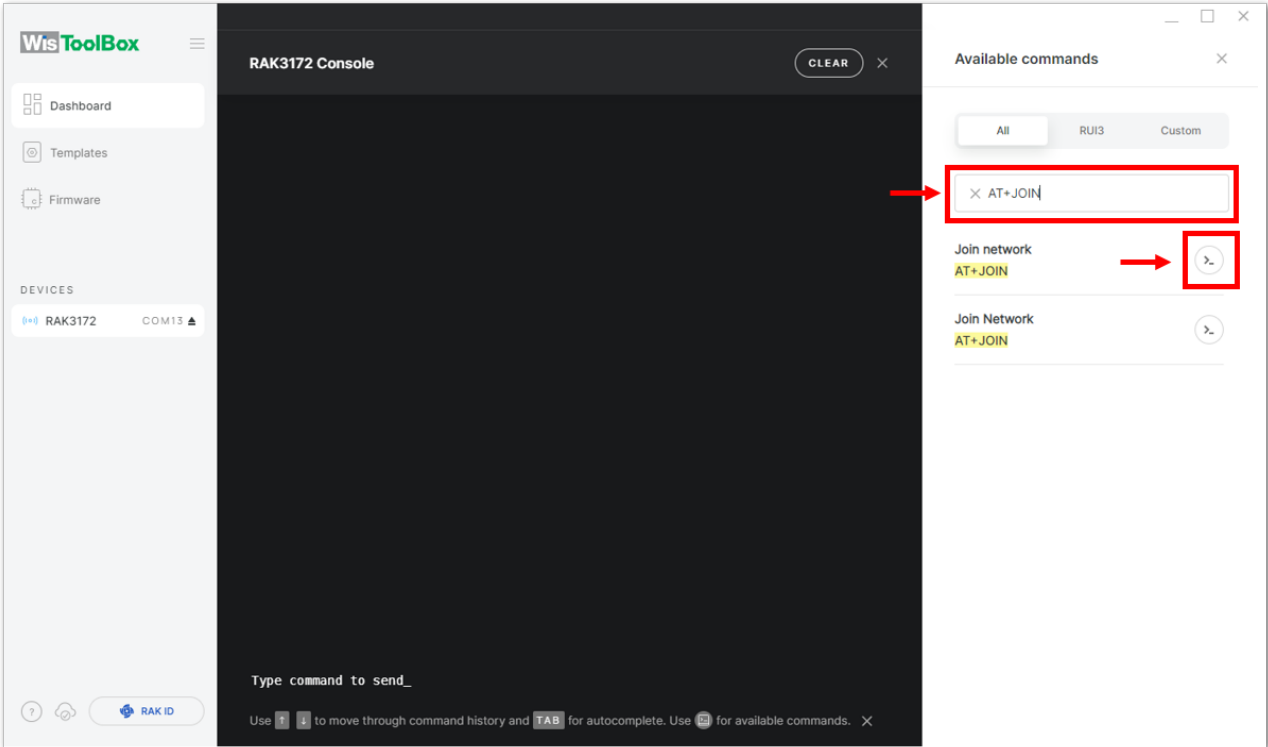


Figure 109: Joining mode using WisToolBox Console

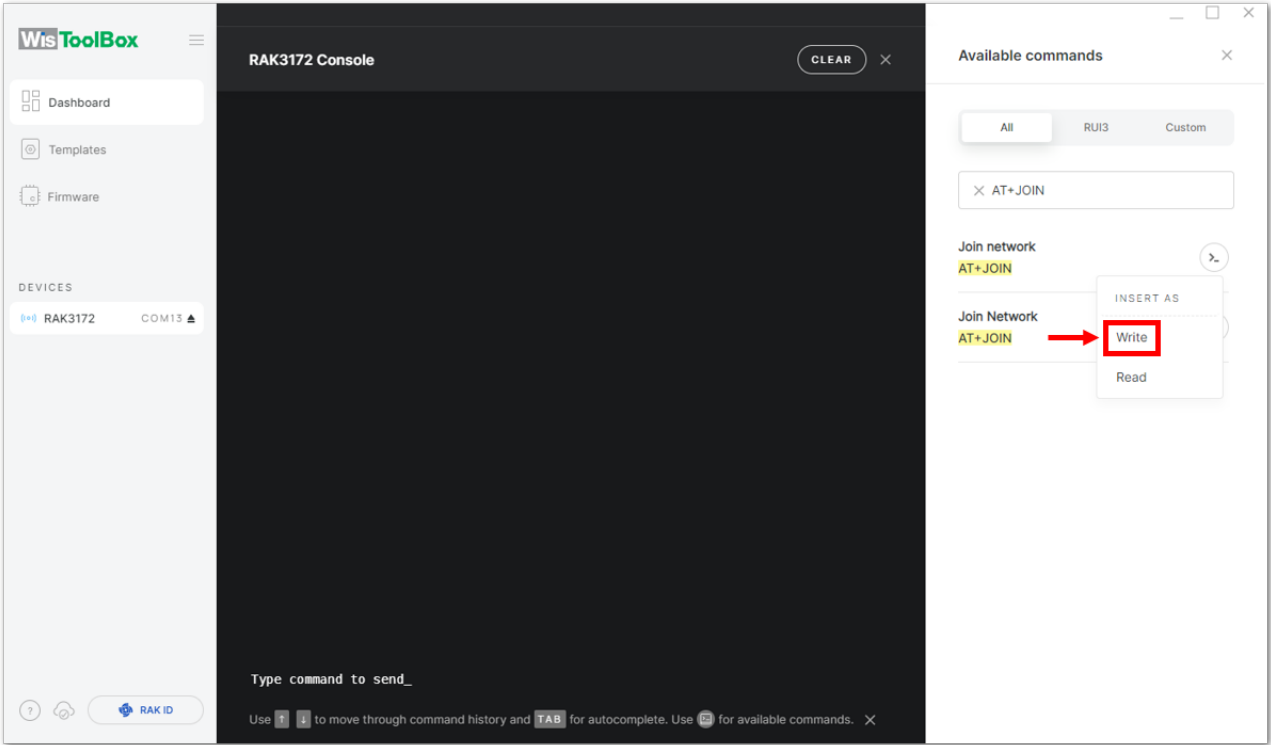


Figure 110: Joining mode using WisToolBox Console

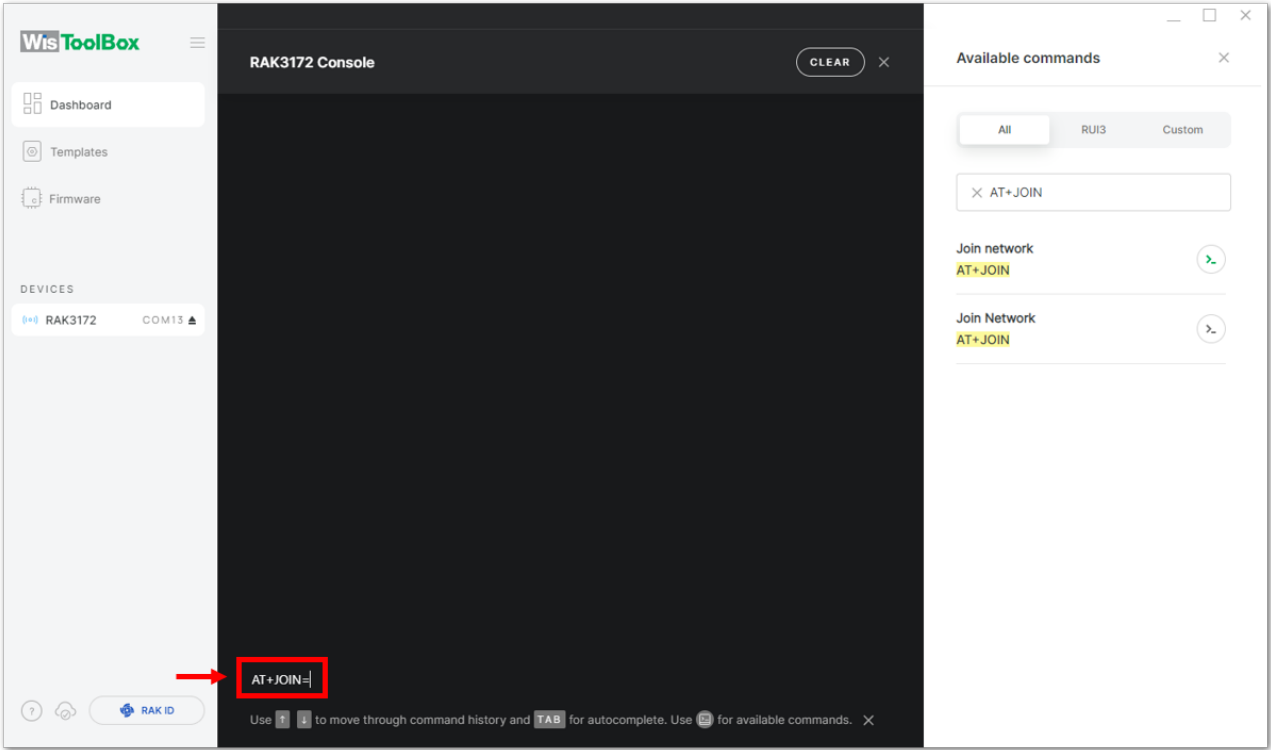


Figure 111: Joining mode using WisToolBox Console

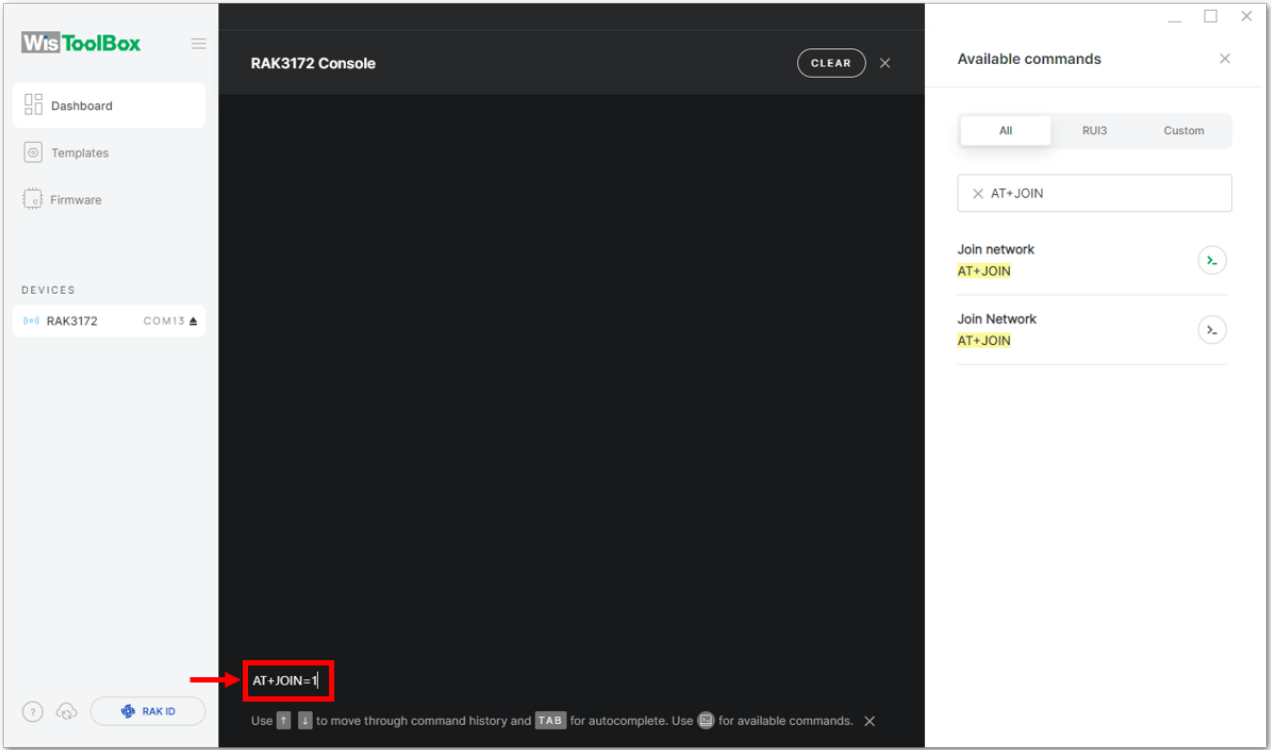


Figure 112: Joining mode using WisToolBox Console

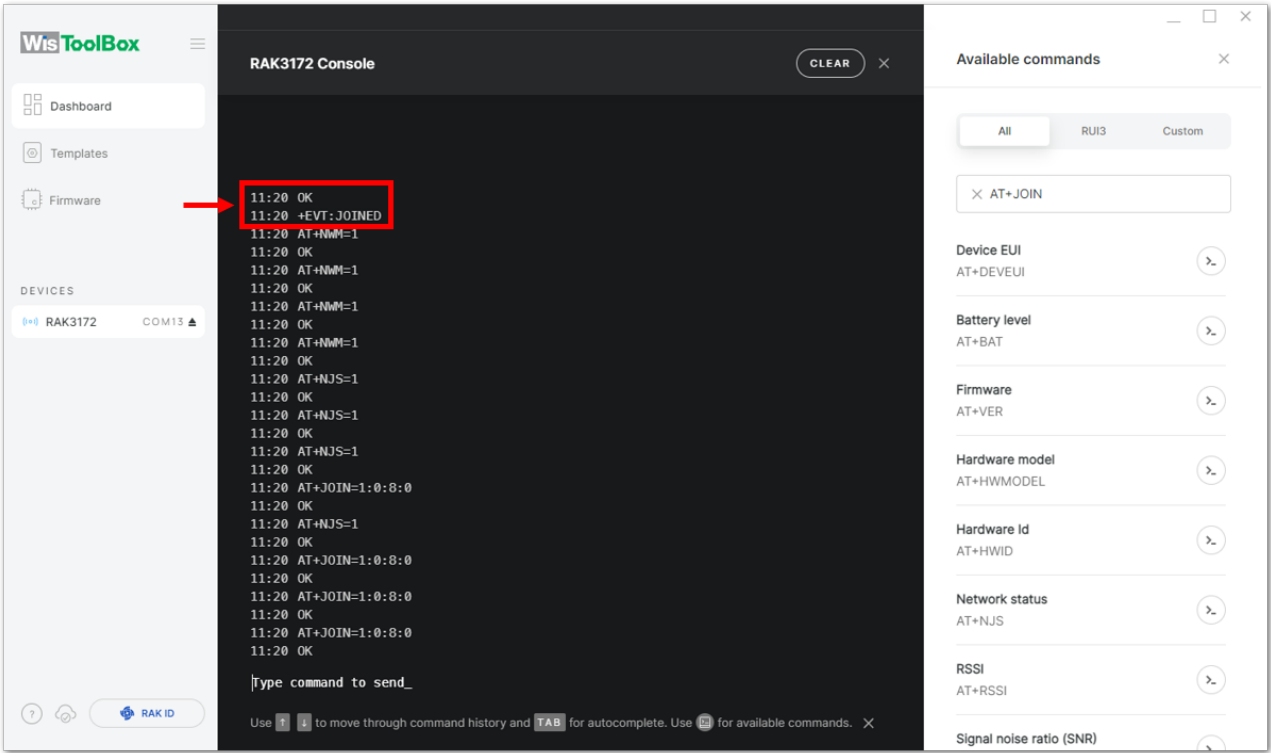


Figure 113: OTAA device successfully joined the network

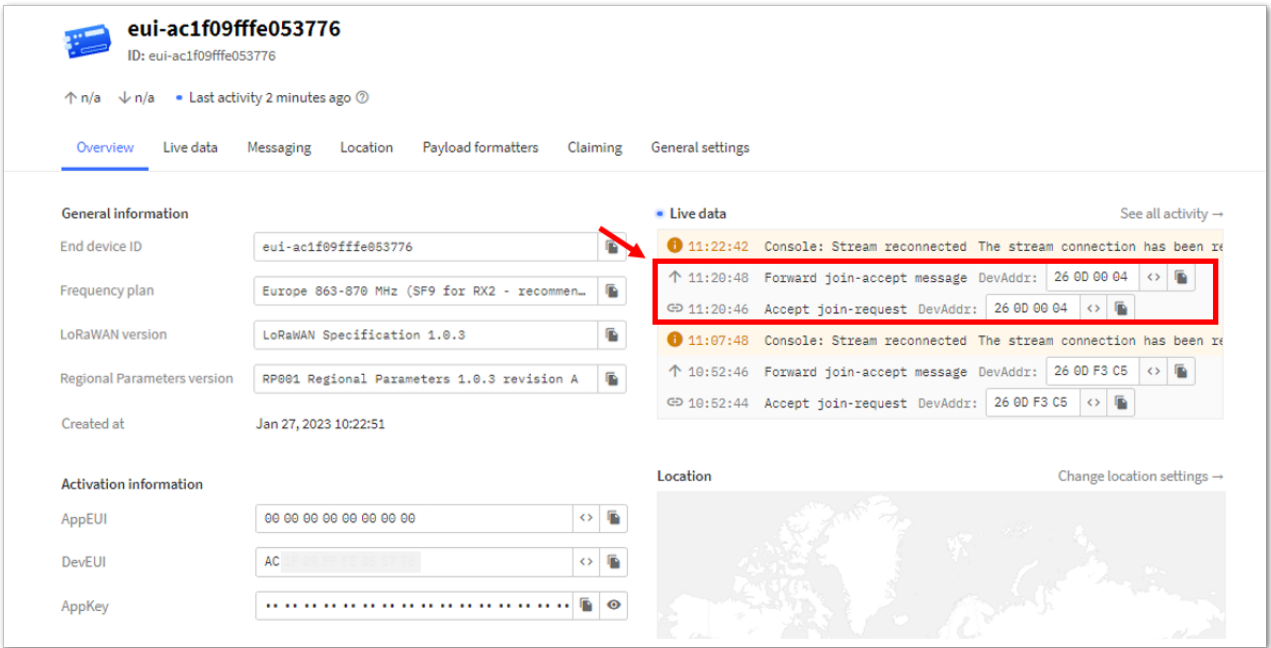


Figure 114: OTAA device successfully joined the network

15. With the end-device properly joined 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
```

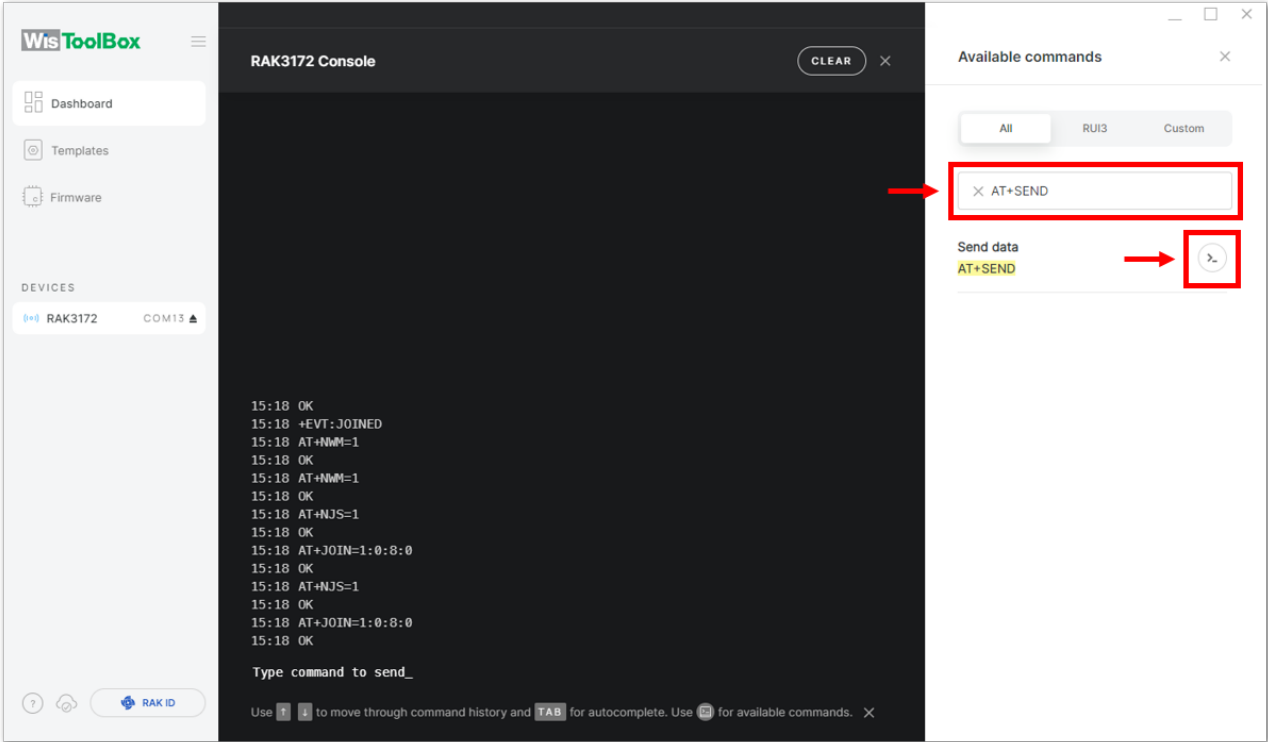


Figure 115: OTAA device sending payload to the network

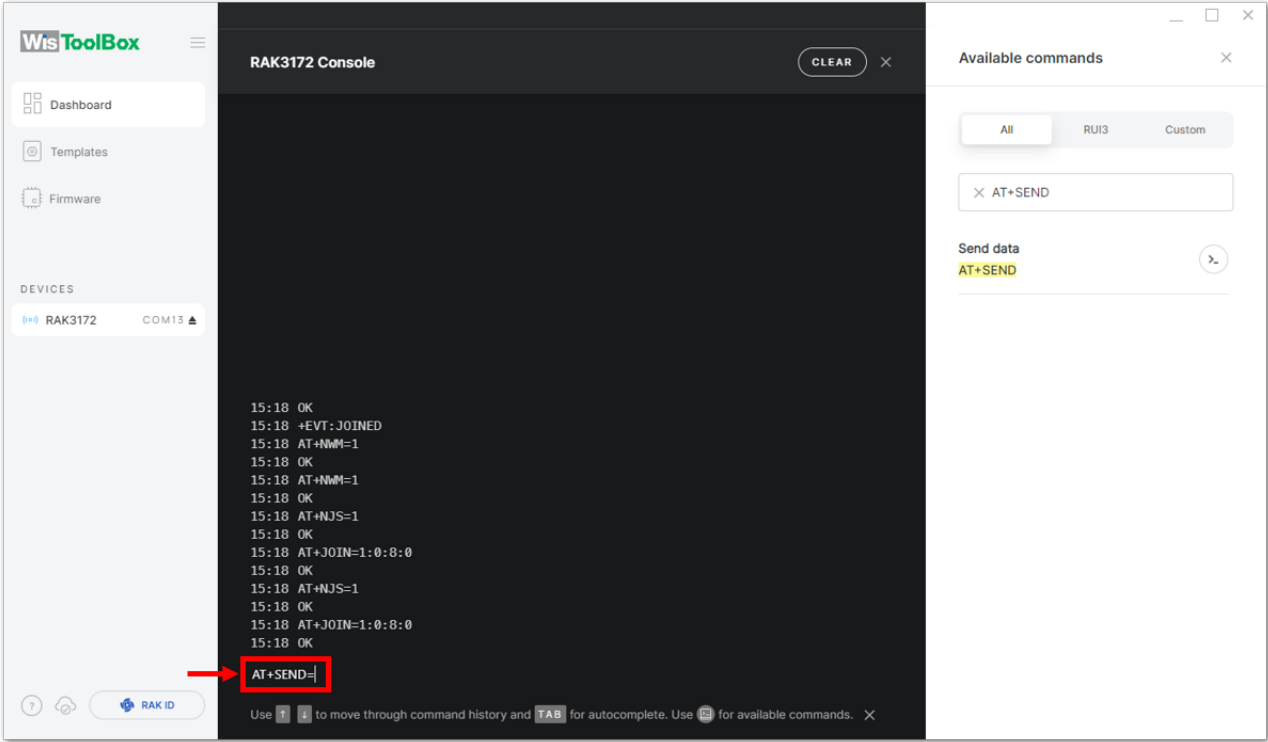


Figure 116: OTAA device sending payload to the network

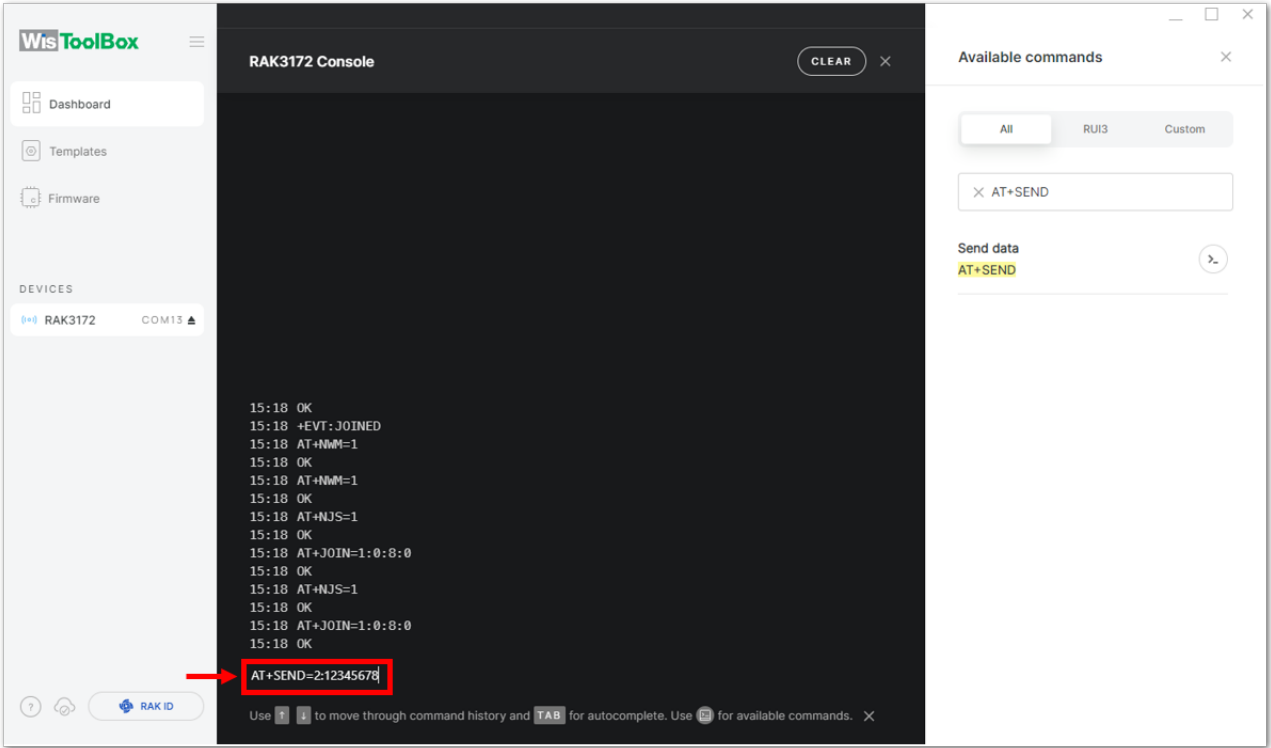


Figure 117: OTAA device sending payload to the network

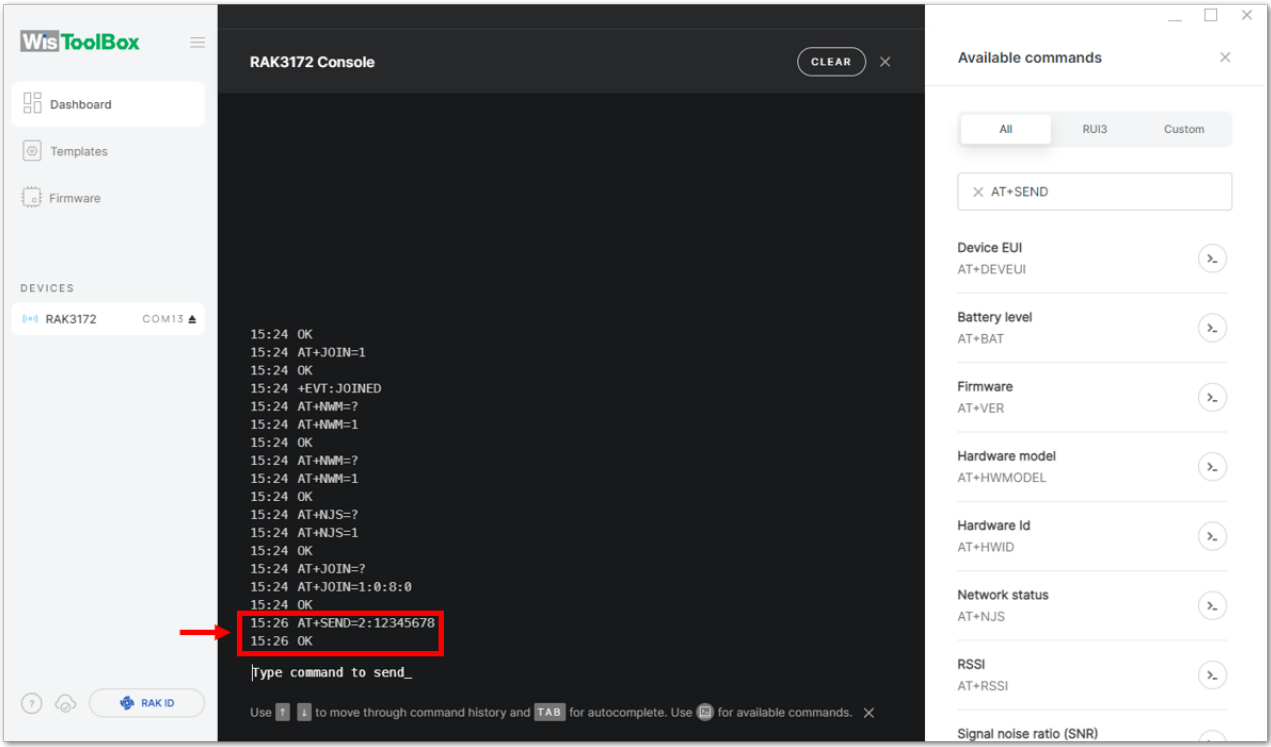


Figure 118: OTAA device sending payload to the network

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.

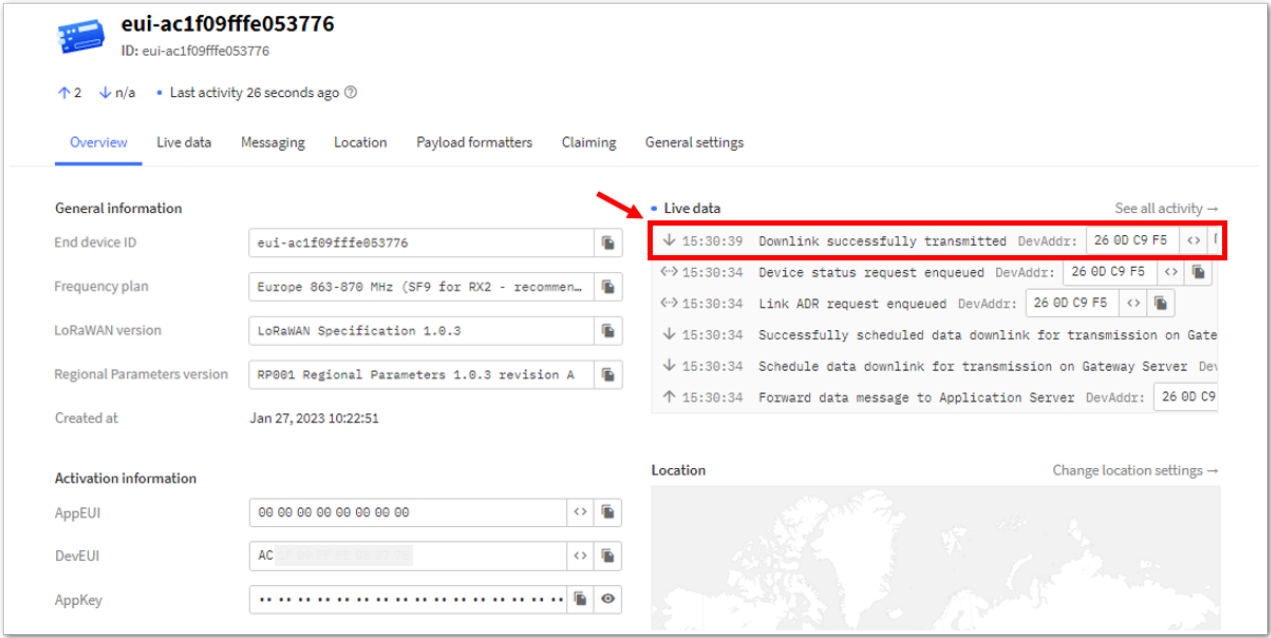


Figure 119: OTAA Test Sample Data Sent Viewed in TTN

TTN ABP Device Registration

1. To register an ABP device, go to your application console and select the application where you want your device to be added. Then click **+ Register end device**, as shown in **Figure 120**.

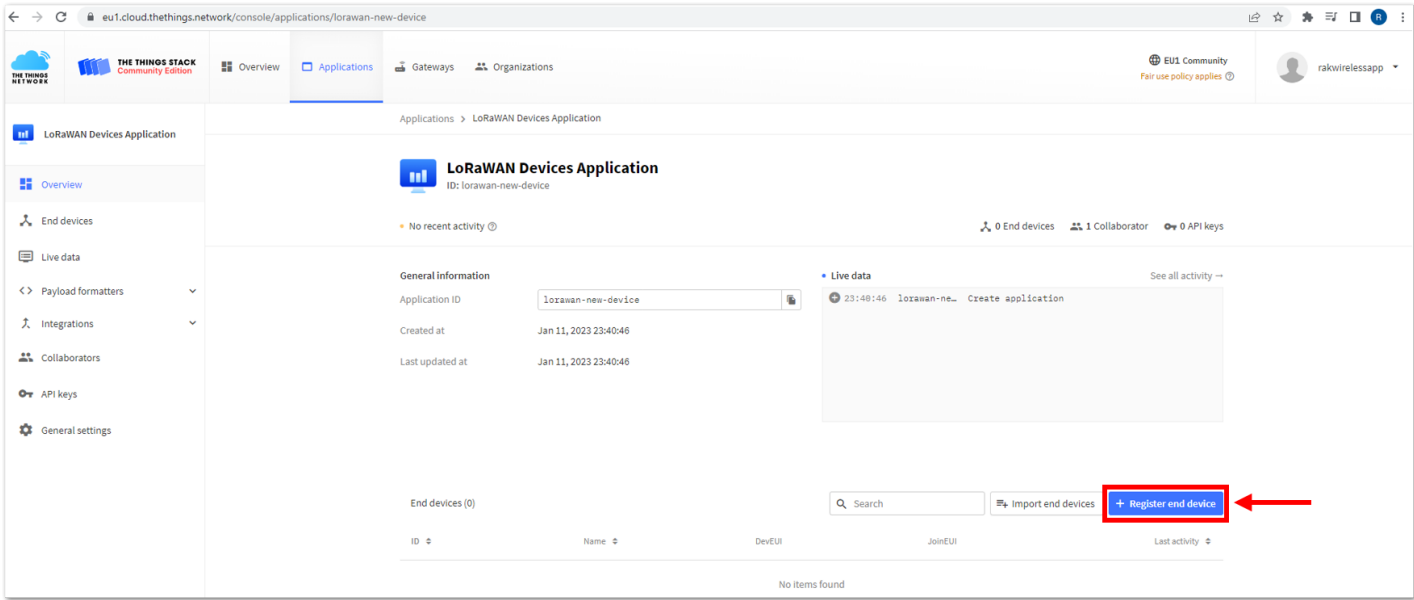


Figure 120: Adding ABP Device

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

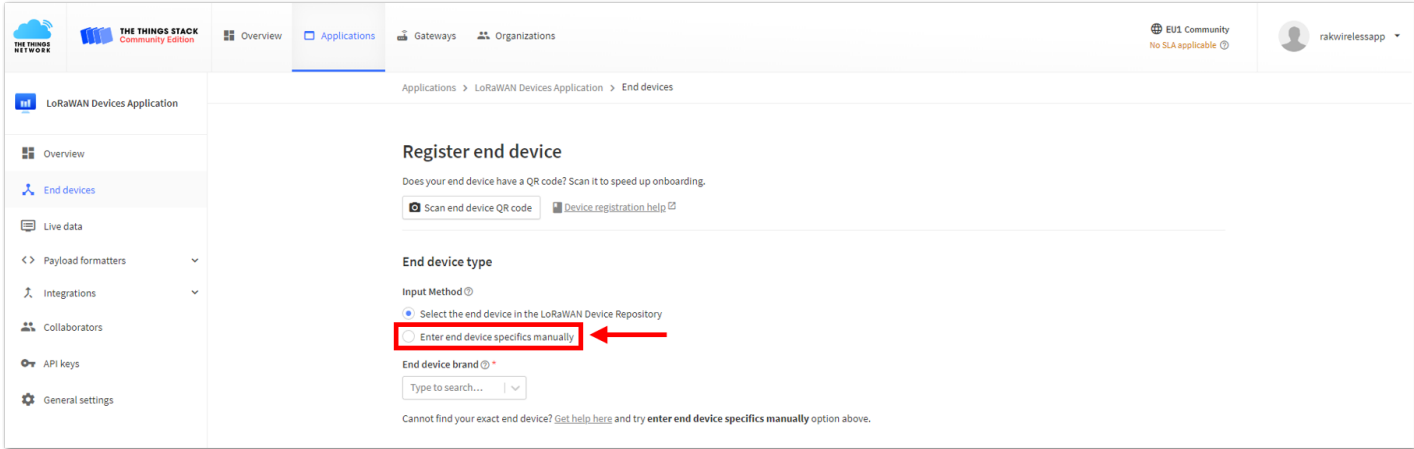


Figure 121: Enter end device specifics manually

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... | v

LoRaWAN version ⓘ *

Select... | v

Regional Parameters version ⓘ *

Select... | v

To continue, please enter versions and frequency plan information

Figure 122: 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) | v

LoRaWAN version ⓘ *

Select... | v

Regional Parameters version ⓘ *

Select... | v

To continue, please enter versions and frequency plan information

Figure 123: 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

.....

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

Figure 124: 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

• • • • •

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

Figure 125: Setting up for your device

→ ☐ Activation by personalization (ABP)

Figure 126: Setting up for your device

- 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 Provisioning information section. Then click **Register end device**.

- 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 RAK3272S device. Also, you should generate your own **Device address**, **AppSKey**, and **NwksKey** credentials for your specific device and application.

DevEUI 

.....

 Generate

0/50 used

Device address  *

.....

 Generate

AppSKey  *

.....

 Generate

NwkSKey  *

.....

 Generate

End device ID  *

my-new-device

This value is automatically prefilled using the DevEUI

Figure 127: 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 128: 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 129: Setting up for your device

Provisioning information

DevEUI ⓘ

AC 1F 89 FF FE 85 36 DF

Generate

0/50 used

Device address ⓘ *

26 88 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 130: Setting up for your device

Provisioning information

DevEUI ⓘ

AC 1F 89 FF FE 85 36 DF

Generate

0/50 used

Device address ⓘ *

26 88 52 DC

Generate

AppSKey ⓘ *

1F FA 86 B1 6F 7D 3E CF 3A AA 11 A5 EA B8 1B 4C

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 131: Setting up for your device

Provisioning information

DevEUI ⓘ

AC 1F 80 FF FE 85 36 DF

Generate

0/50 used

Device address ⓘ *

26 80 82 DC

Generate

AppSKey ⓘ *

1F FA 80 81 6F 7D 3E CE 3A AA 11 A5 EA 88 1B 4C

Generate

NwkSKey ⓘ *

92 DD 80 81 6E 9F 78 31 F6 BE 47 96 CC FA 81 D7

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 132: 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 133.

THE THINGS STACK
Community Edition

OverviewApplicationsGatewaysOrganizations

Community
No support plan ⓘ

rakwirelessapp

LoRaWAN Devices Application

OverviewEnd devicesLive dataPayload formattersIntegrationsCollaboratorsAPI keysGeneral settings

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

eui-ac1f09fffe0536df

ID: eui-ac1f09fffe0536df

↑ n/a ↓ n/a • No activity yet ⓘ

OverviewLive dataMessagingLocationPayload formattersGeneral settings

General information

End device IDeui-ac1f09fffe0536df

Frequency planEurope 863-870 MHz (SF9 for RX2 - recommen...

LoRaWAN versionLoRaWAN Specification 1.0.3

Regional Parameters versionRP001 Regional Parameters 1.0.3 revision A

Created atJan 16, 2023 23:23:20

Activation information

AppEUI n/a

DevEUIAC

Session information

Session startJan 16, 2023 23:23:21

Device address26

NwkSKey

SNwkSIntKey

NwkSEncKey

AppSKey

Live data

23:23:20 Create end device DevAddr: 26 80 82 DC ⓘ

See all activity →

Location

Change location settings →

No location information available

Figure 133: ABP device successfully registered to TTN

ABP Configuration for TTN

The RAK3272 Breakout Board which has a RAK3172 module in it can be configured using WisToolBox to do the ABP configuration. **WisToolBox** is a software tool that supports **RAK3172** module. It automatically detects RAK3172 module once connected to PC. Below are the options in WisToolBox that 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 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 **CONNECT DEVICE** button to launch the WisToolBox Dashboard.

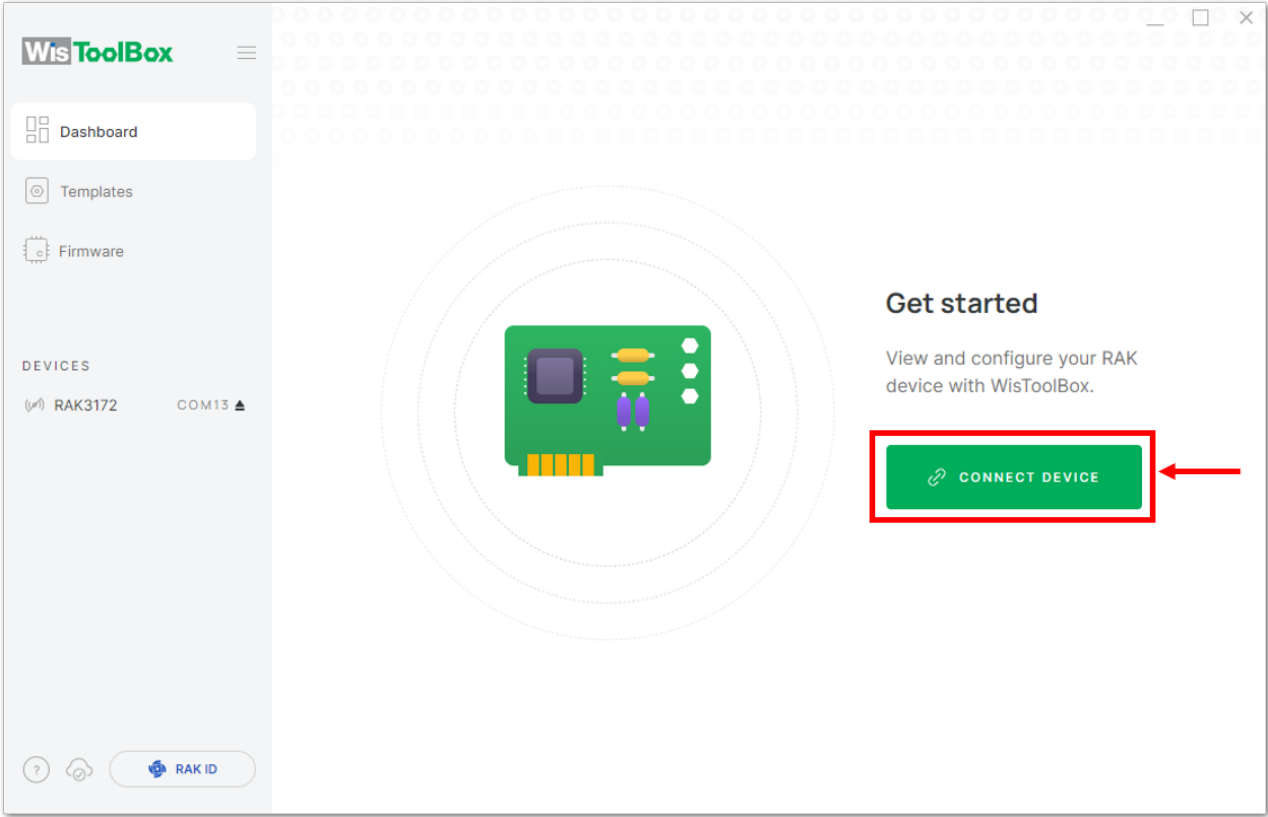


Figure 134: CONNECT DEVICE

3. Then select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT** as shown in Figure 136.

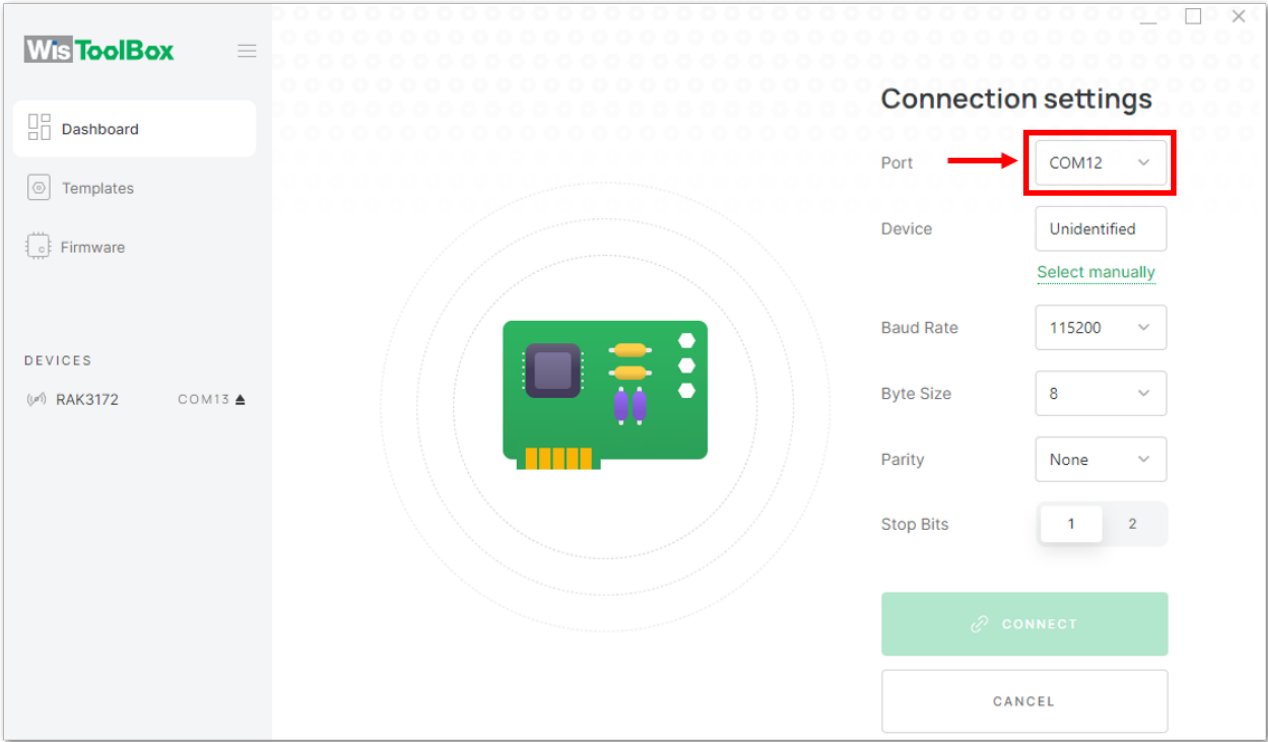


Figure 135: Setting up your device

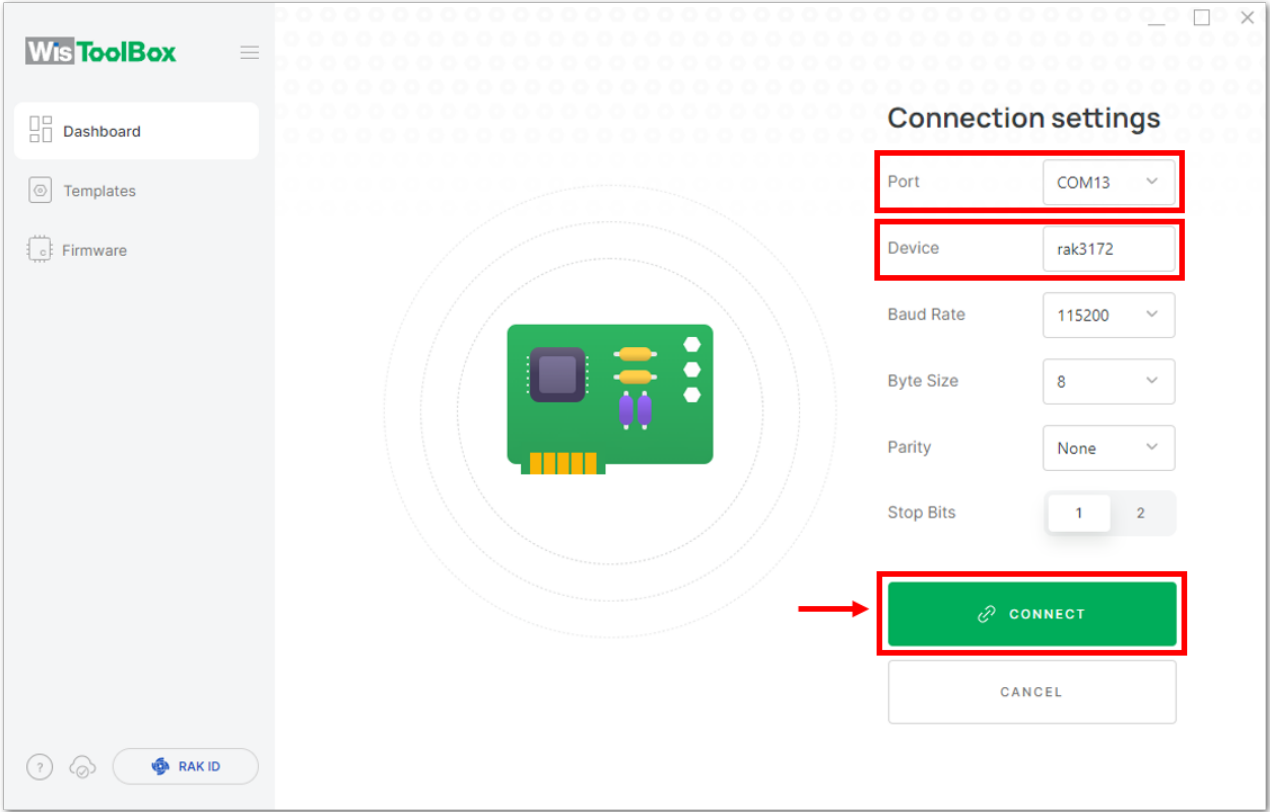


Figure 136: Setting up your device

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

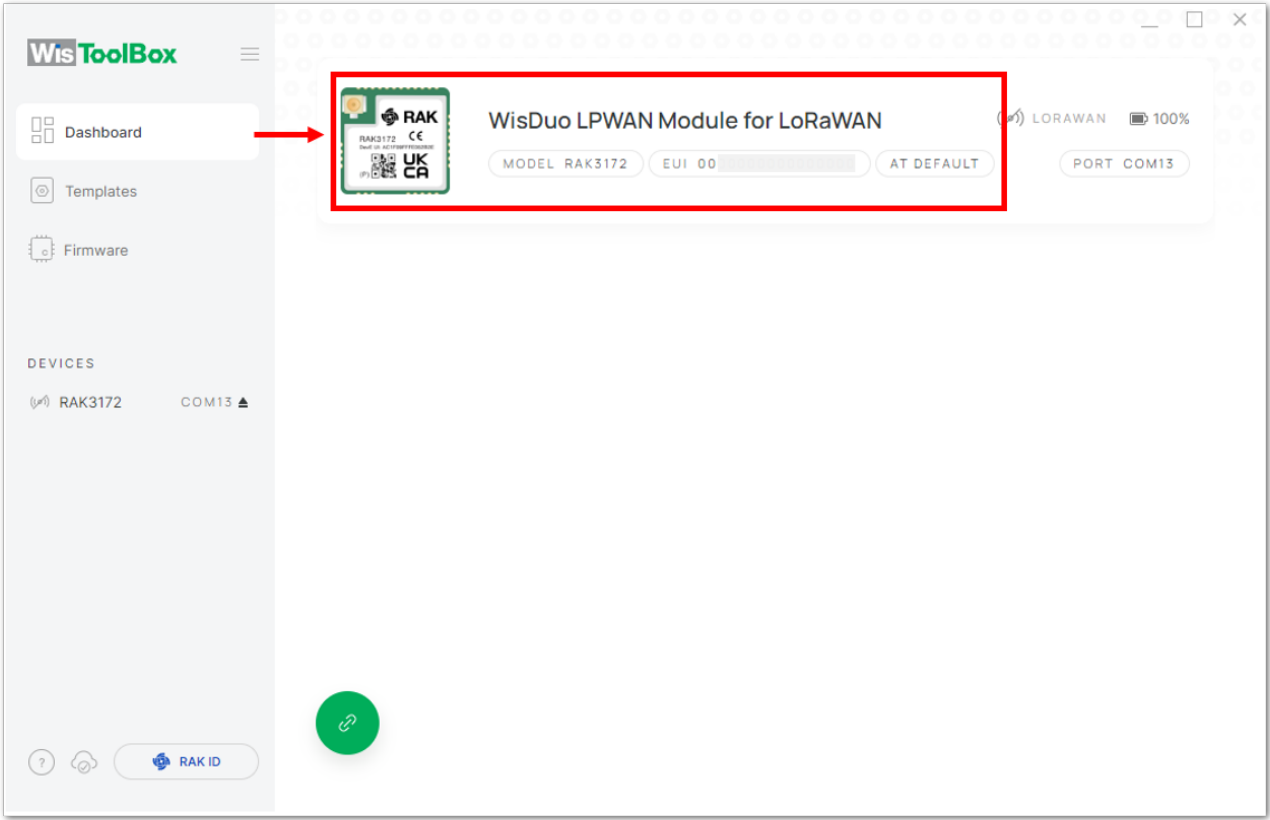


Figure 137: Device seen from WisToolBox dashboard

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.

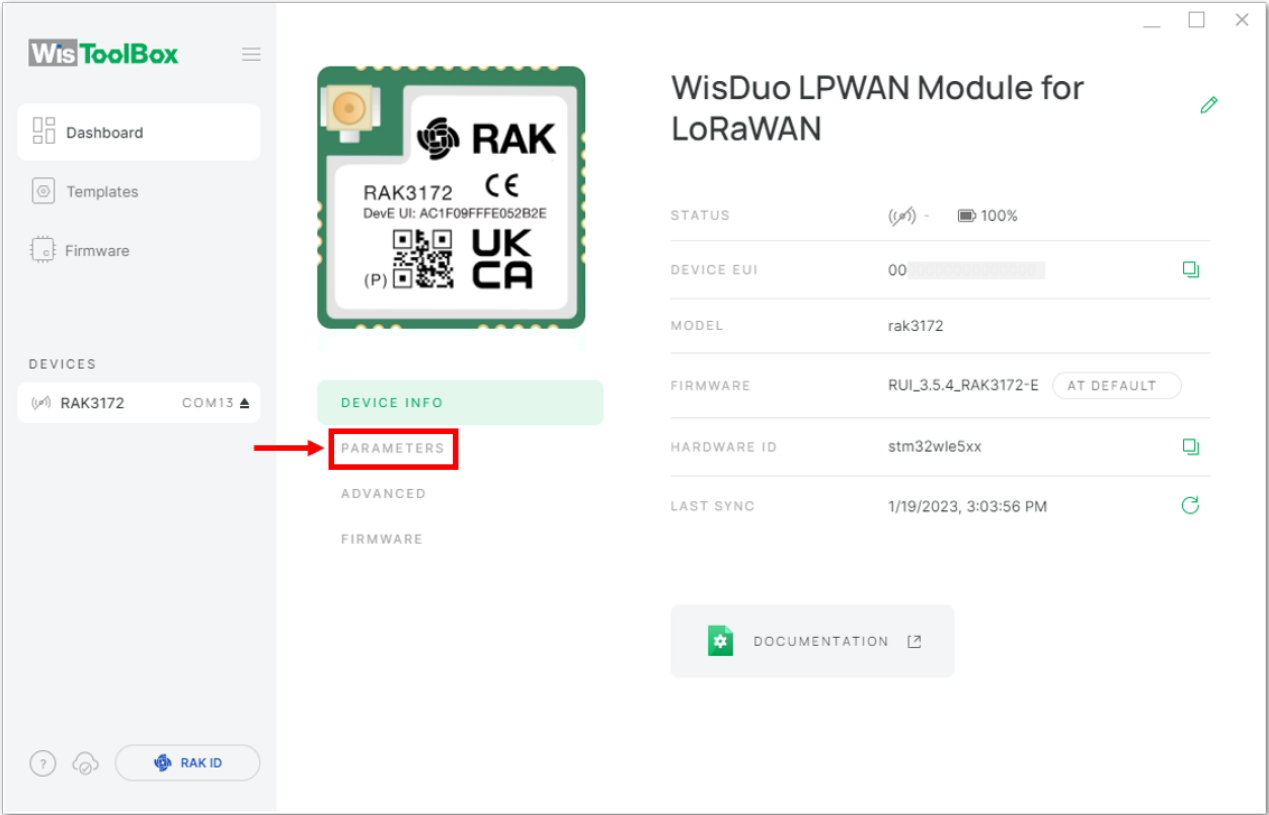


Figure 138: Setting up your device

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 band, you can choose among active regions based on your location.
- LoRa network mode: **LoRaWAN**
 - LoRaWAN join mode: **ABP**
 - LoRaWAN region: **EU868**

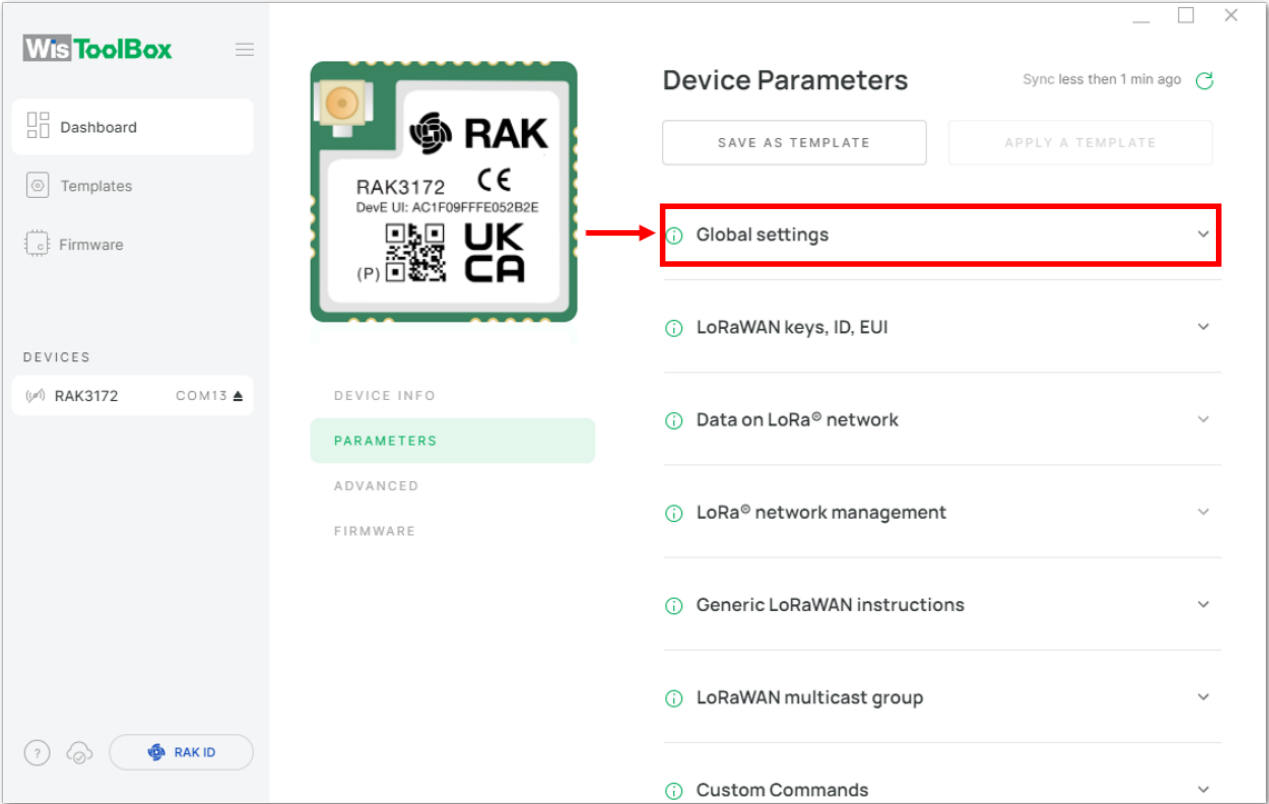


Figure 139: Global settings

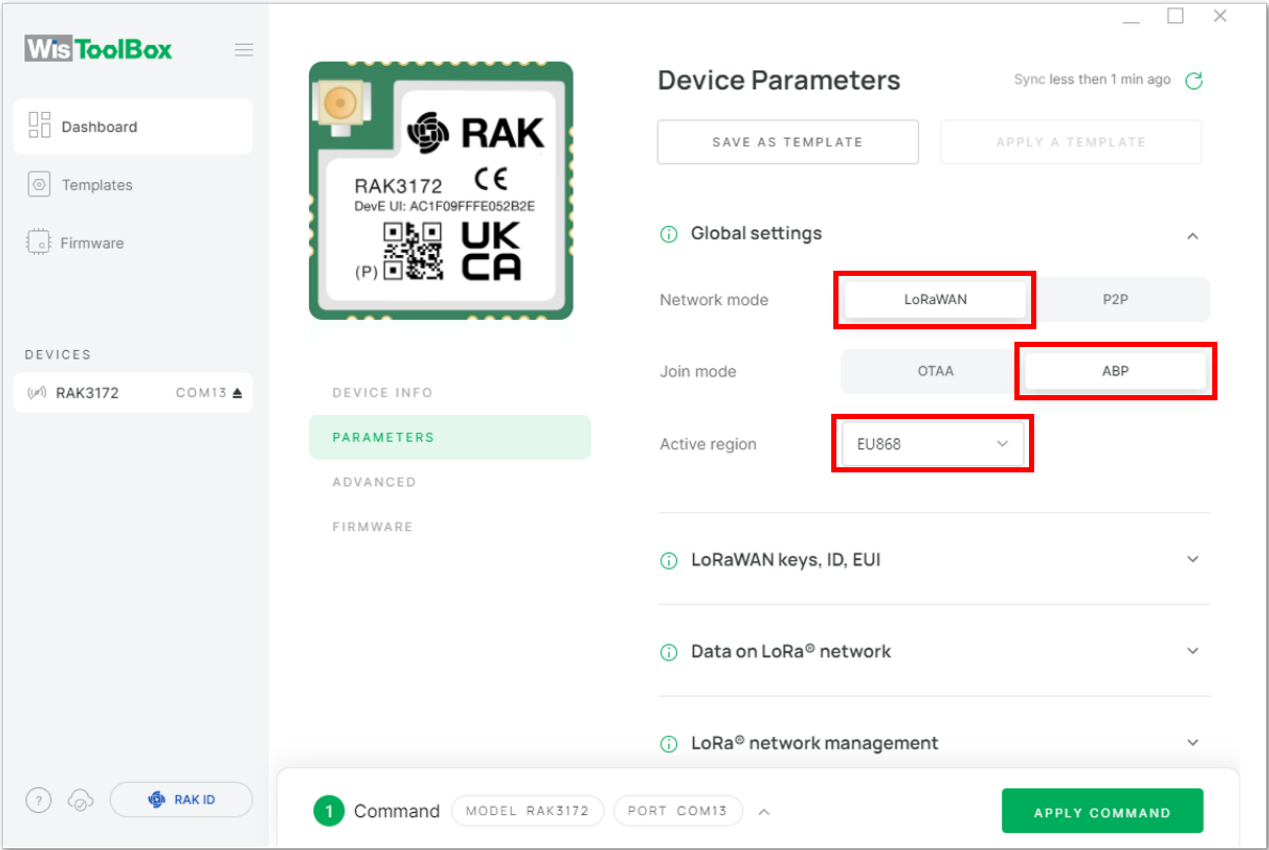


Figure 140: Global settings

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

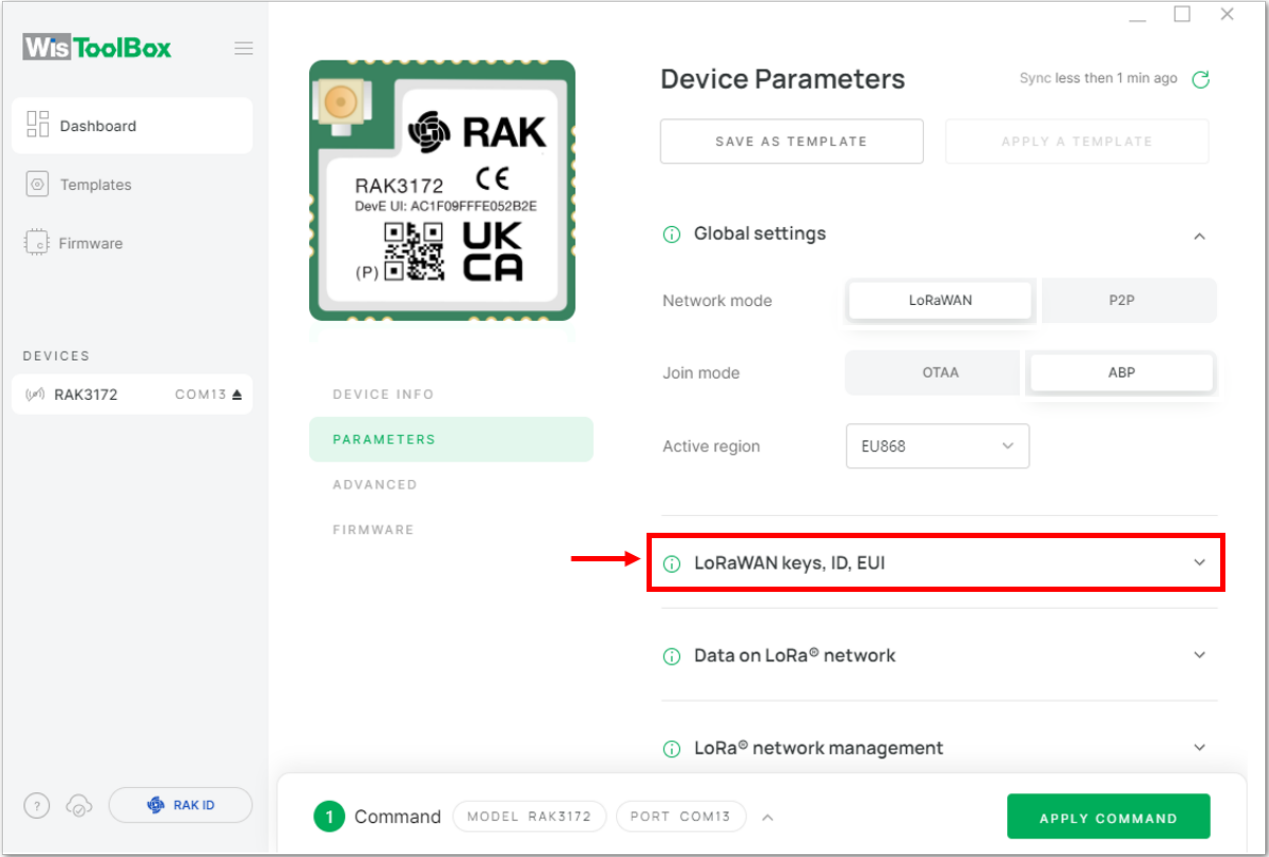


Figure 141: LoRaWAN keys, ID, EUI

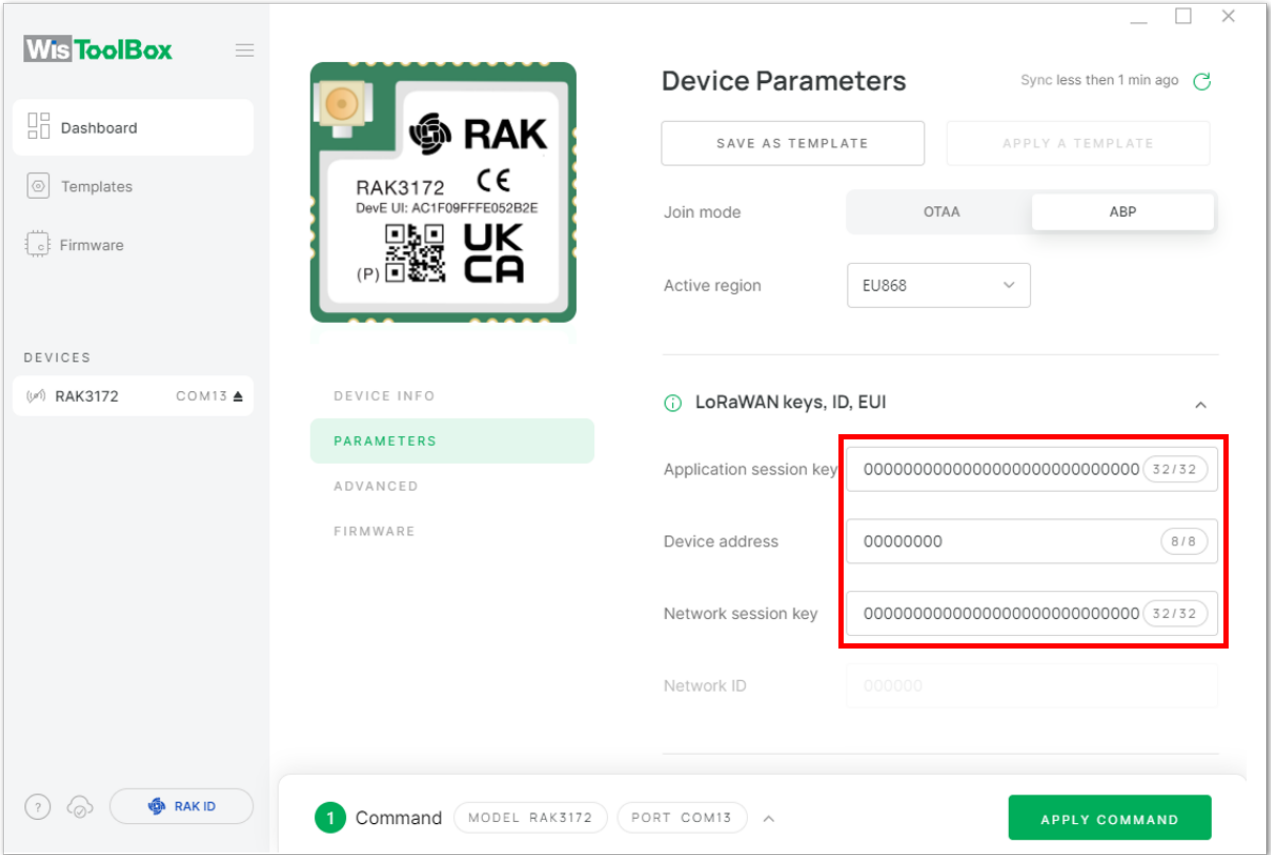


Figure 142: Setting up your device

8. Then go back to 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 150**.

 **NOTE:**

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

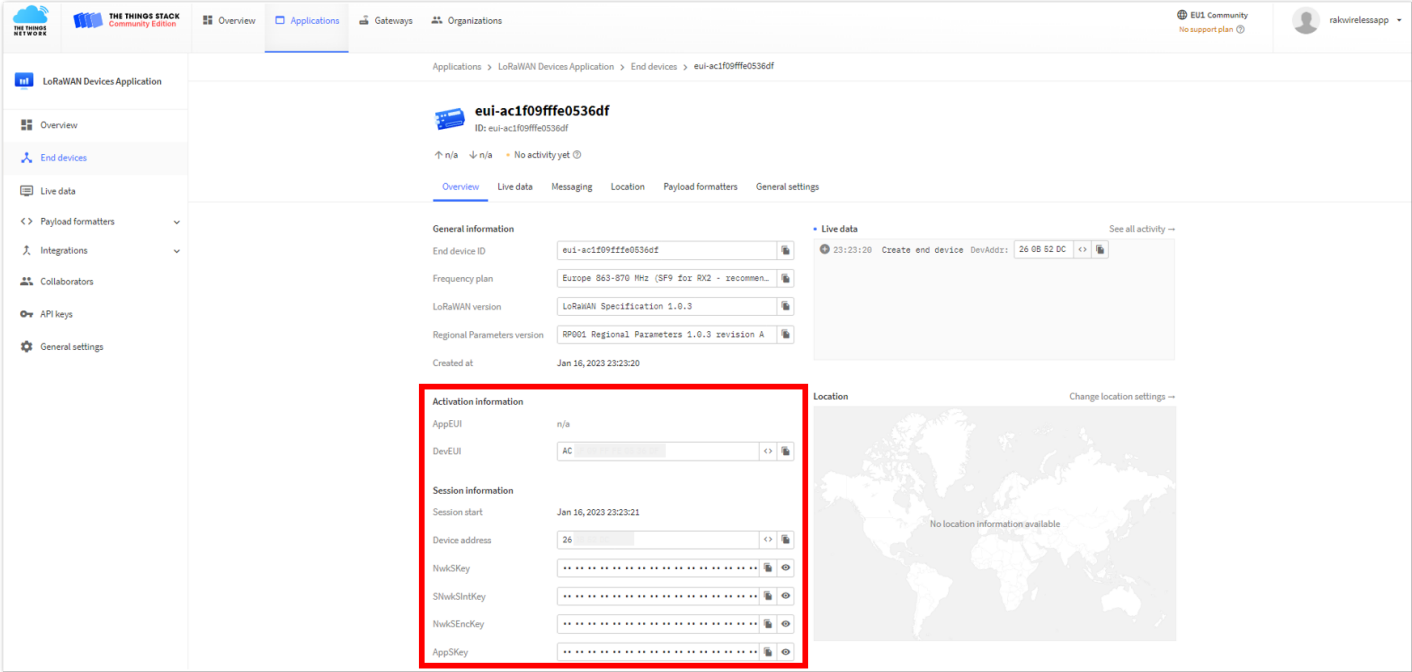


Figure 143: Your created ABP device from your console

- For Application session key (AppSKey)

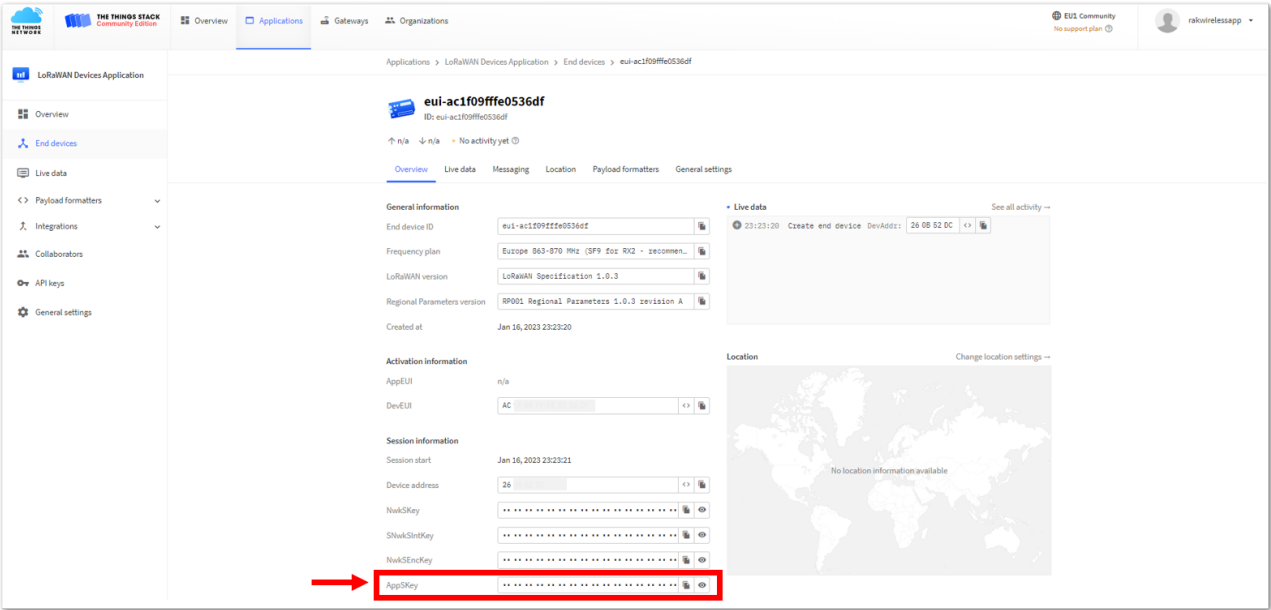


Figure 144: Copying the AppKey credential from TTN to WisToolBox

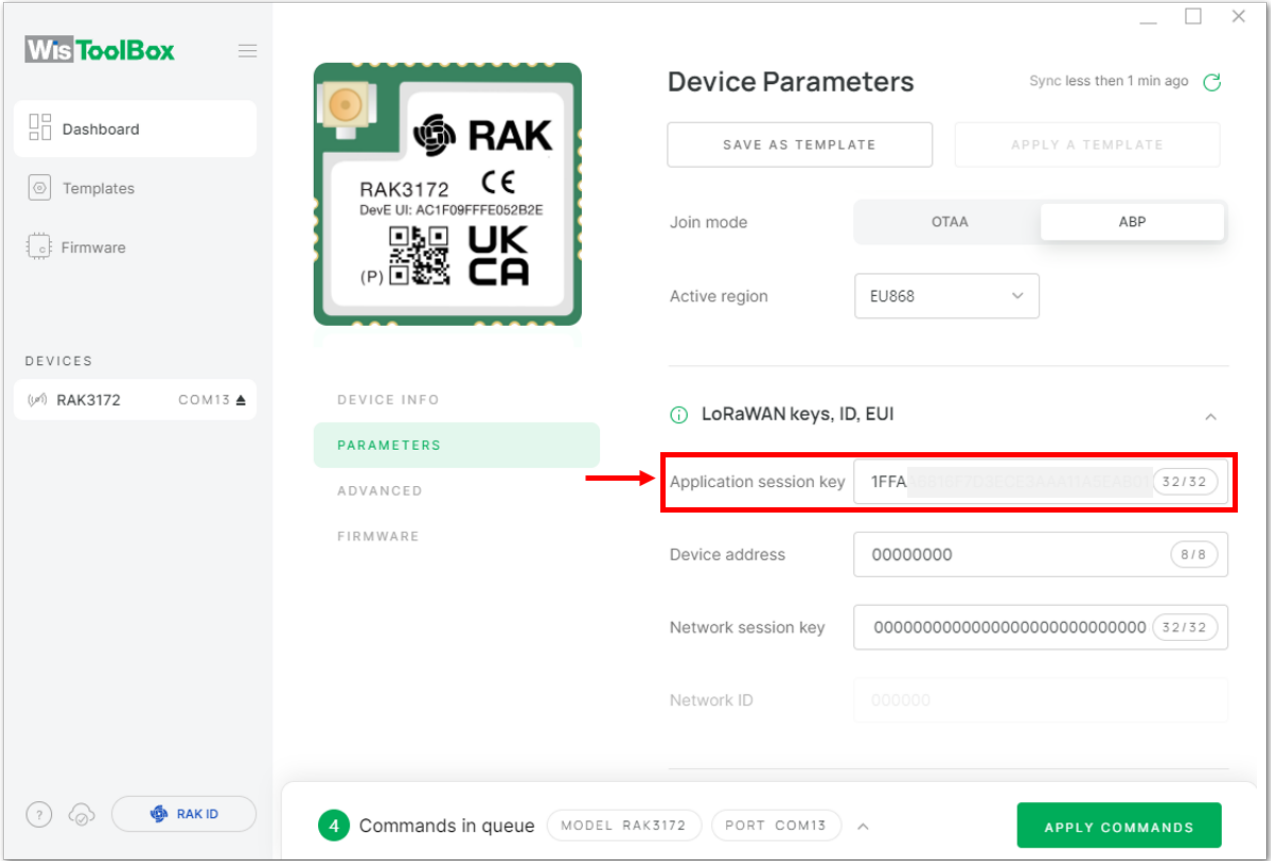


Figure 145: Copying the AppKey credential from TTN to WisToolBox

- For Device address

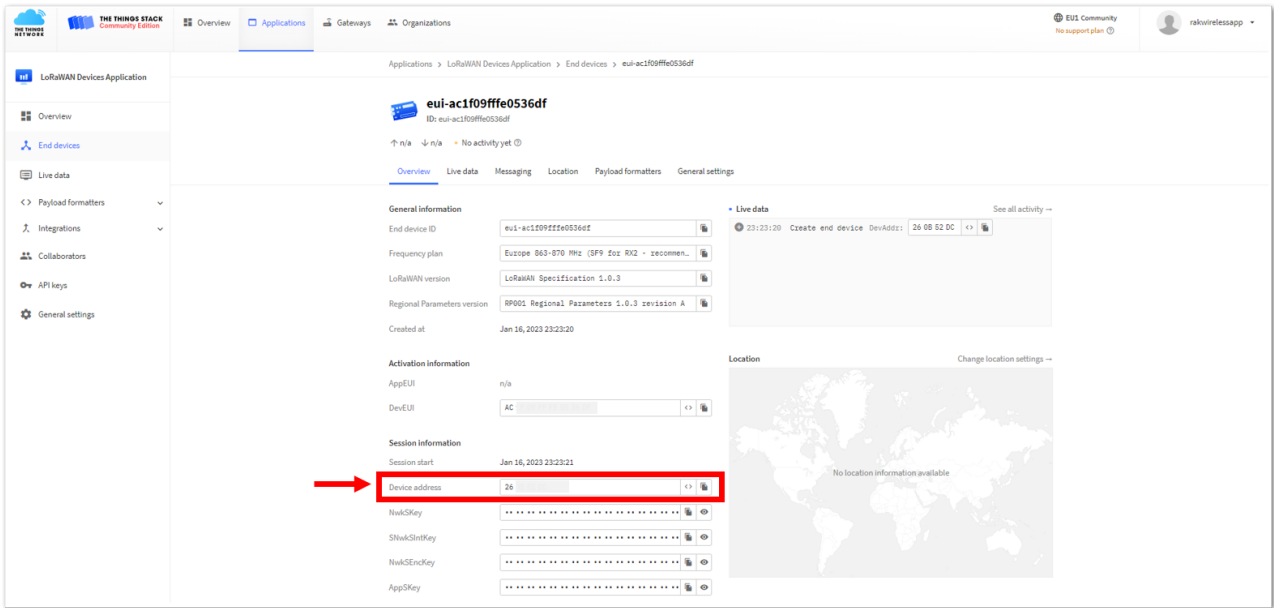


Figure 146: Copying the Device address credential from TTN to WisToolBox

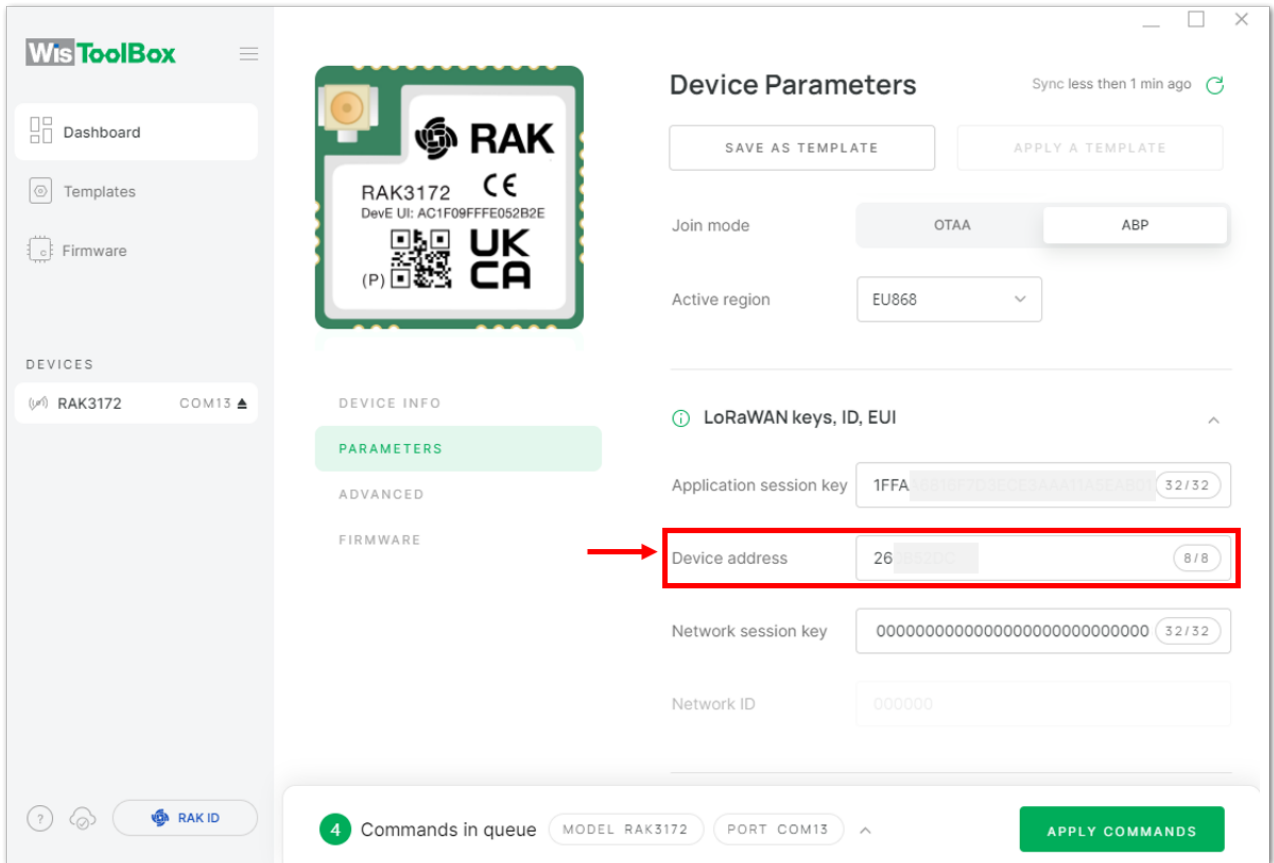


Figure 147: Copying the Device address credential from TTN to WisToolBox

- For Network session key (NwkSKey)

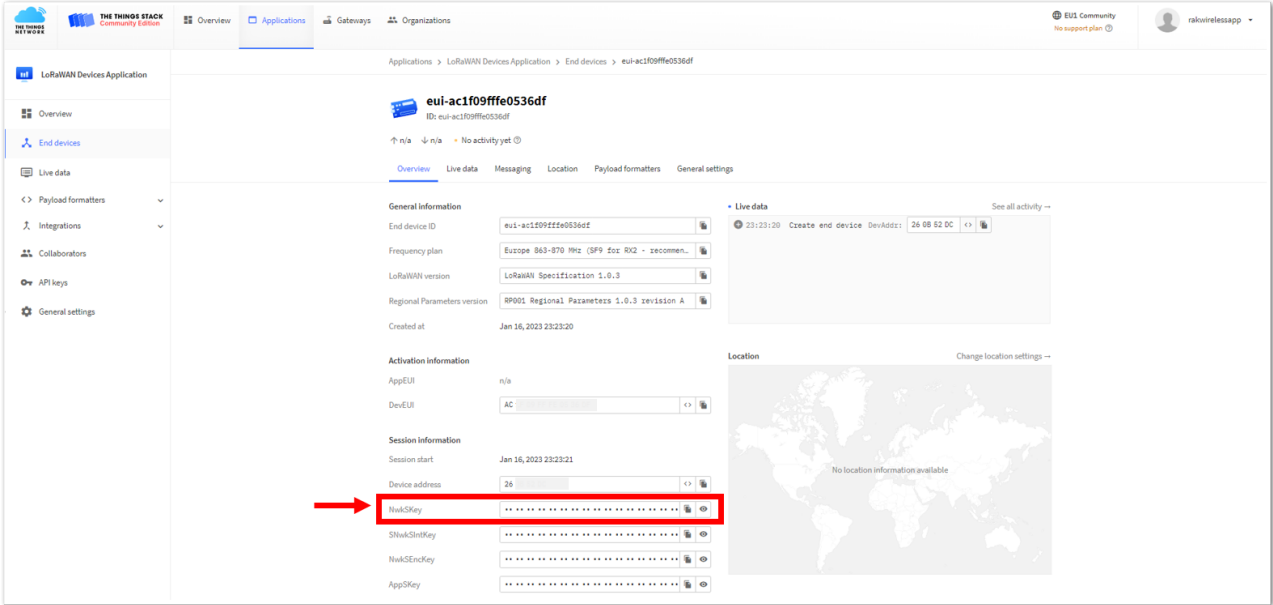


Figure 148: Copying the NwkKey credential from TTN to WisToolBox

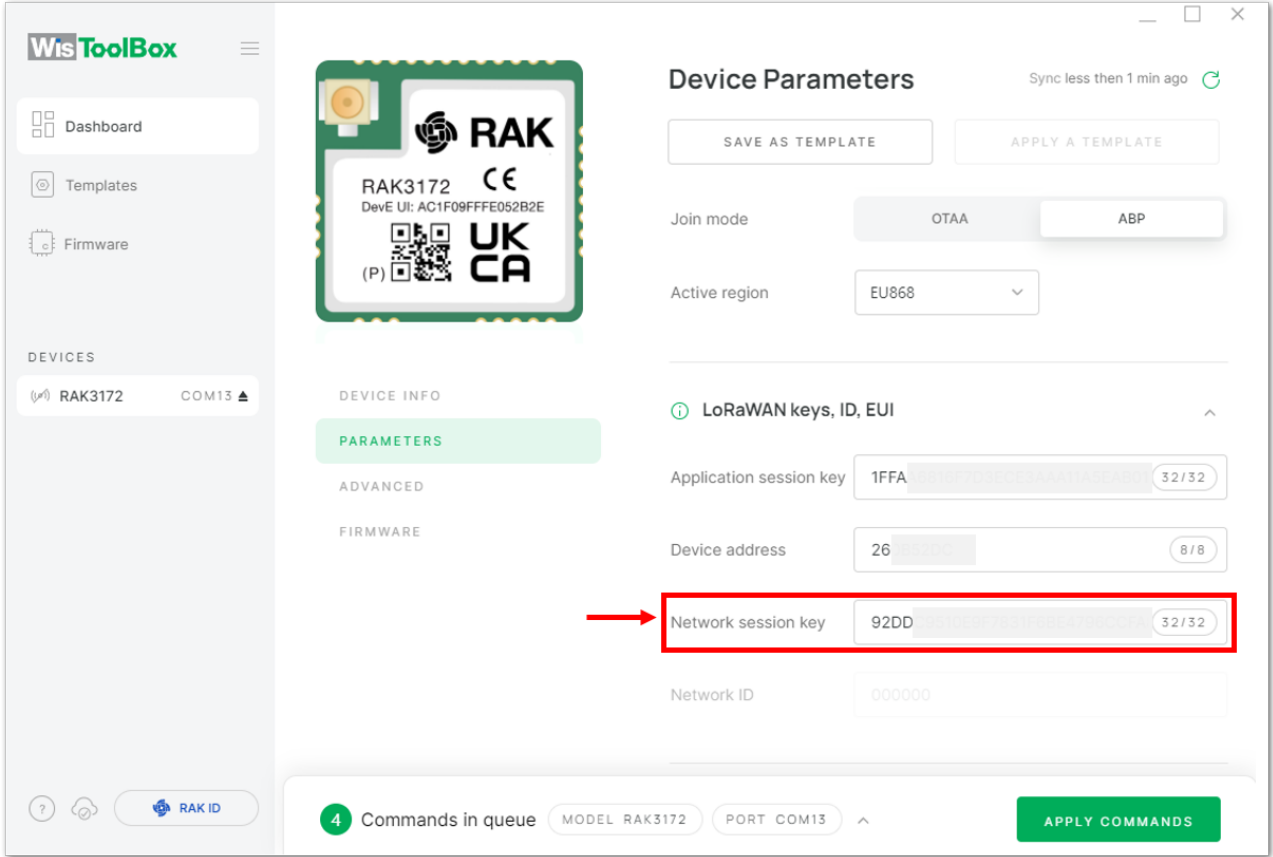


Figure 149: Copying the NwkKey credential from TTN to WisToolBox

- WisToolBox Dashboard

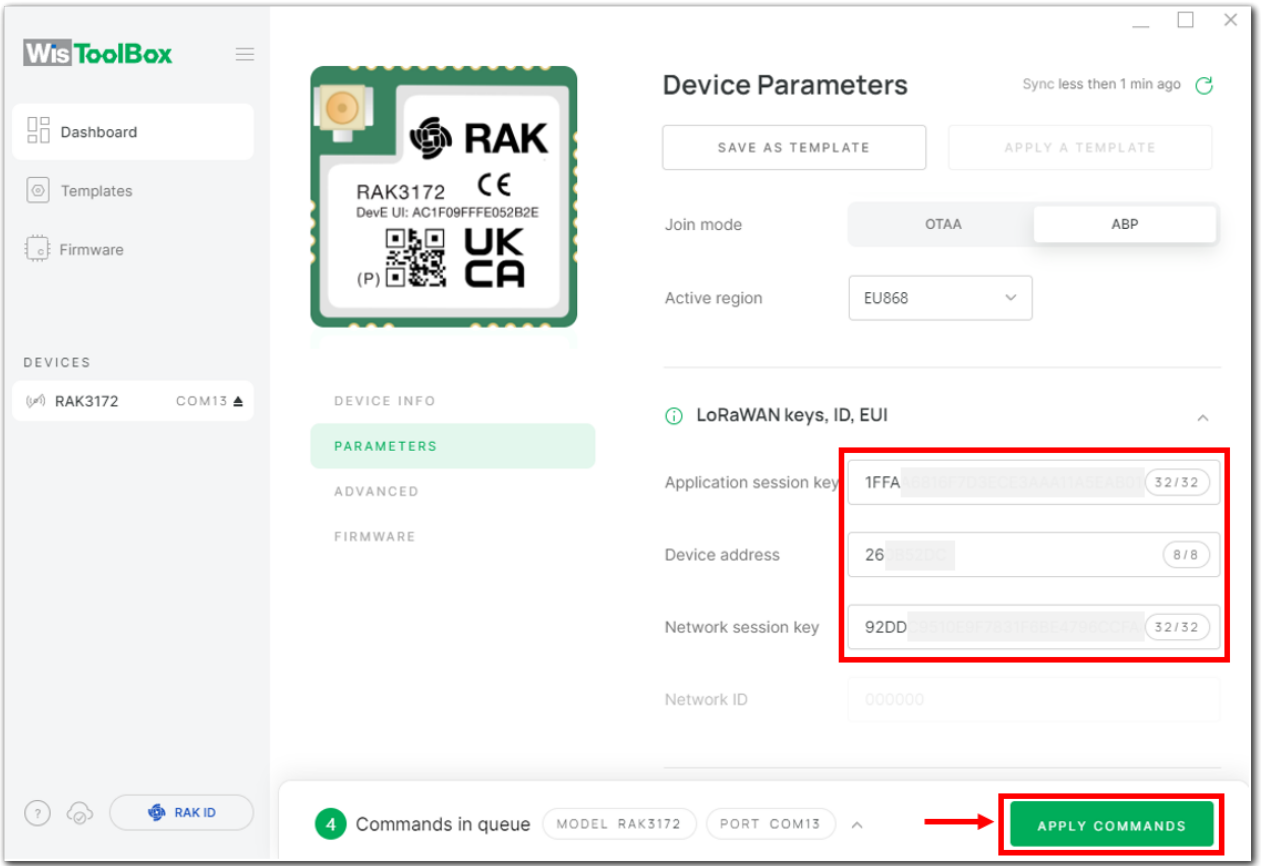


Figure 150: Used credentials from your console in WisToolBox dashboard

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

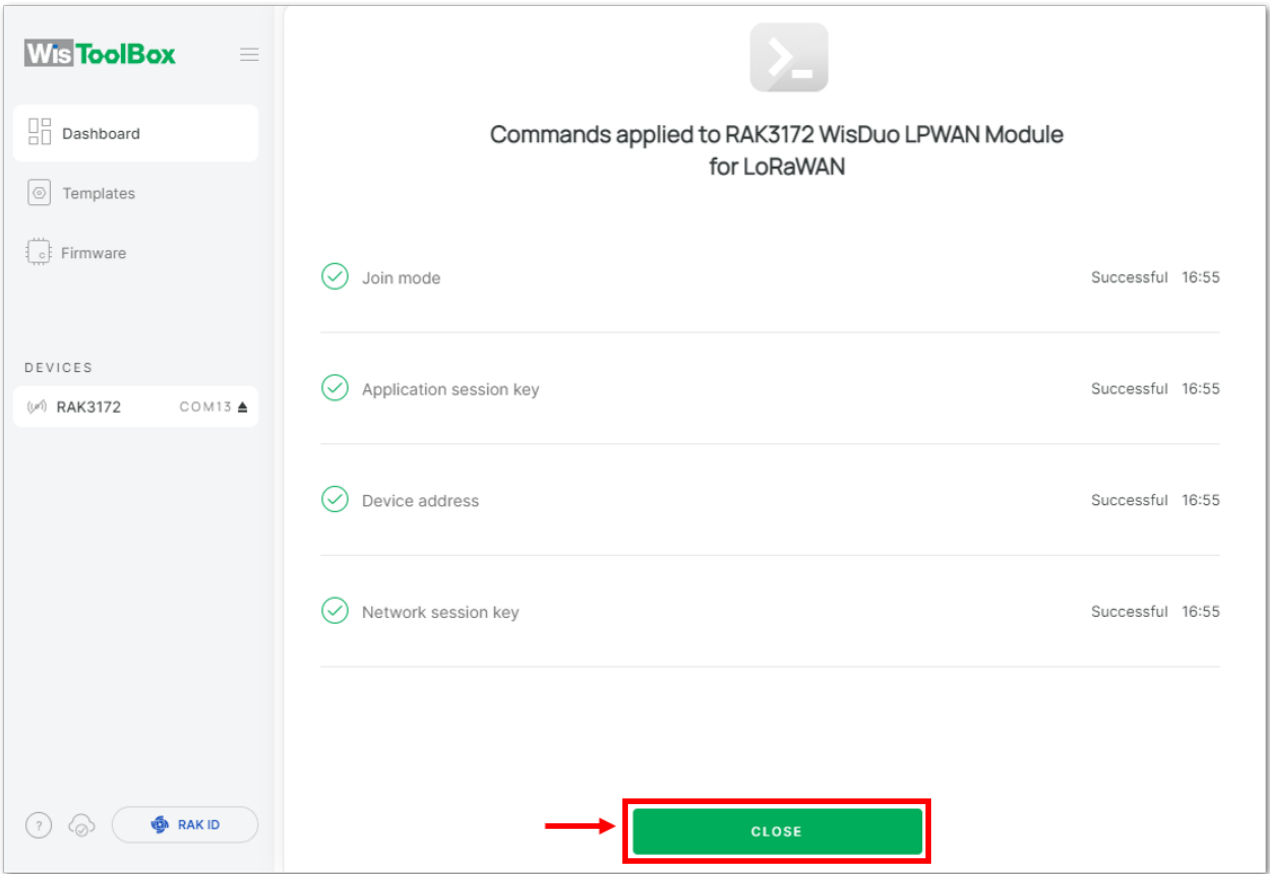


Figure 151: Summary of commands

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

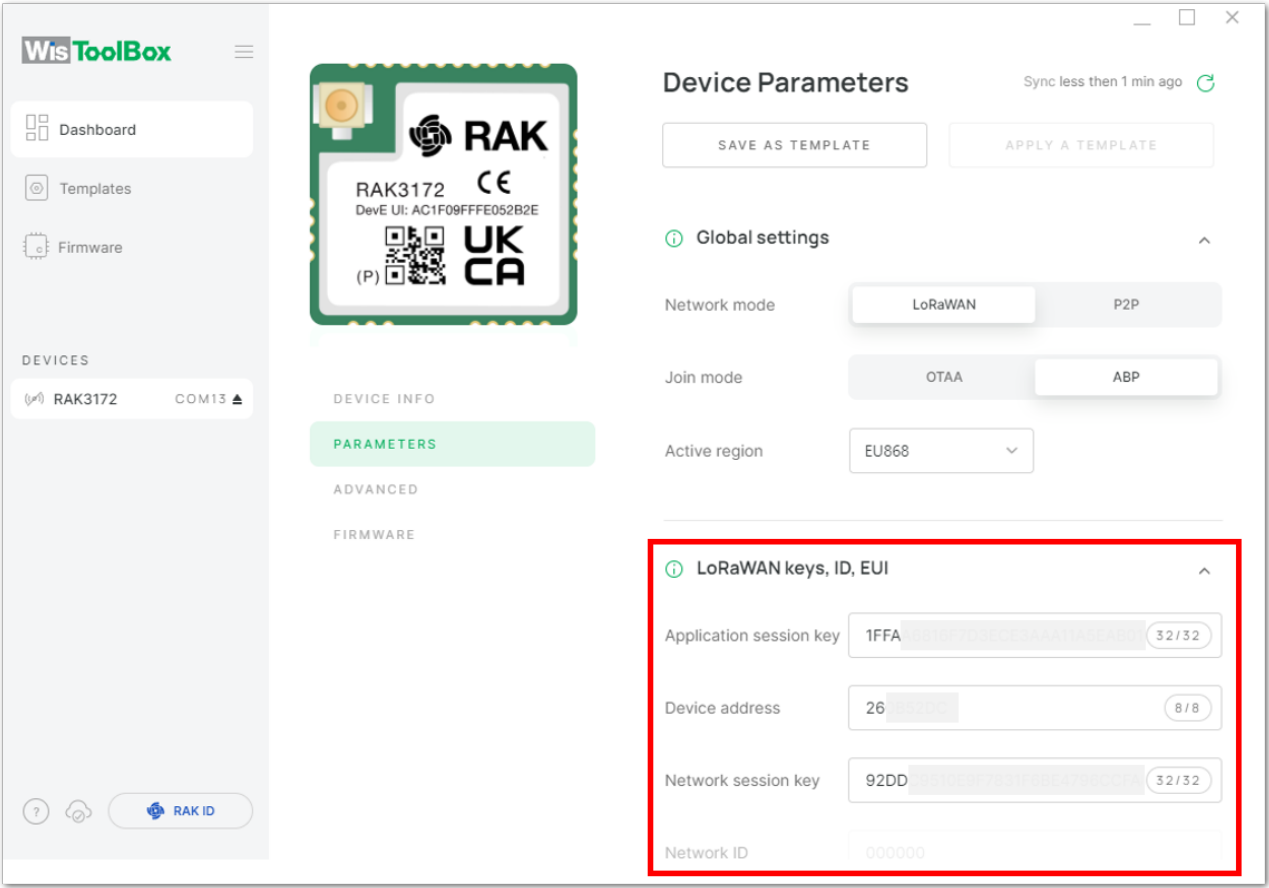


Figure 152: Successfully configured ABP device via WisToolBox dashboard

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

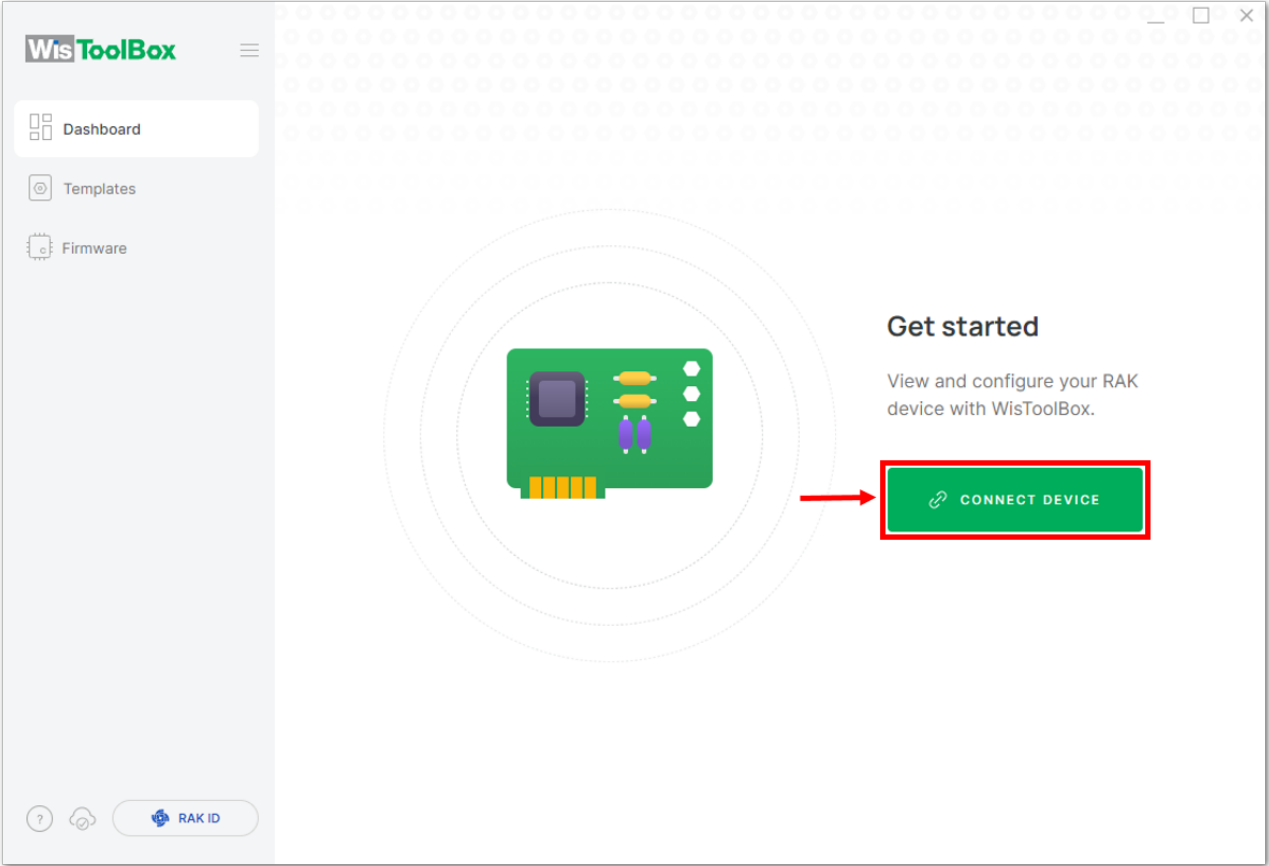


Figure 153: CONNECT DEVICE

3. Then select your target port where your **RAK3172** is connected. Once recognized, click **CONNECT** as shown in **Figure 155**.

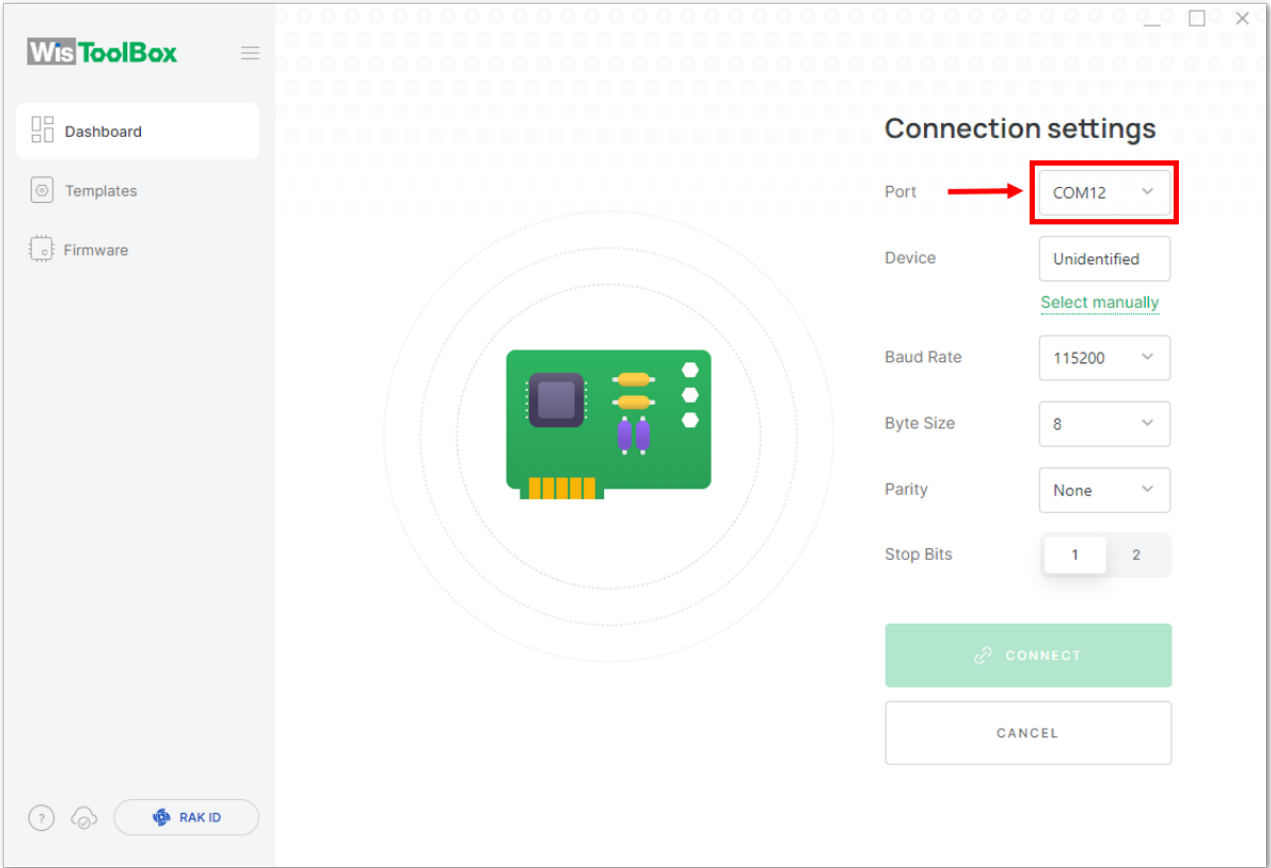


Figure 154: Setting up your device

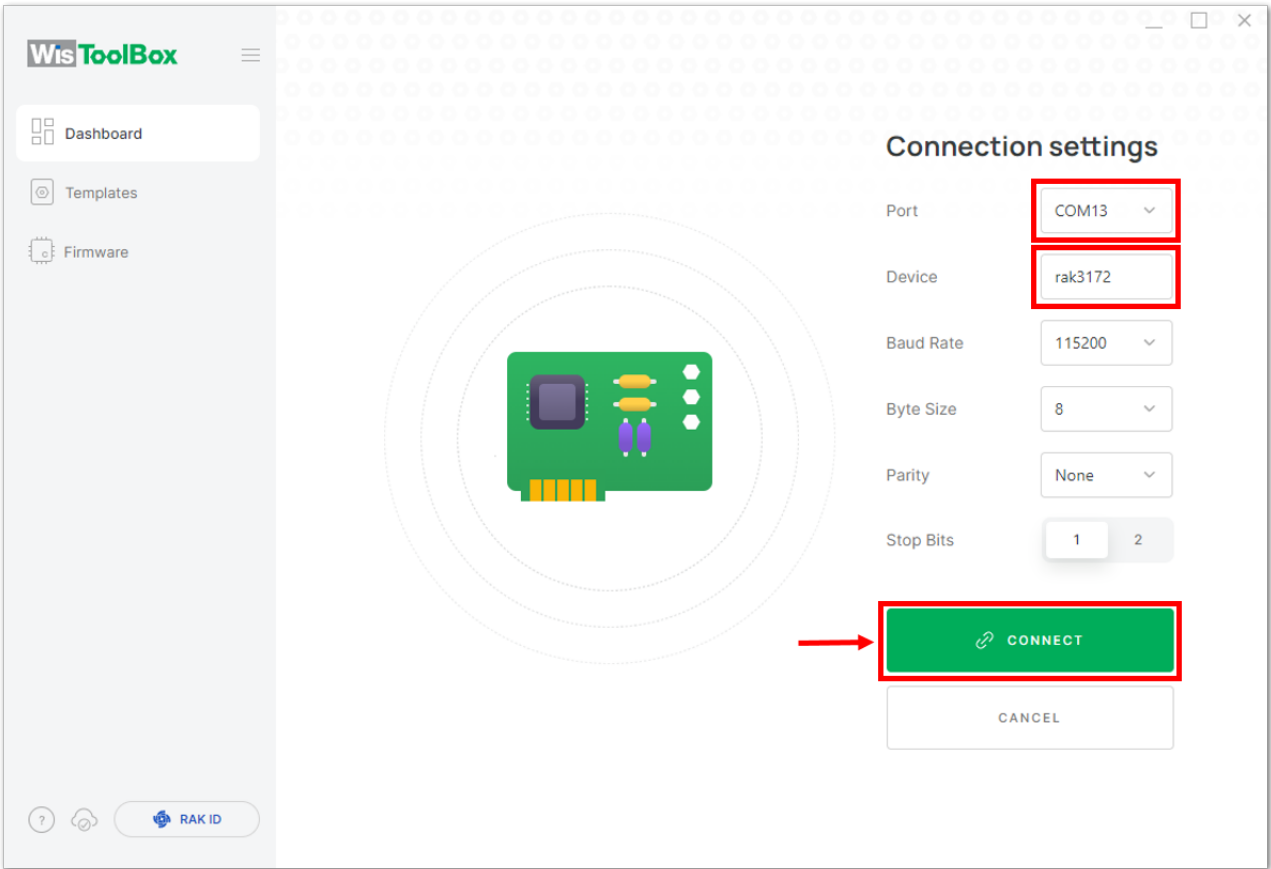


Figure 155: Setting up your device

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

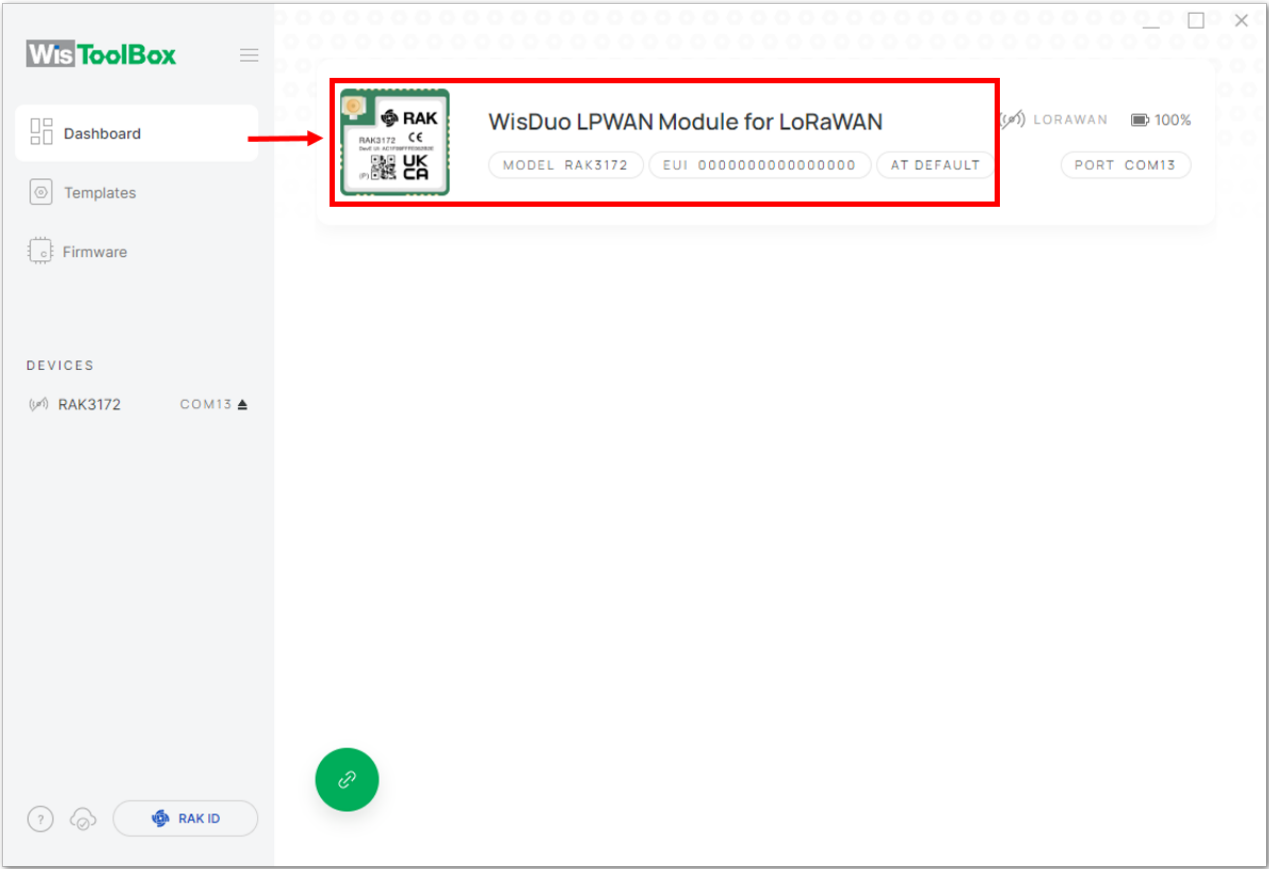


Figure 156: Device seen from WisToolBox dashboard

5. Then click **ADVANCED**.

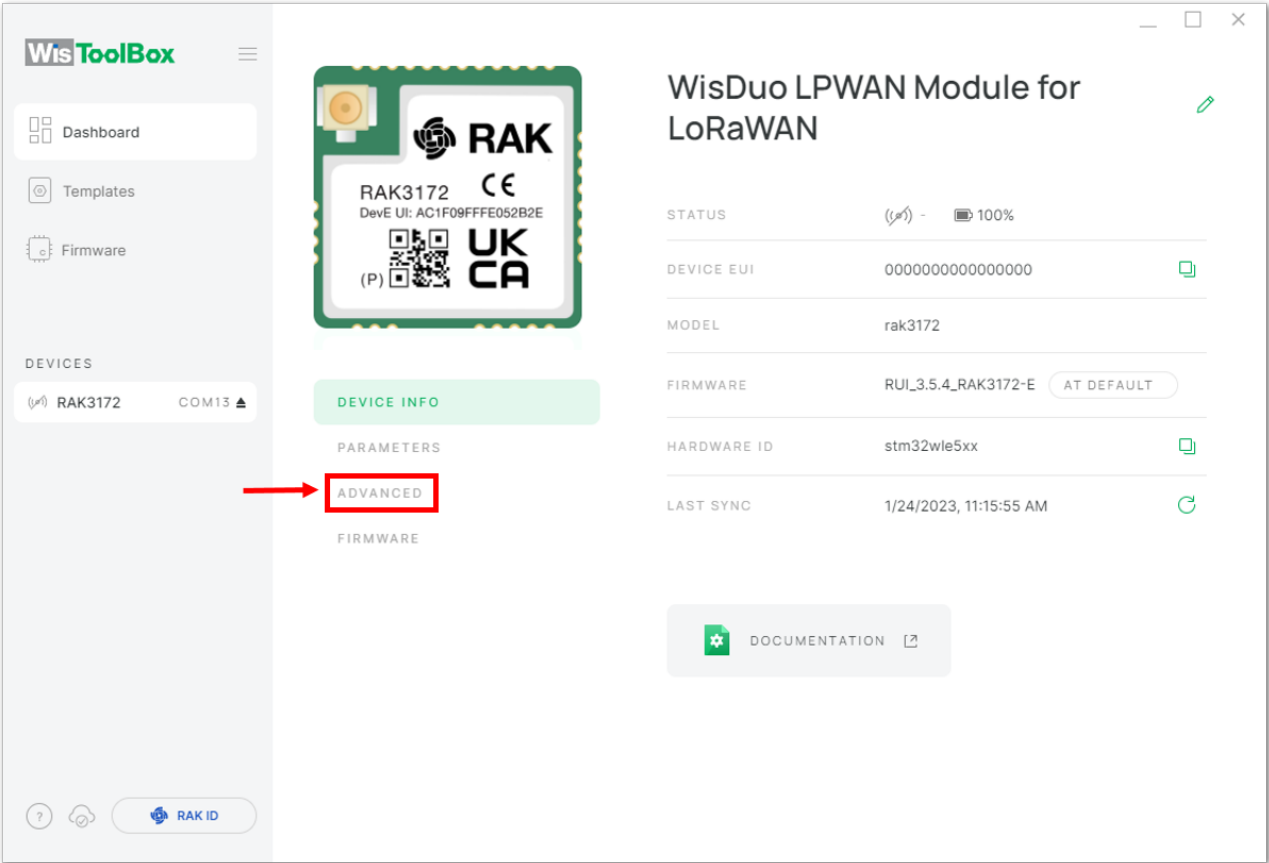


Figure 157: Setting up your device

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

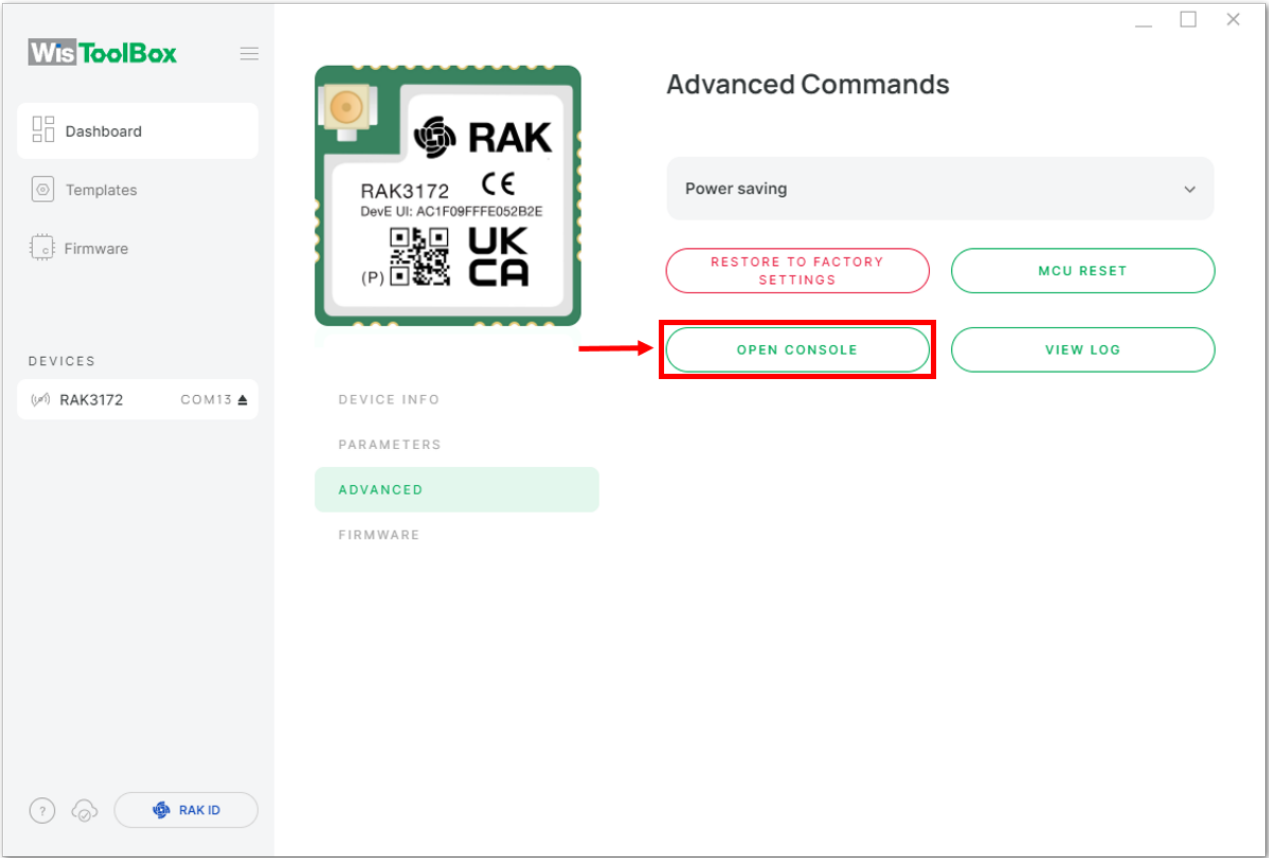


Figure 158: OPEN CONSOLE

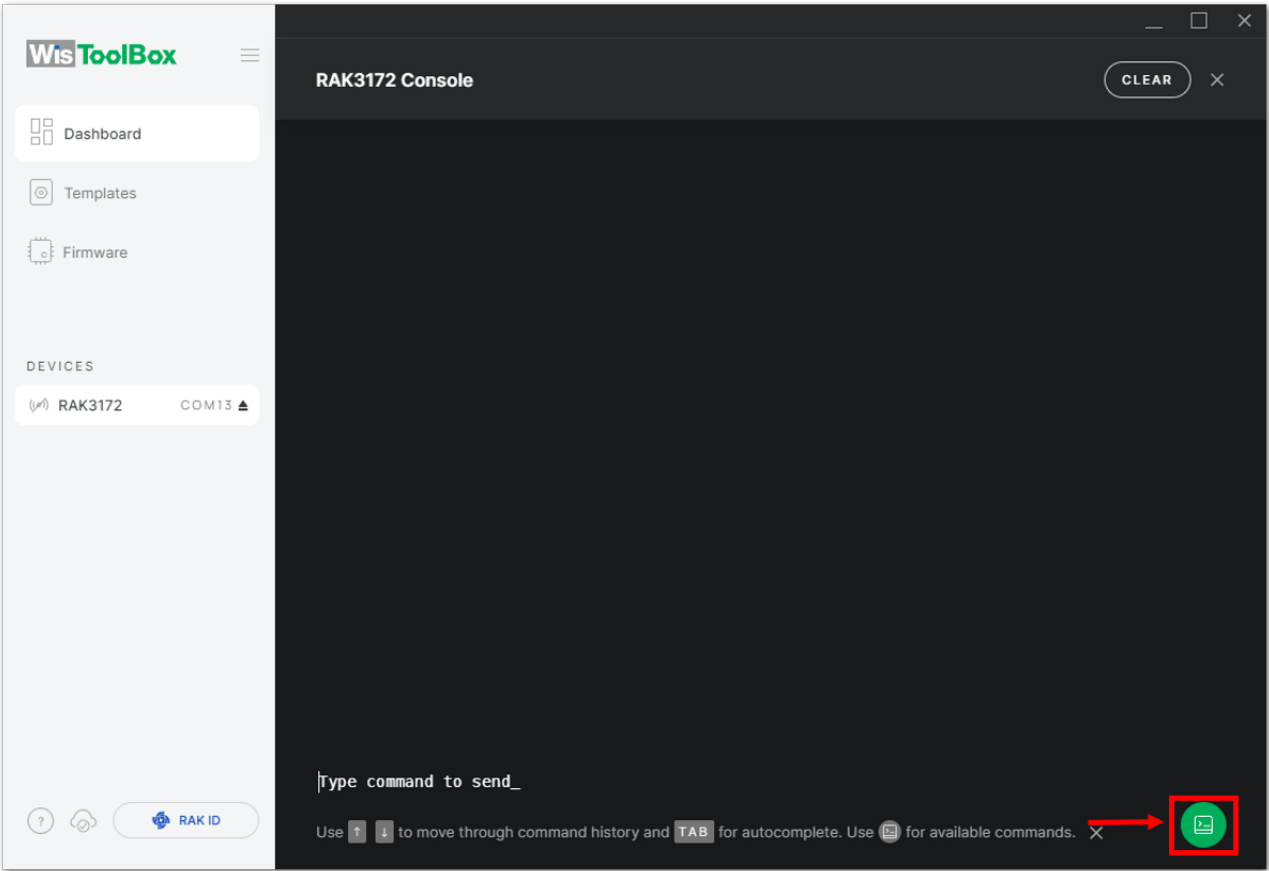


Figure 159: Opening the Console terminal of WisToolBox

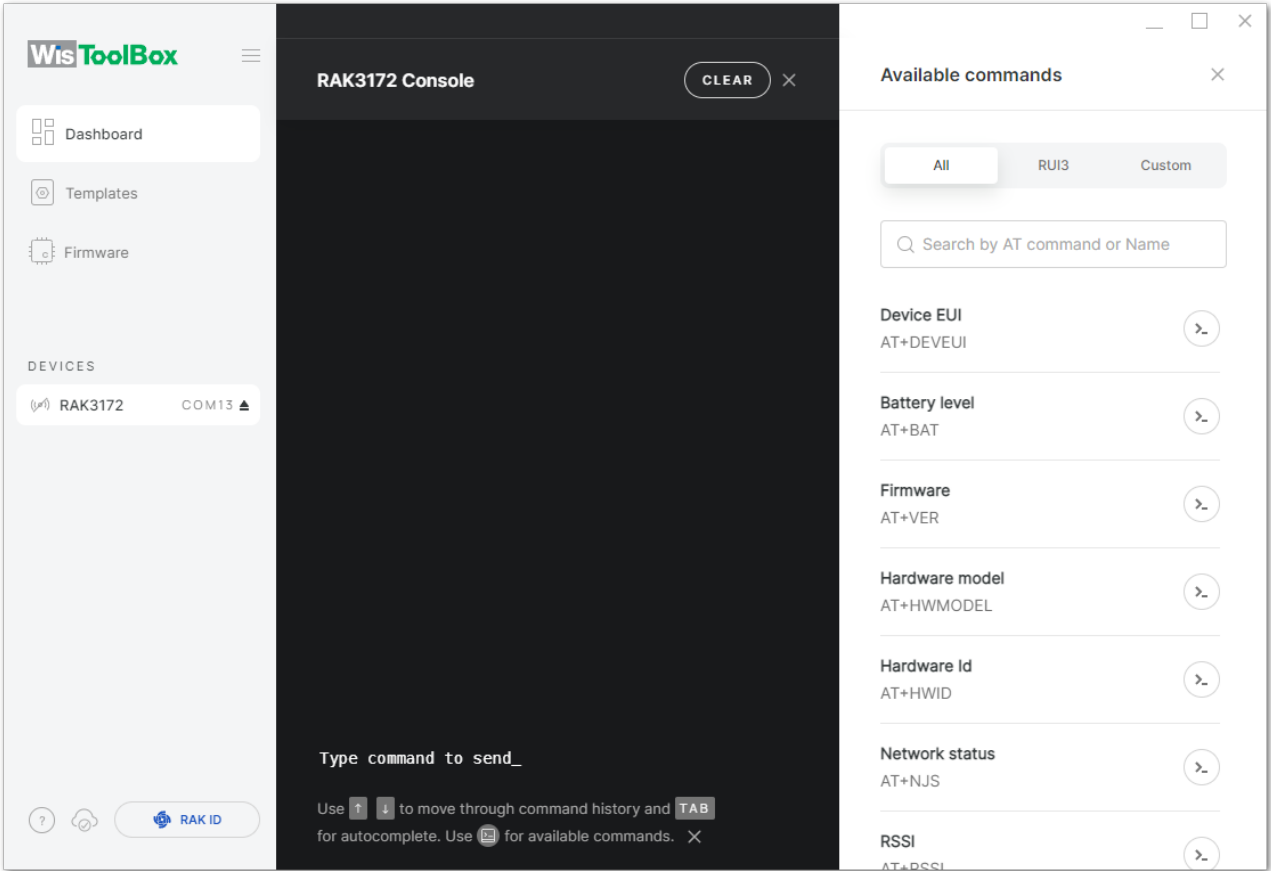


Figure 160: Opening the Console terminal of WisToolBox

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.

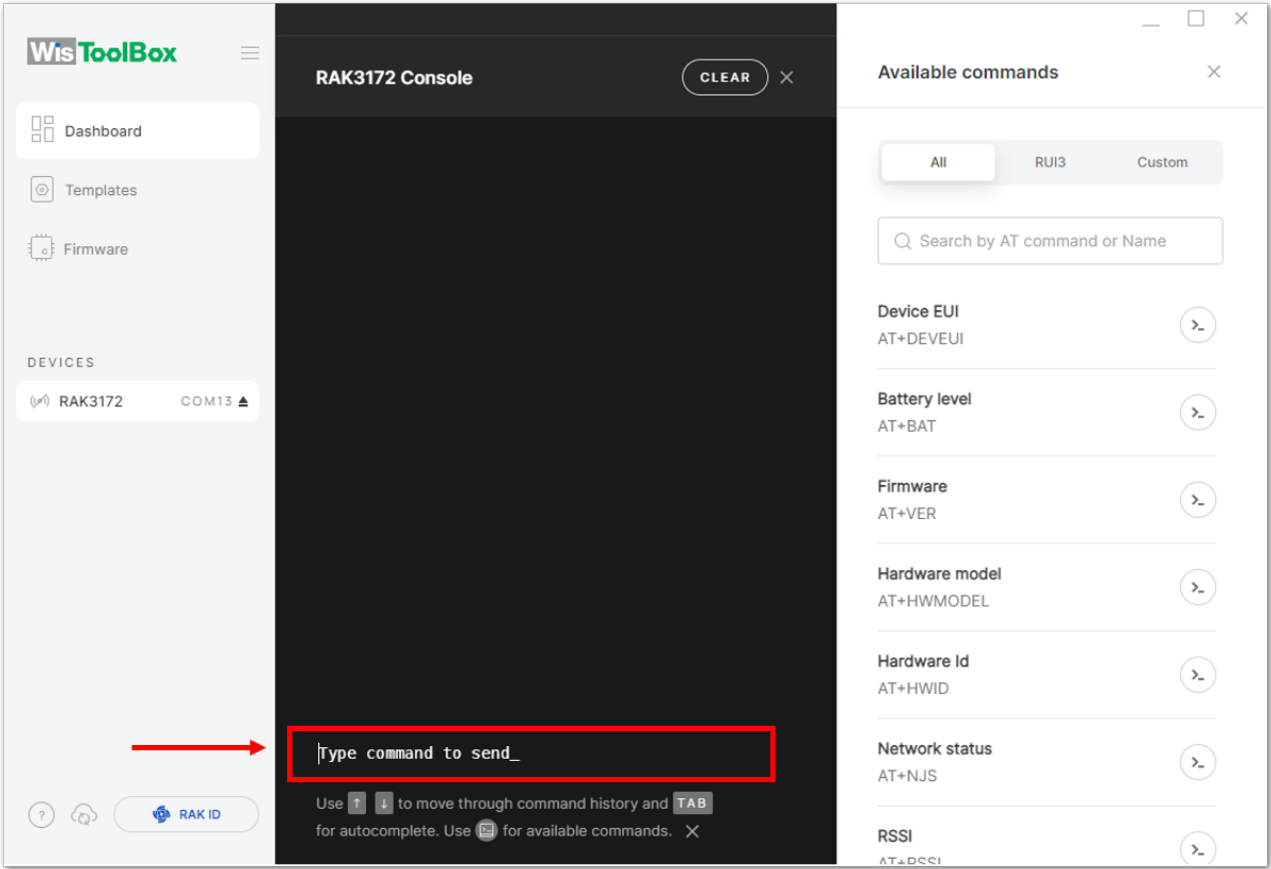


Figure 161: Setting up your Console

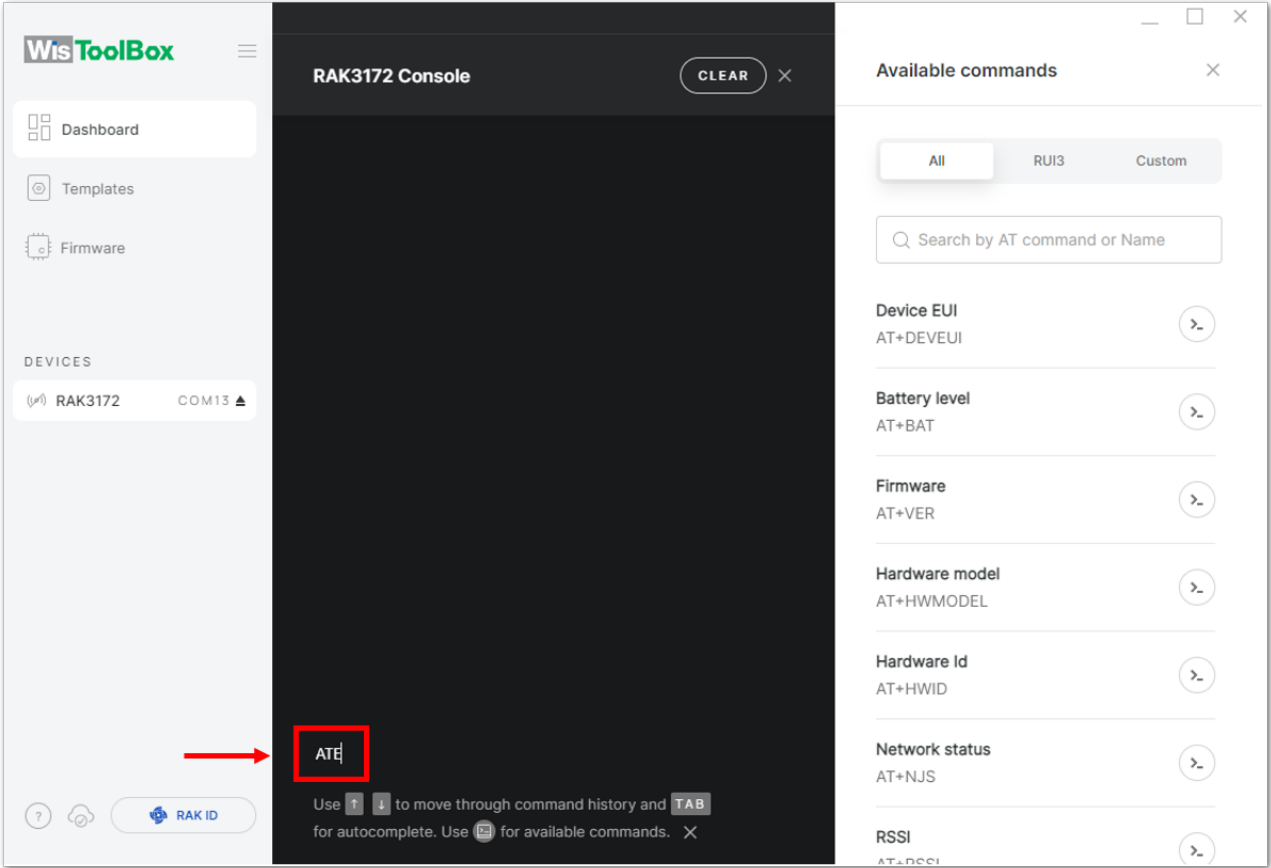


Figure 162: Setting up your Console

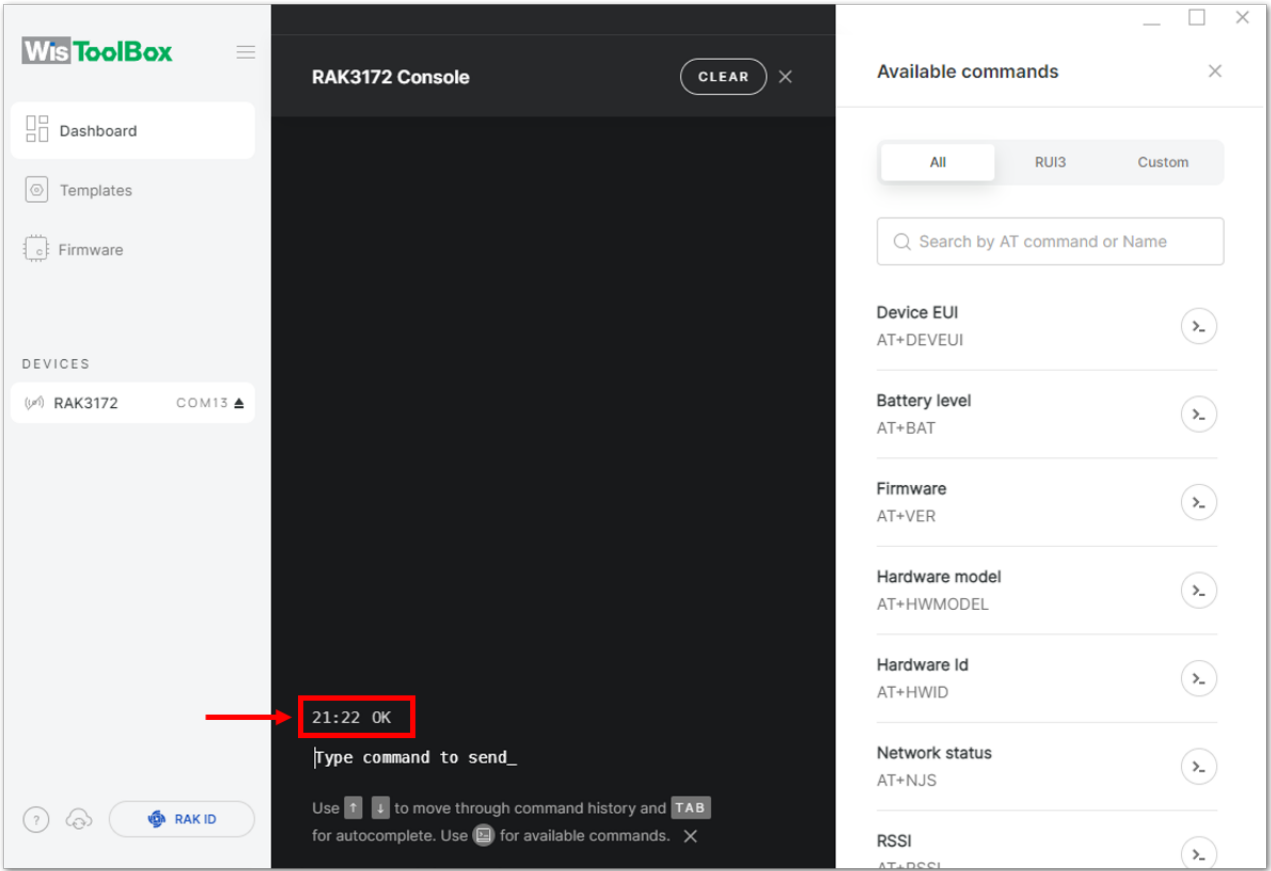


Figure 163: Setting up your Console

8. Then configure the LoRaWAN join mode to **ABP**. You can check what parameter you will input by typing **AT+NJM?** then **Enter** into the console terminal. For **ABP**, you should input **AT+NJM=0** then press **Enter** as shown in Figure 166.

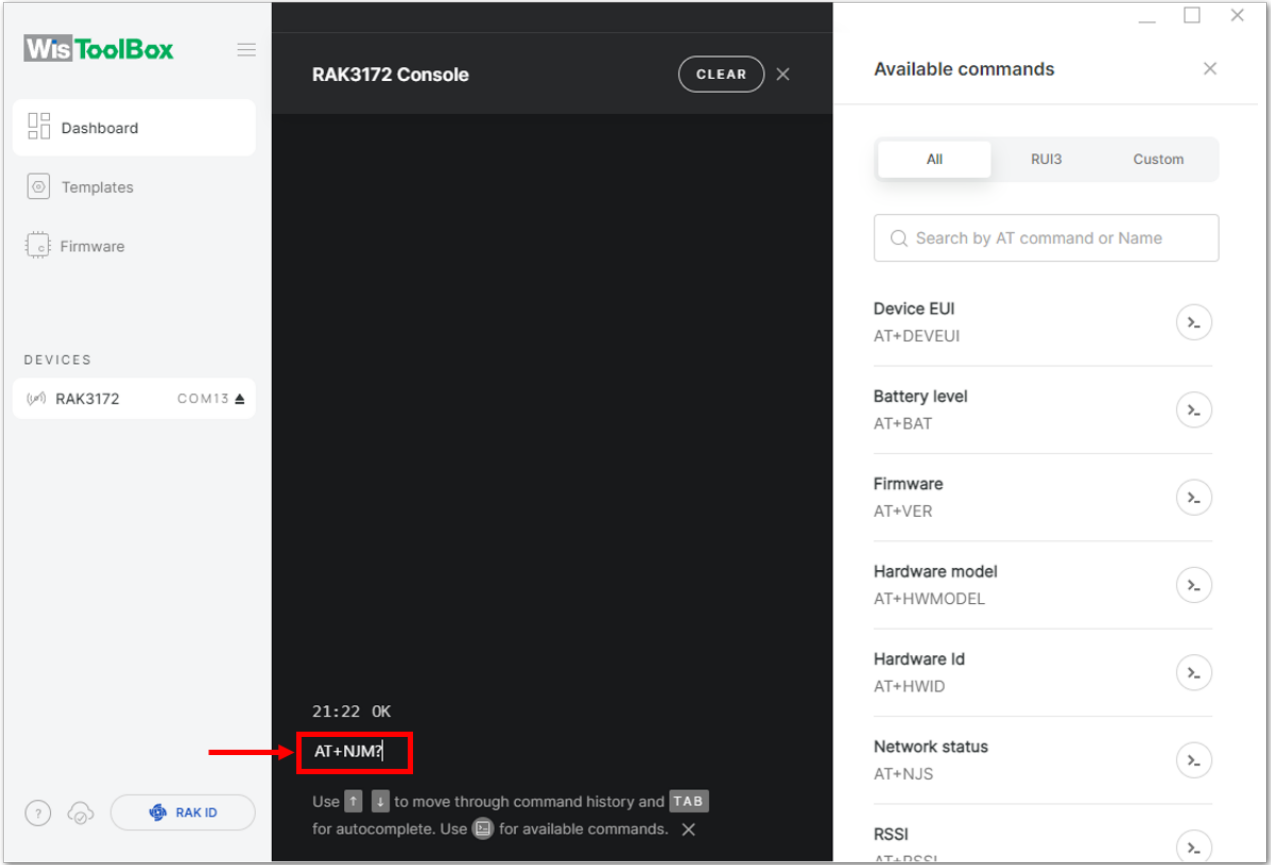


Figure 164: Setting up your Console

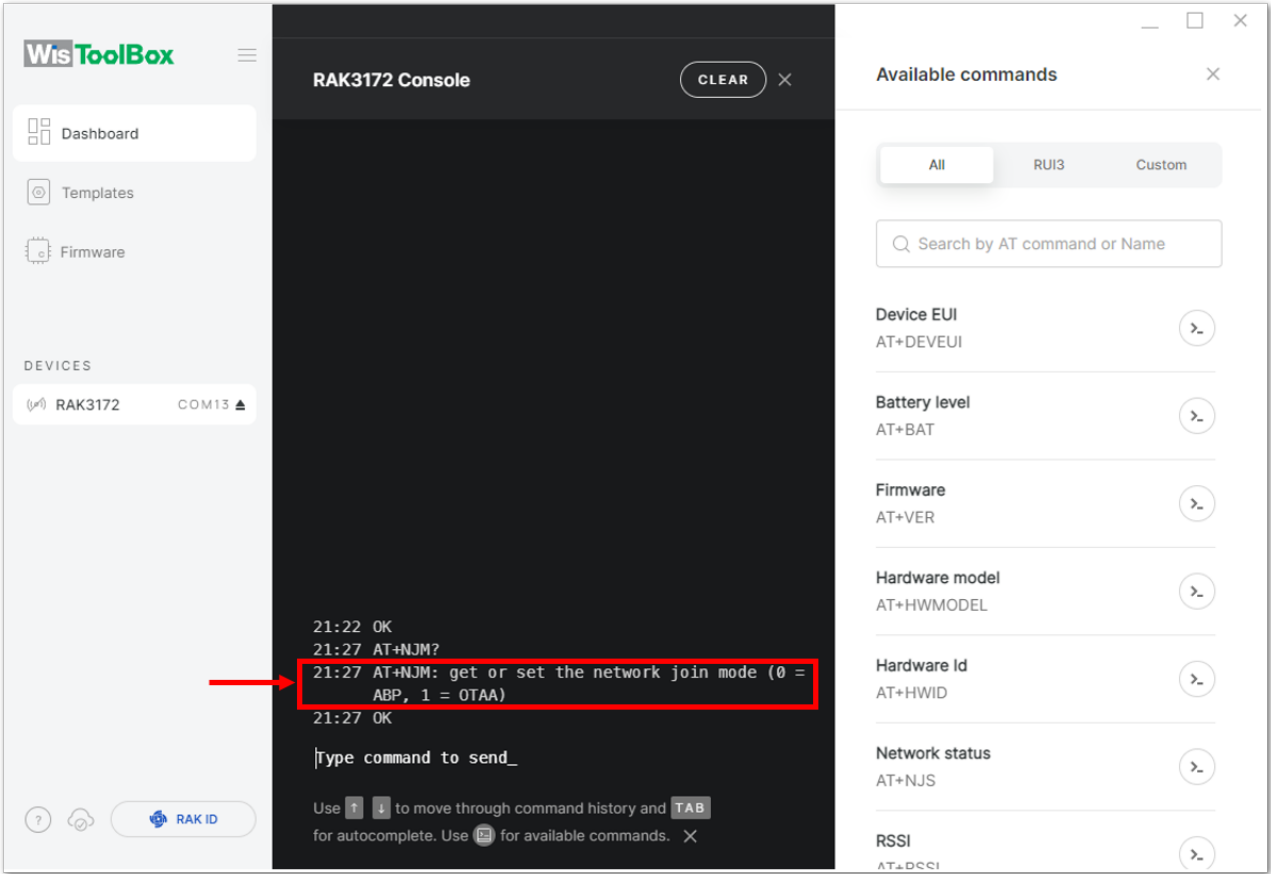


Figure 165: Setting up your Console

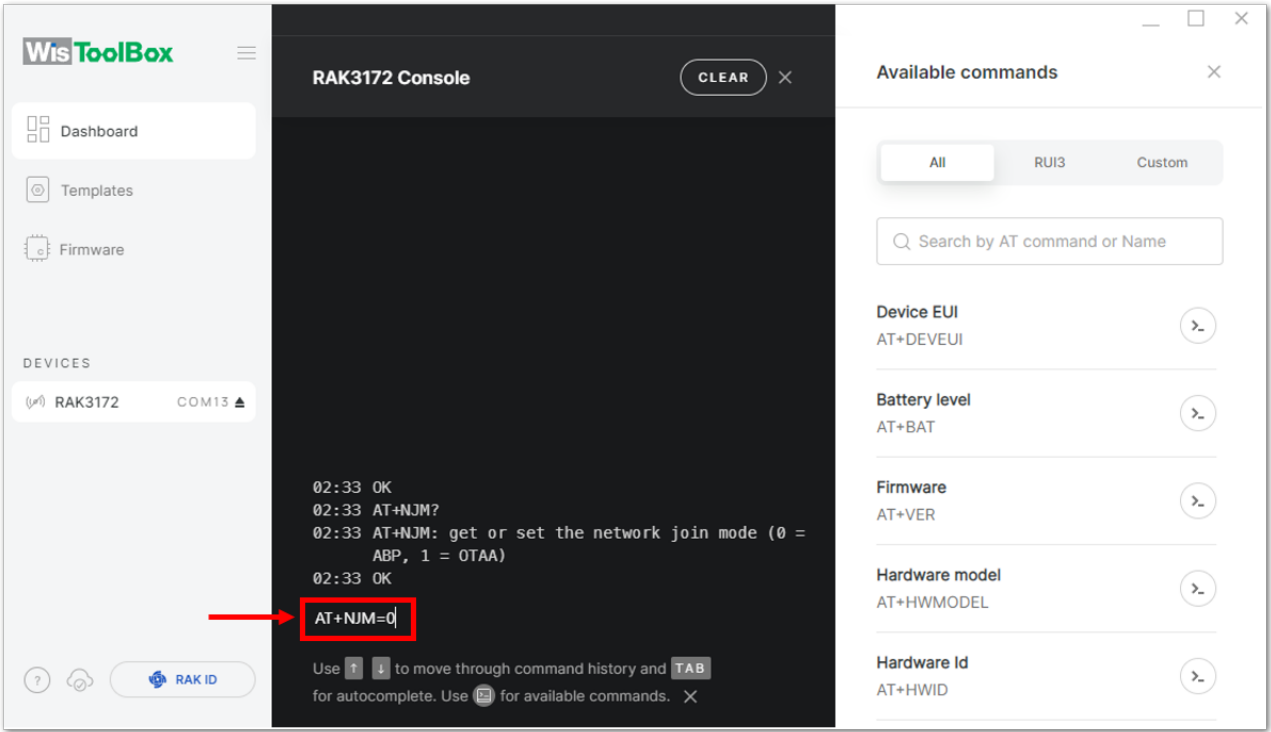


Figure 166: Setting up your Console

9. Once done, set-up your LoRaWAN region to EU868. You can check what parameter you will input by typing **AT+BAND?** 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 band, 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

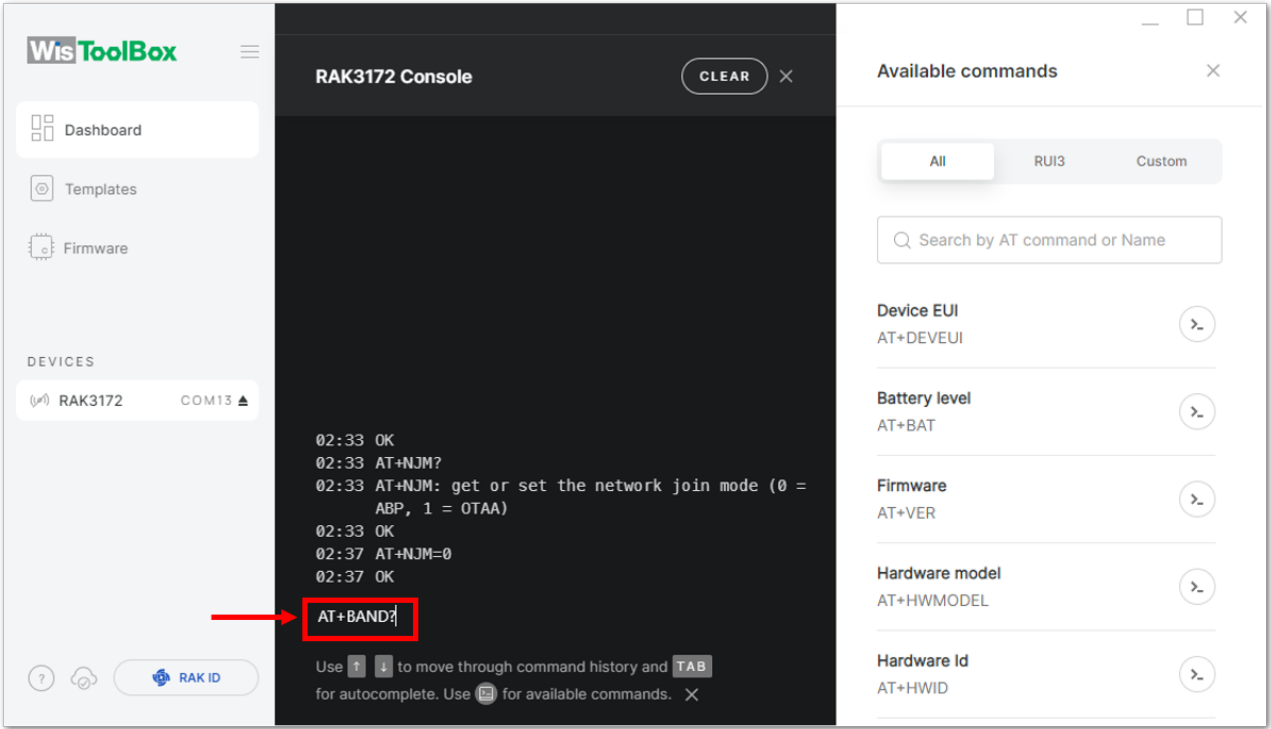


Figure 167: Setting up your Console

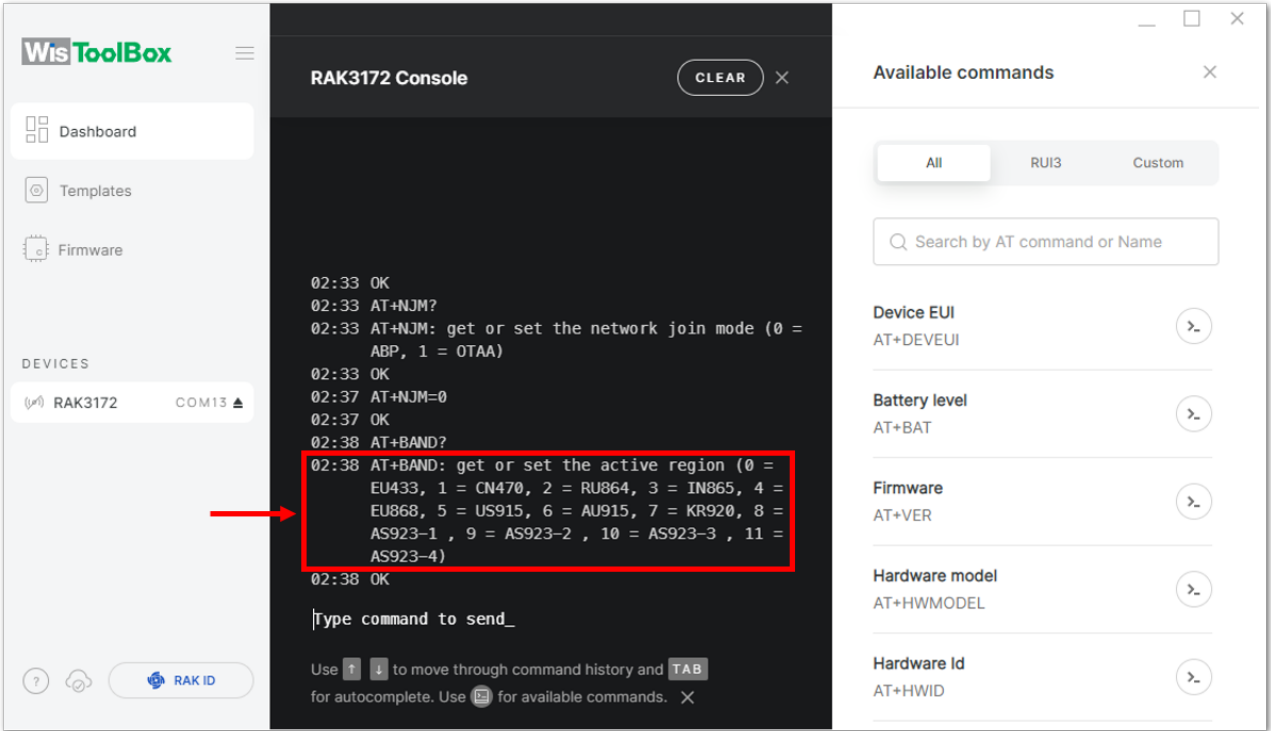


Figure 168: Setting up your Console

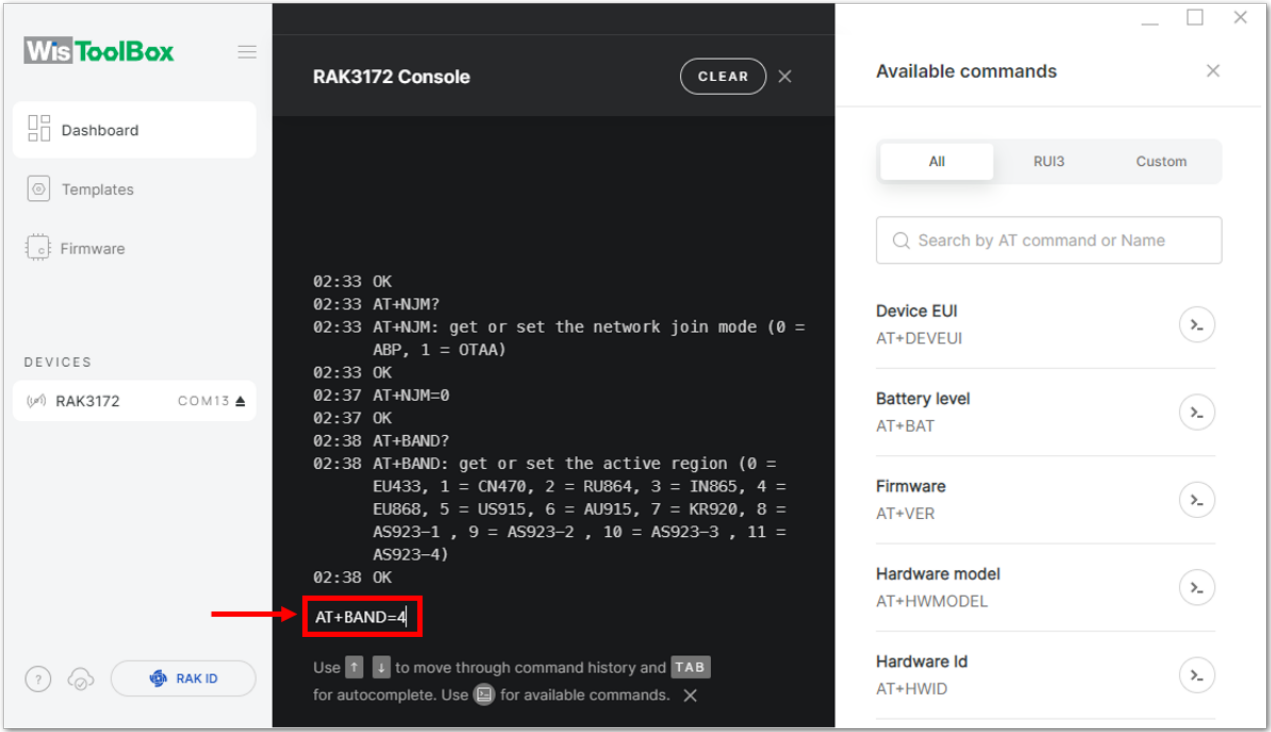


Figure 169: Setting up your Console

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**.

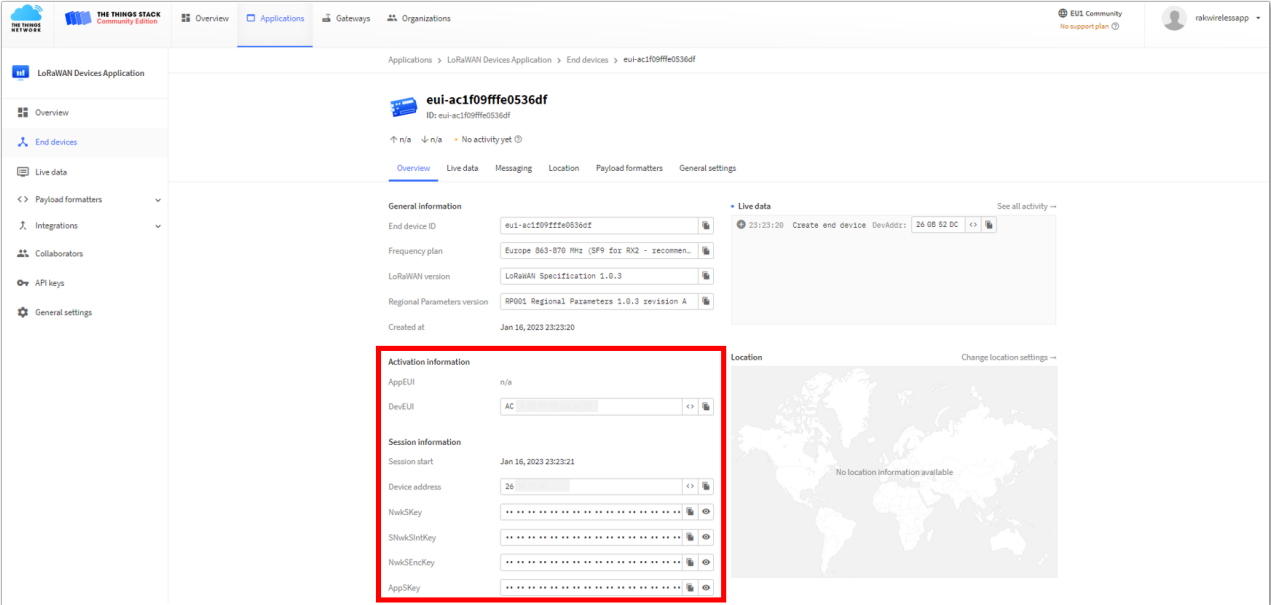


Figure 170: Your created ABP device from your TTN console

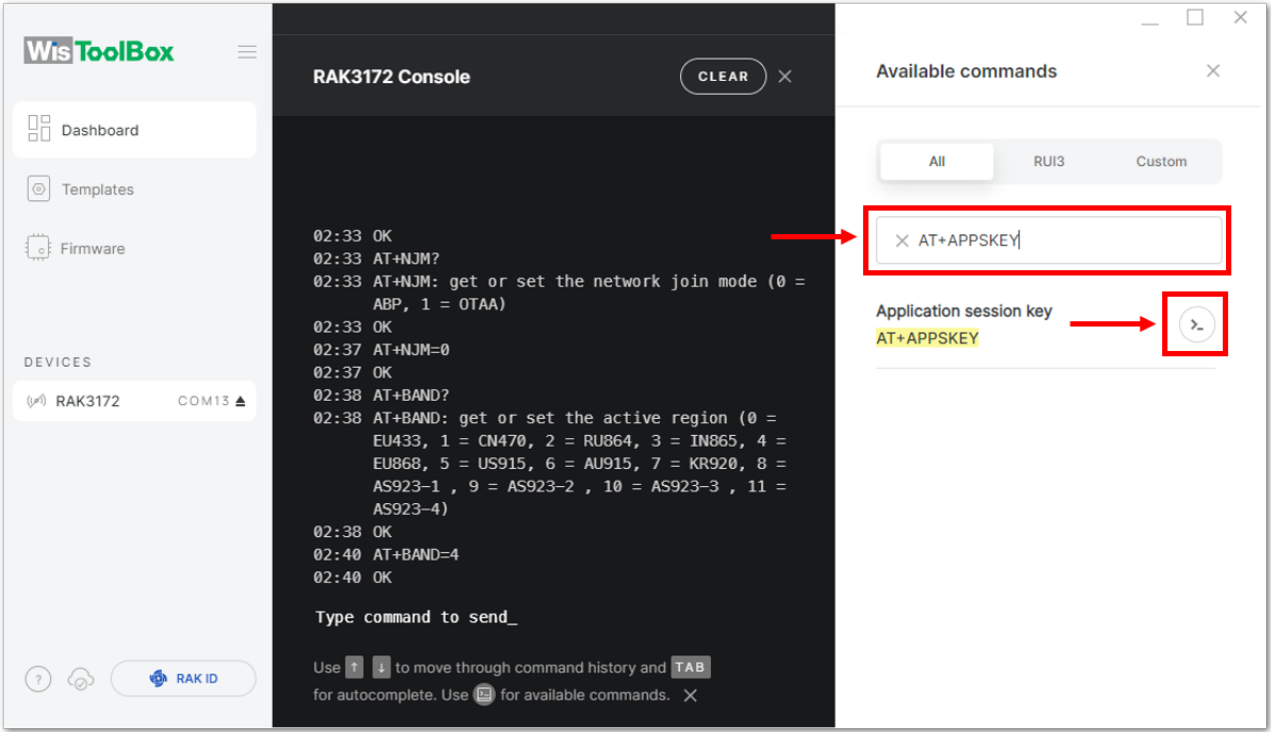


Figure 171: Setting up your Console

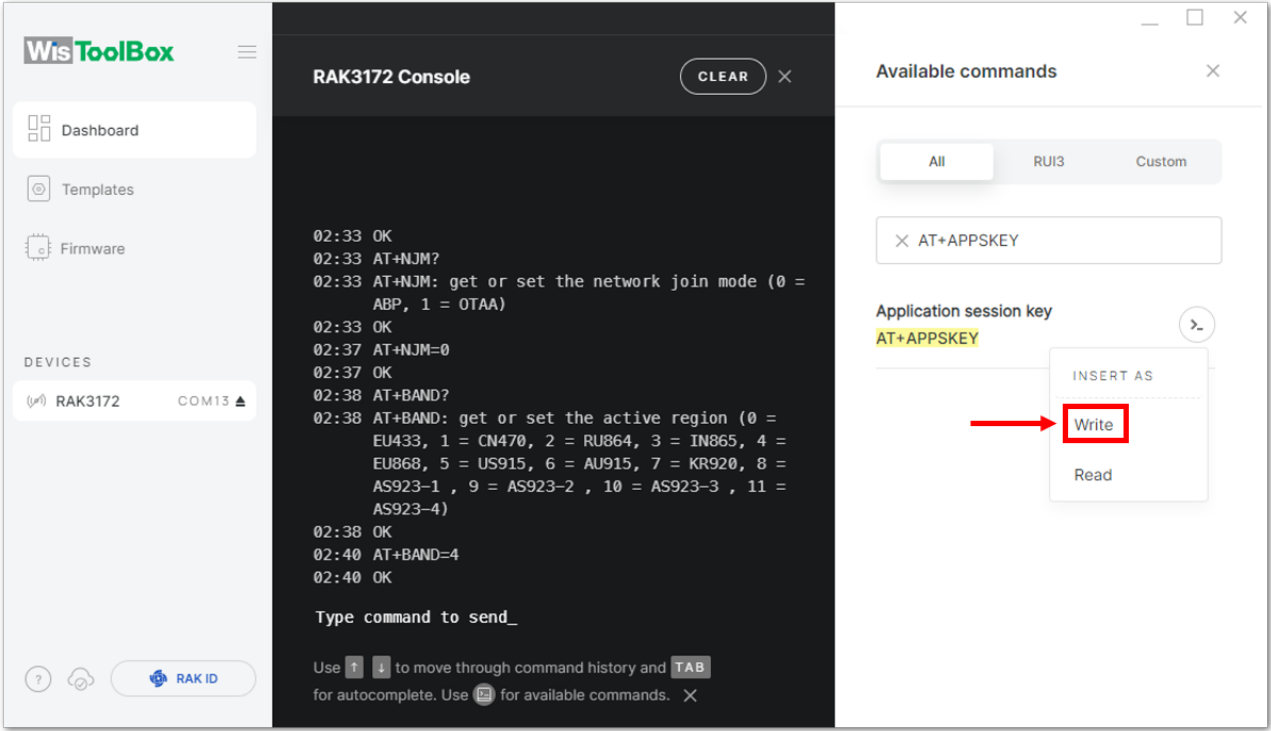


Figure 172: Setting up your Console

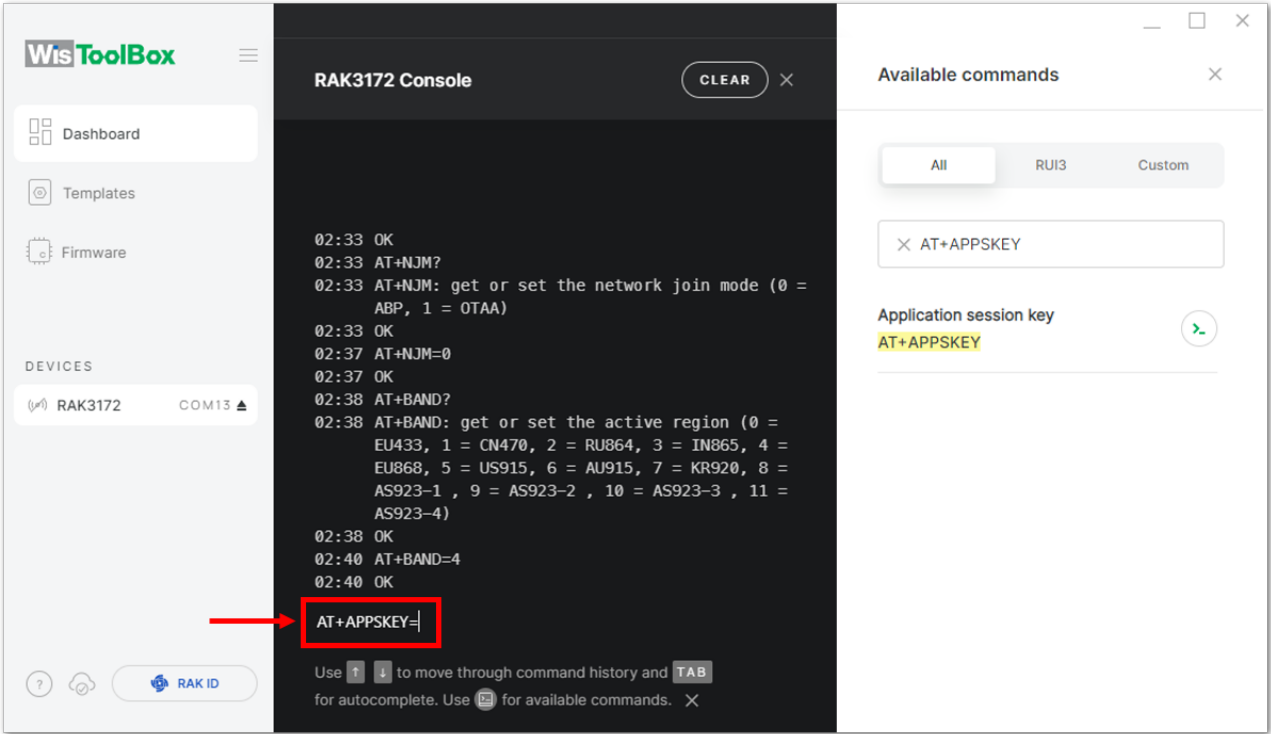


Figure 173: Setting up your Console

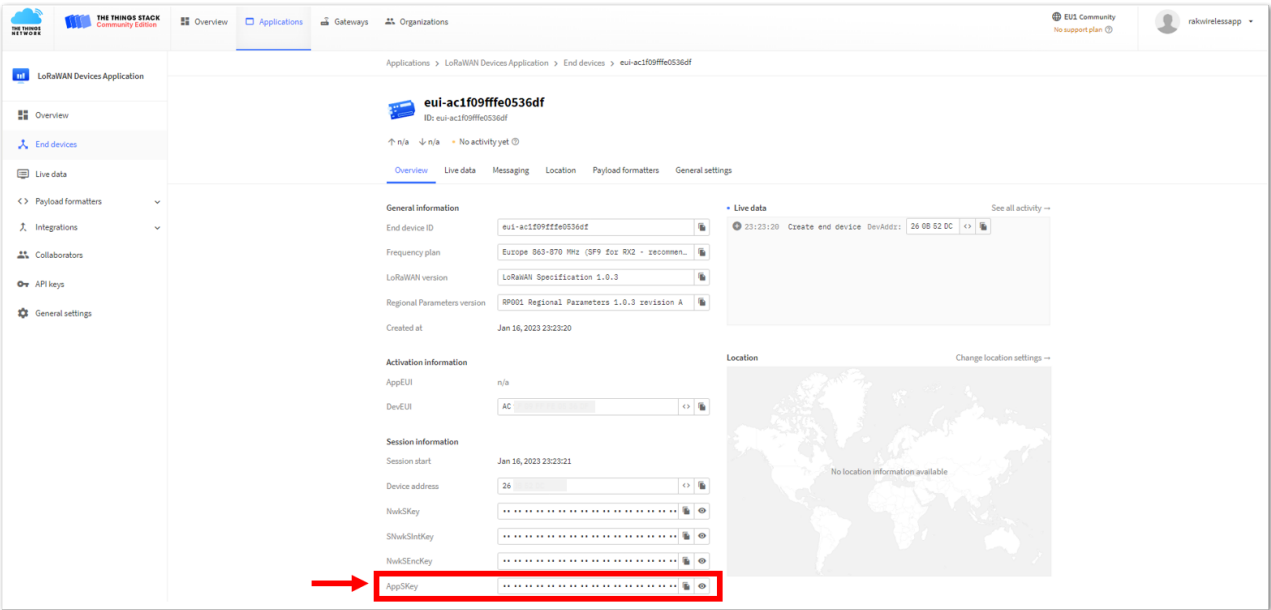


Figure 174: Copying the AppKey credential from TTN to WisToolBox

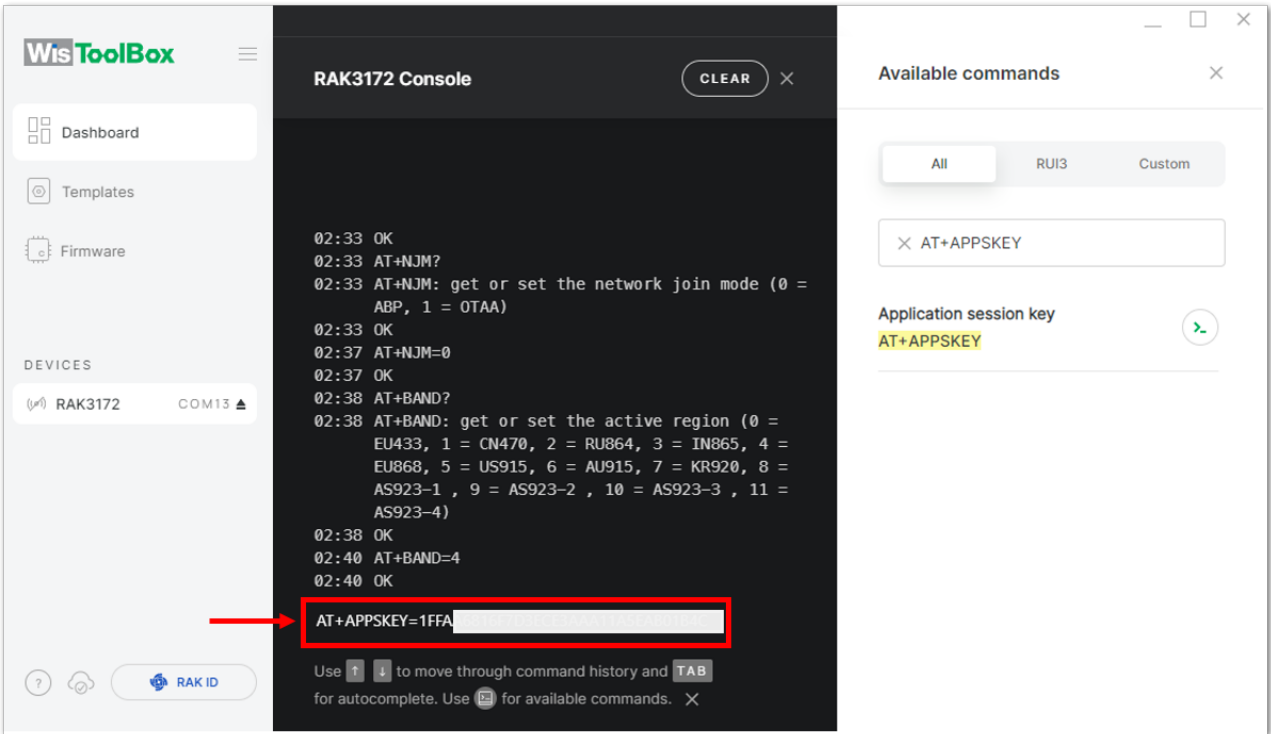


Figure 175: Setting up your Console

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

- For Device address

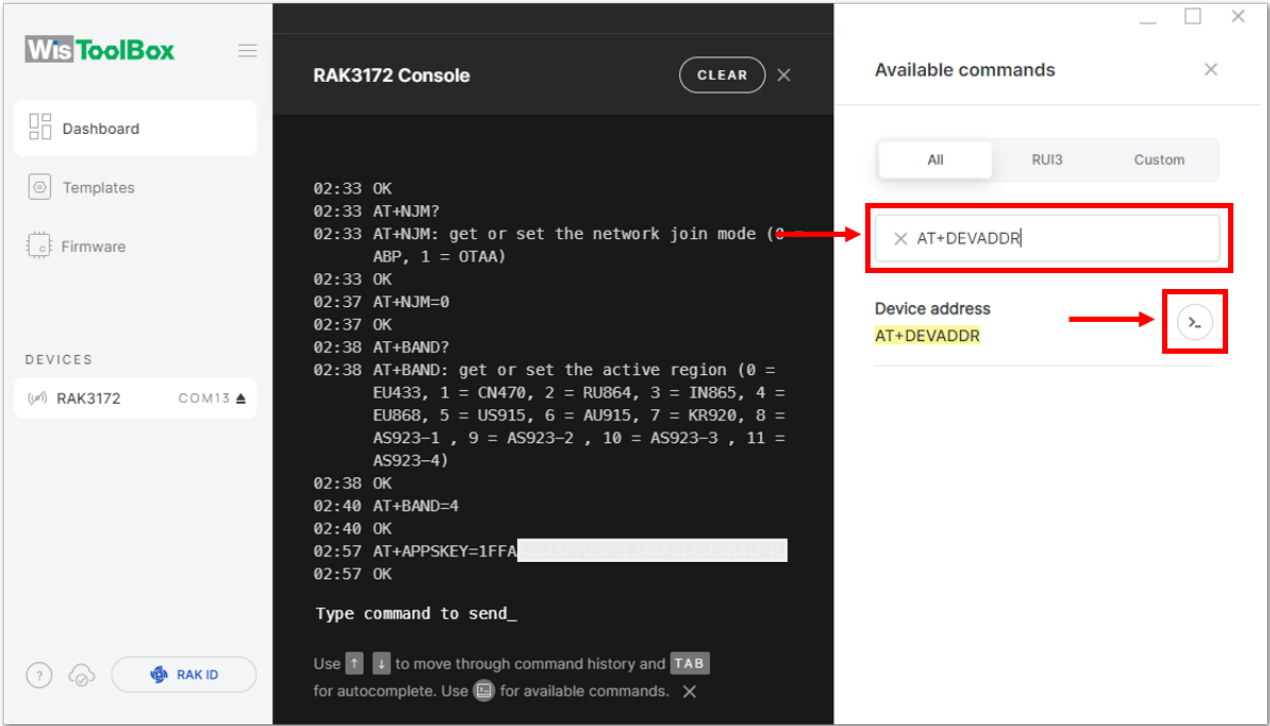


Figure 176: Setting up your Console

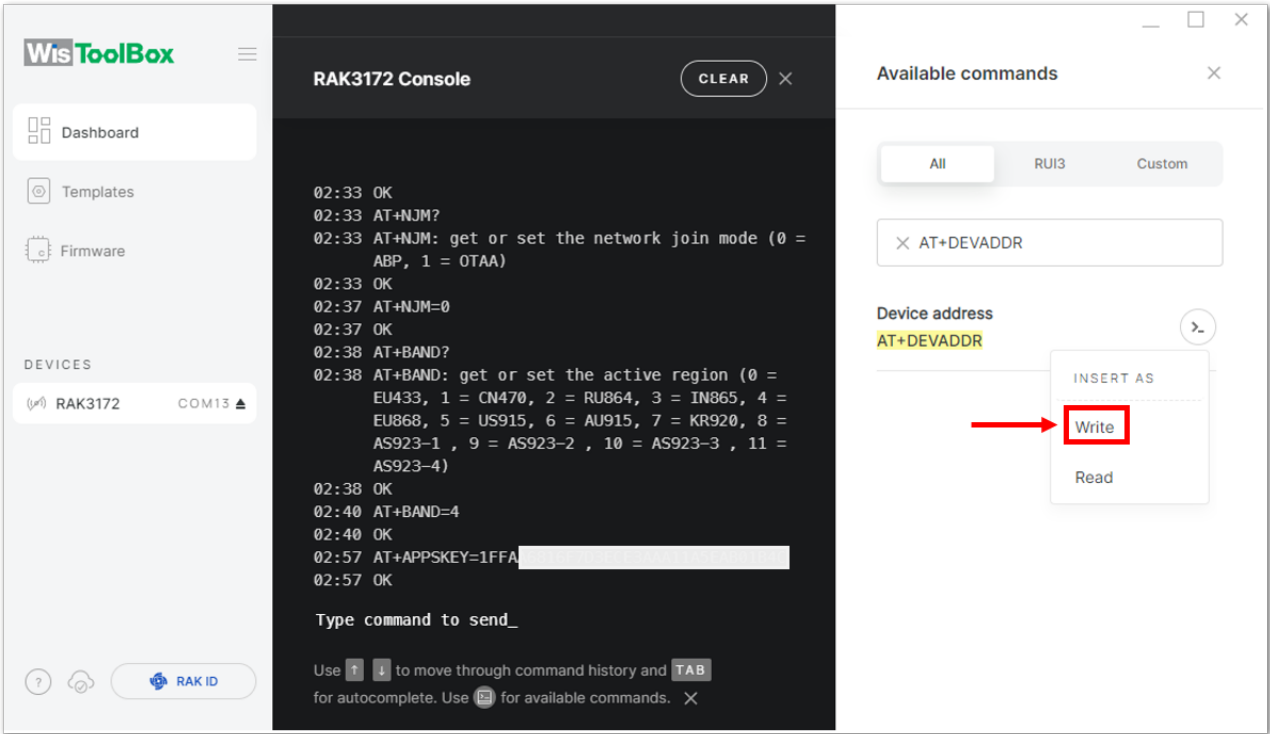


Figure 177: Setting up your Console

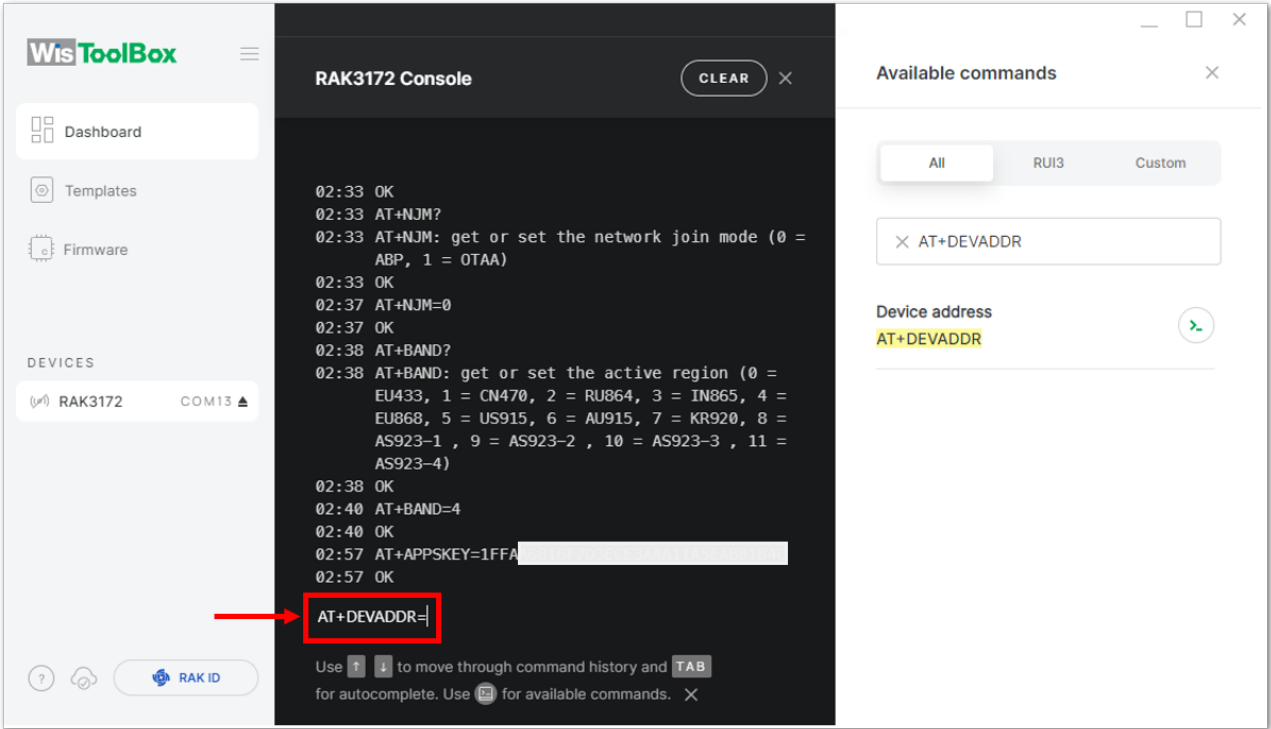


Figure 178: Setting up your Console

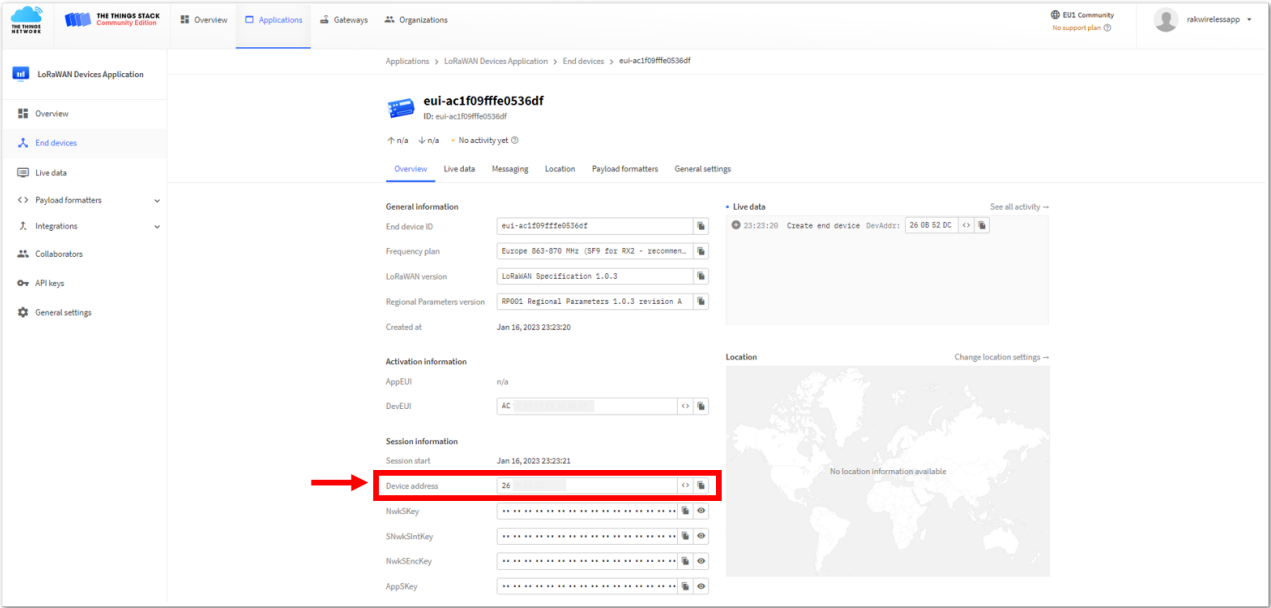


Figure 179: Copying the Device address credential from TTN to WisToolBox

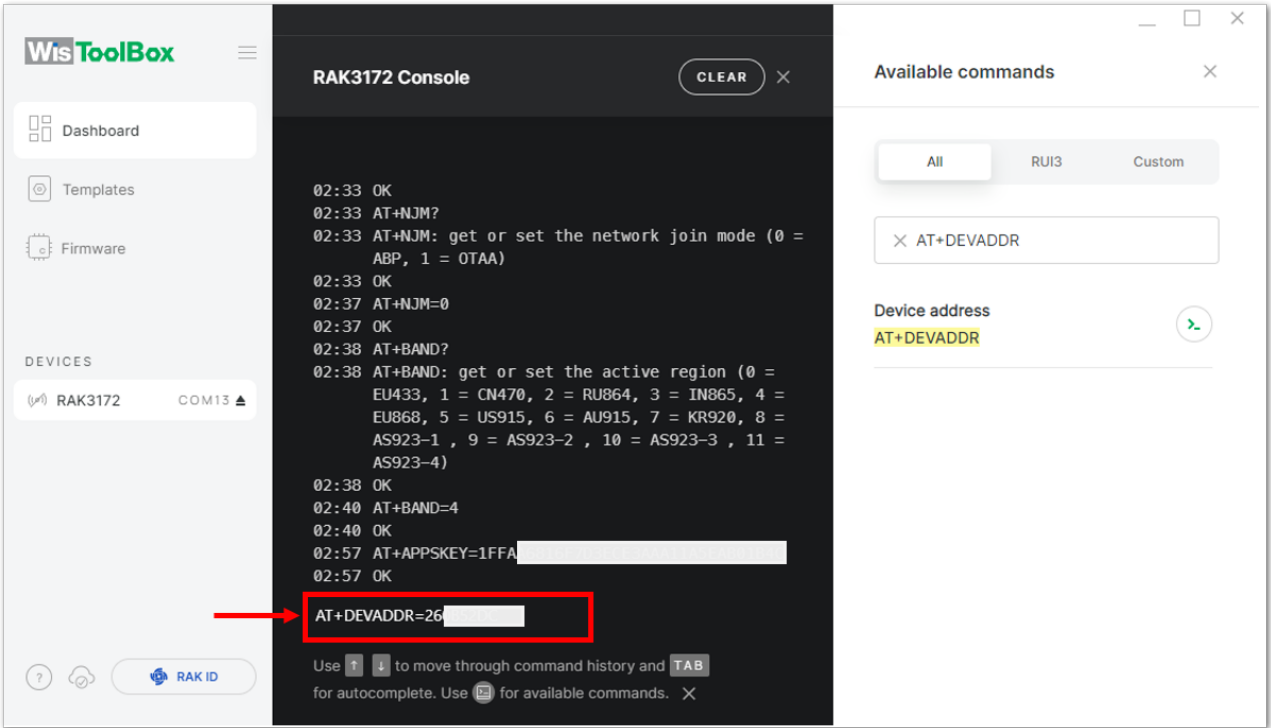


Figure 180: Setting up your Console

- For Network session key (NwkSKey)

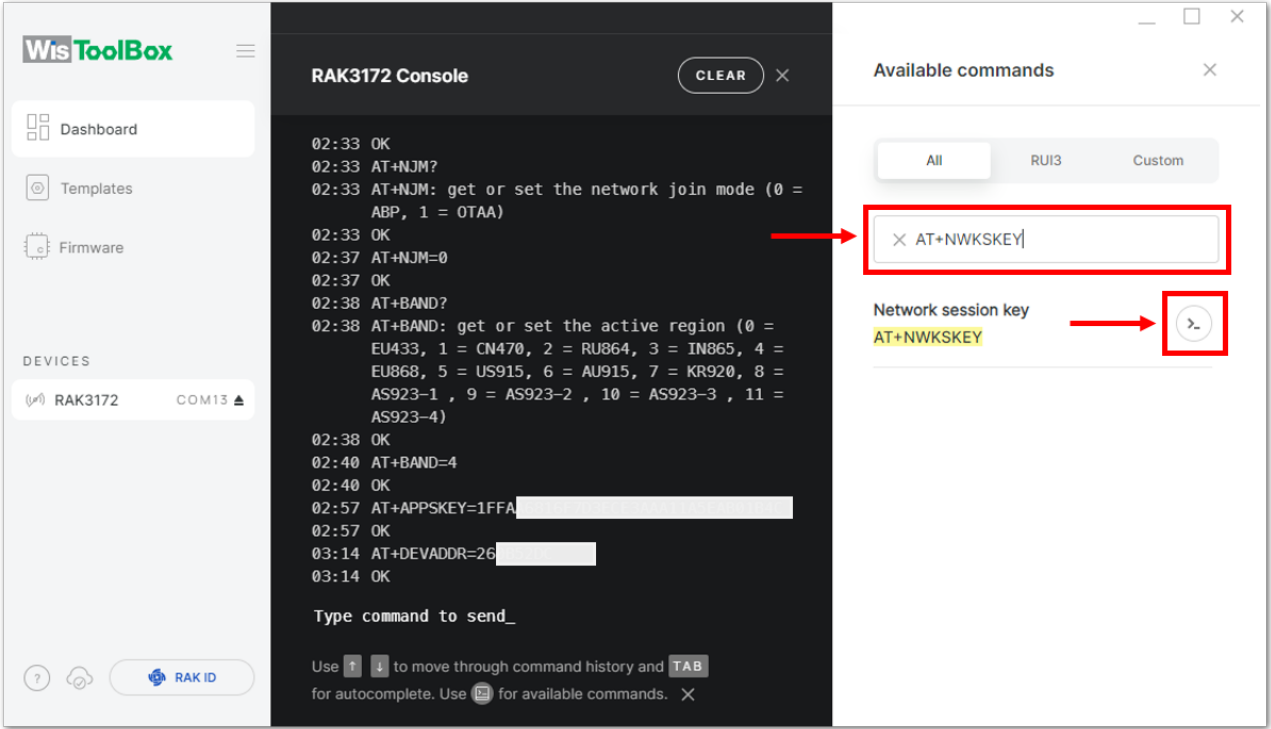


Figure 181: Setting up your Console

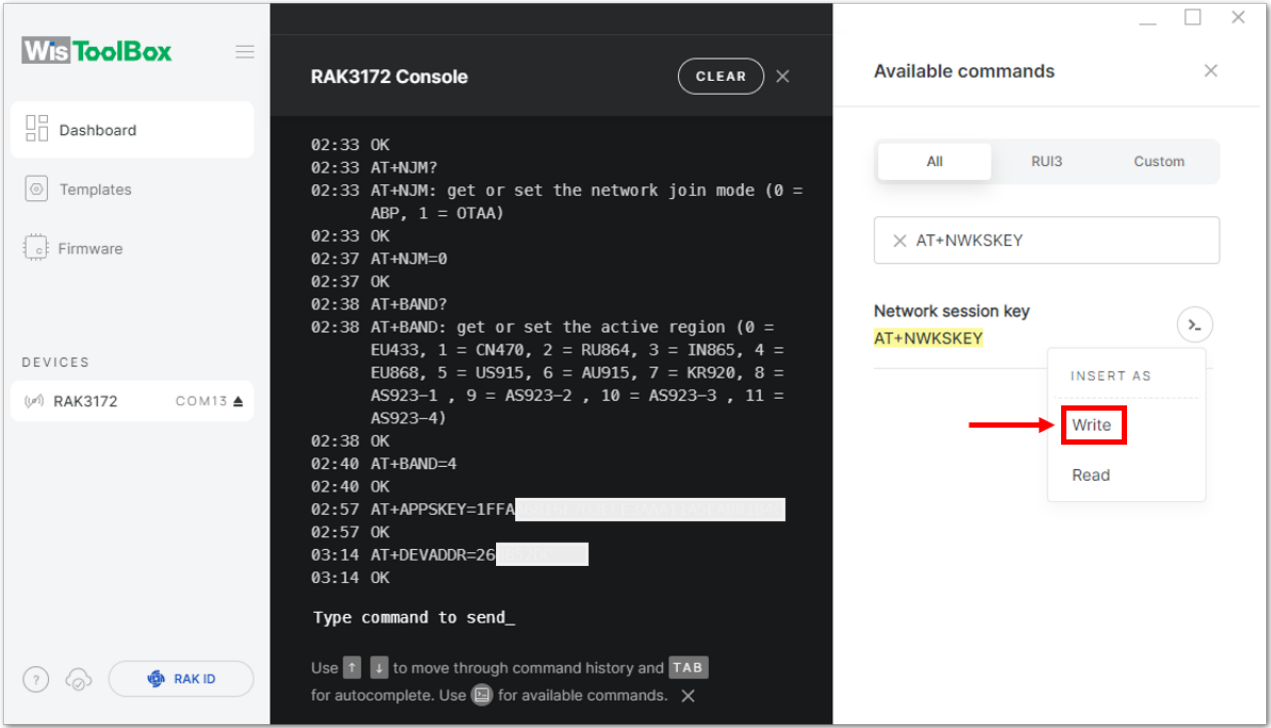


Figure 182: Setting up your Console

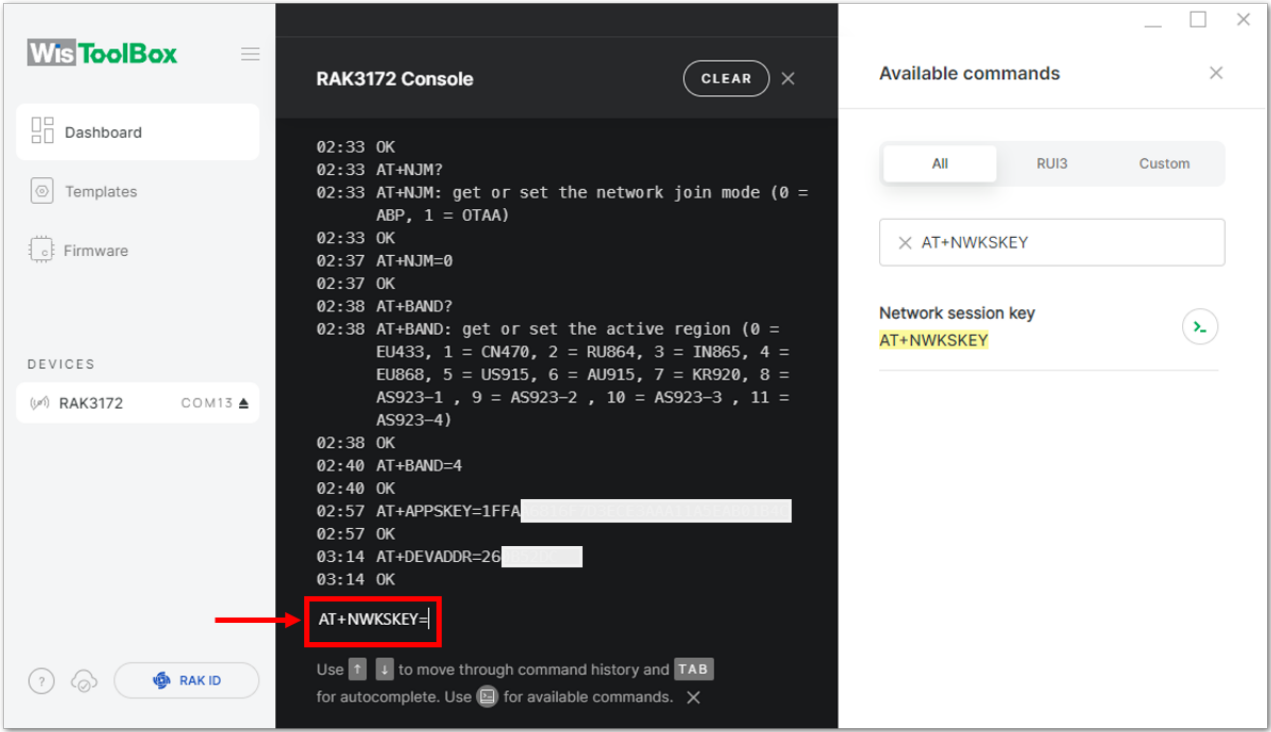


Figure 183: Setting up your Console

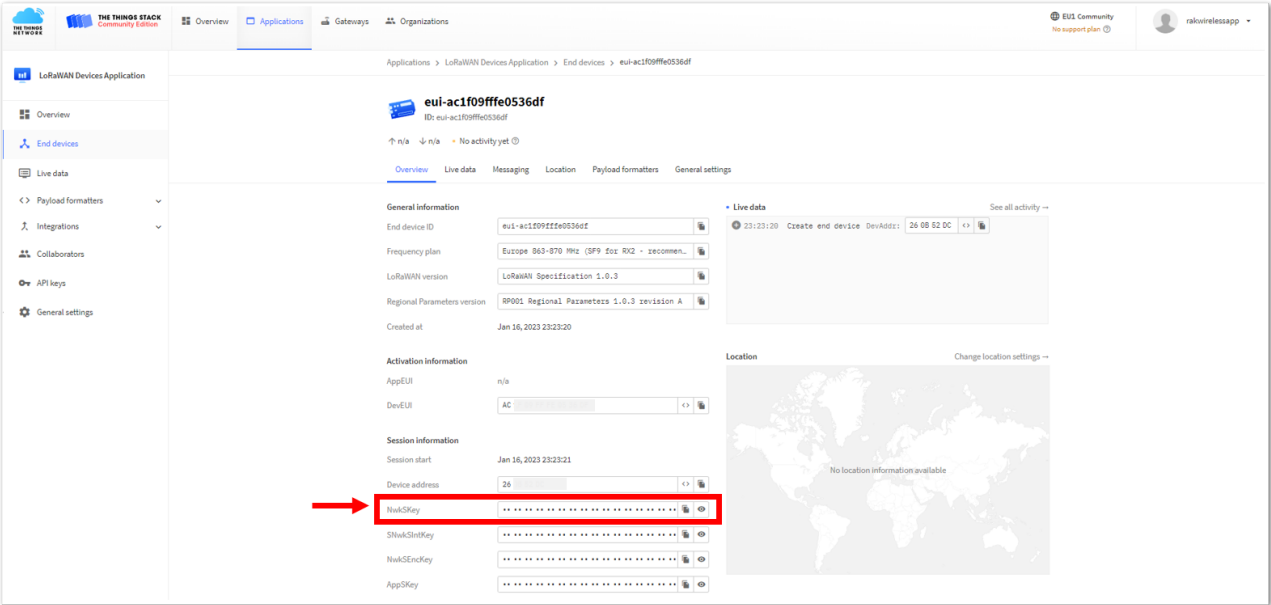


Figure 184: Copying the NwkSKey credential from TTN to WisToolBox

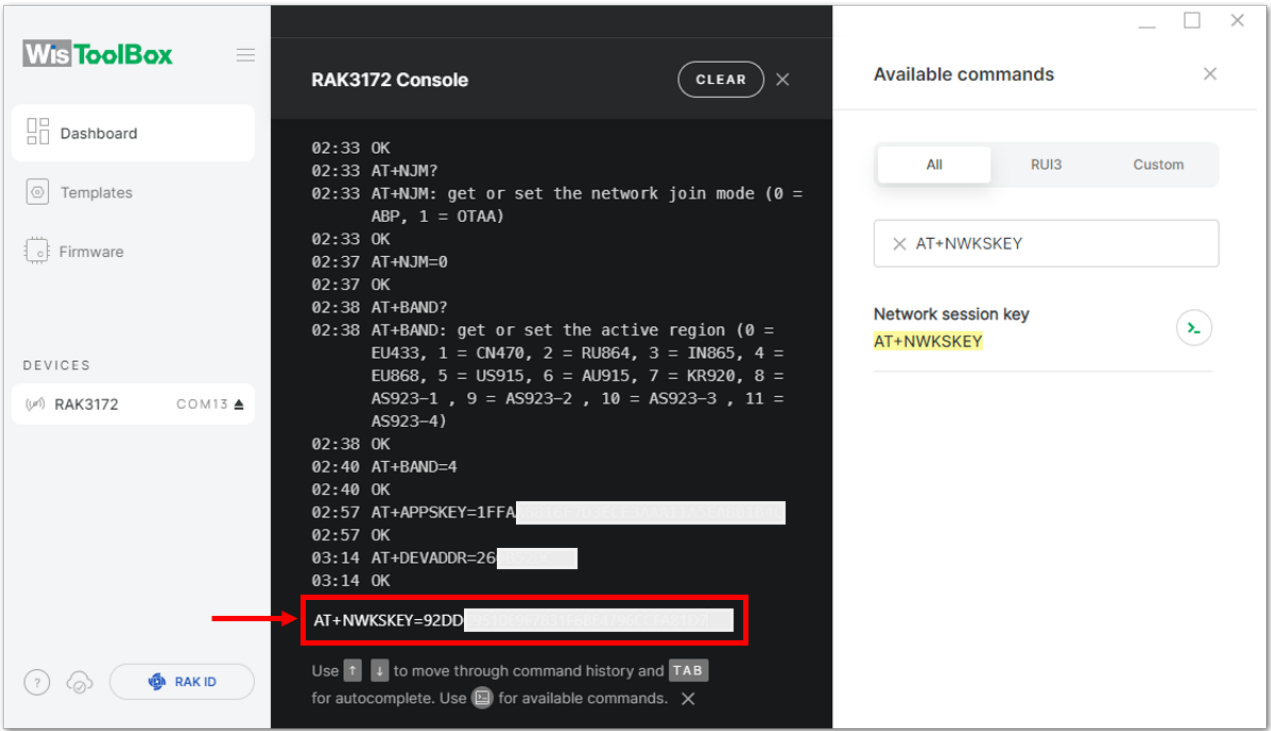


Figure 185: Setting up your Console

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.

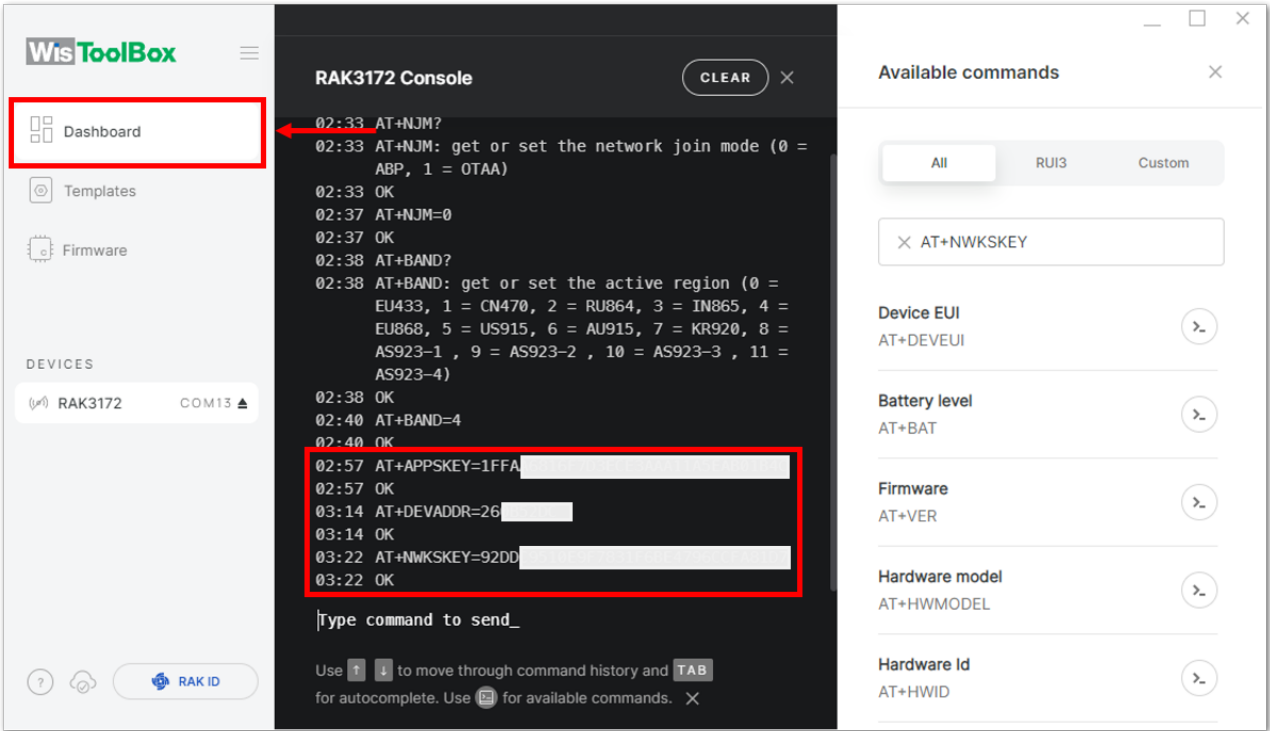


Figure 186: Setting up your Console

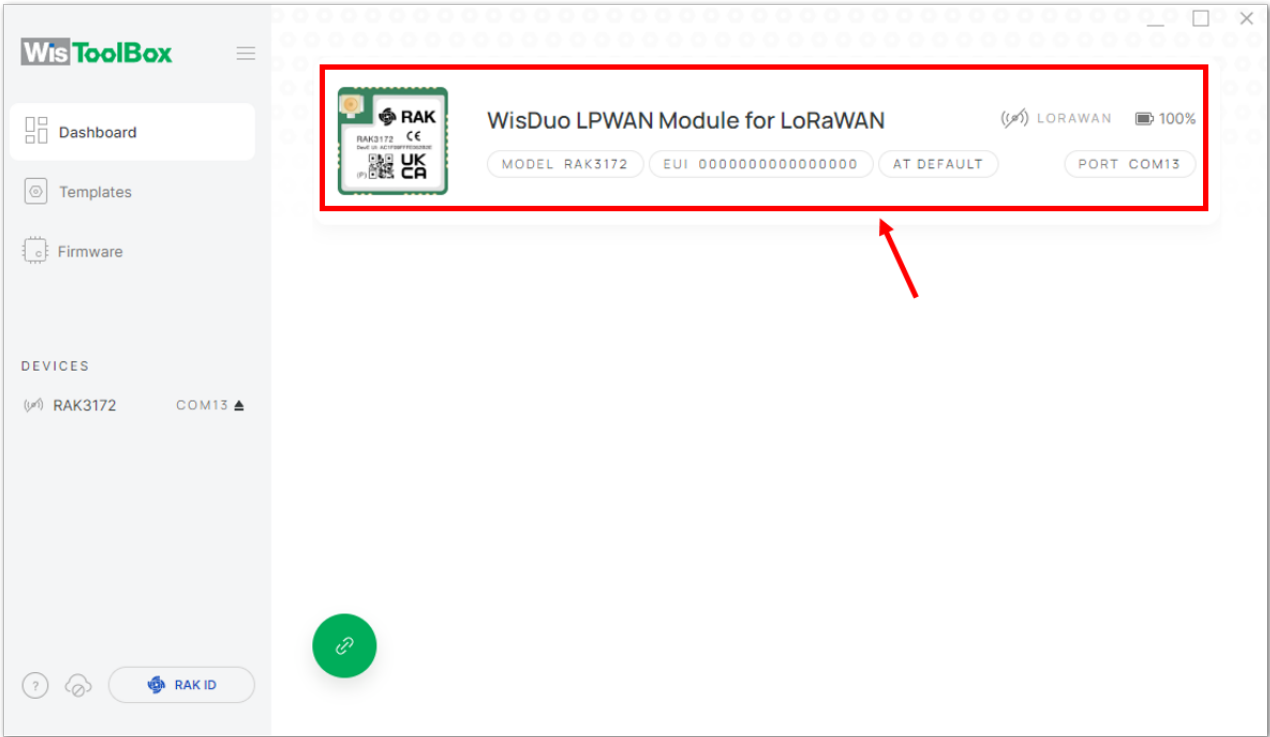


Figure 187: Setting up your Console

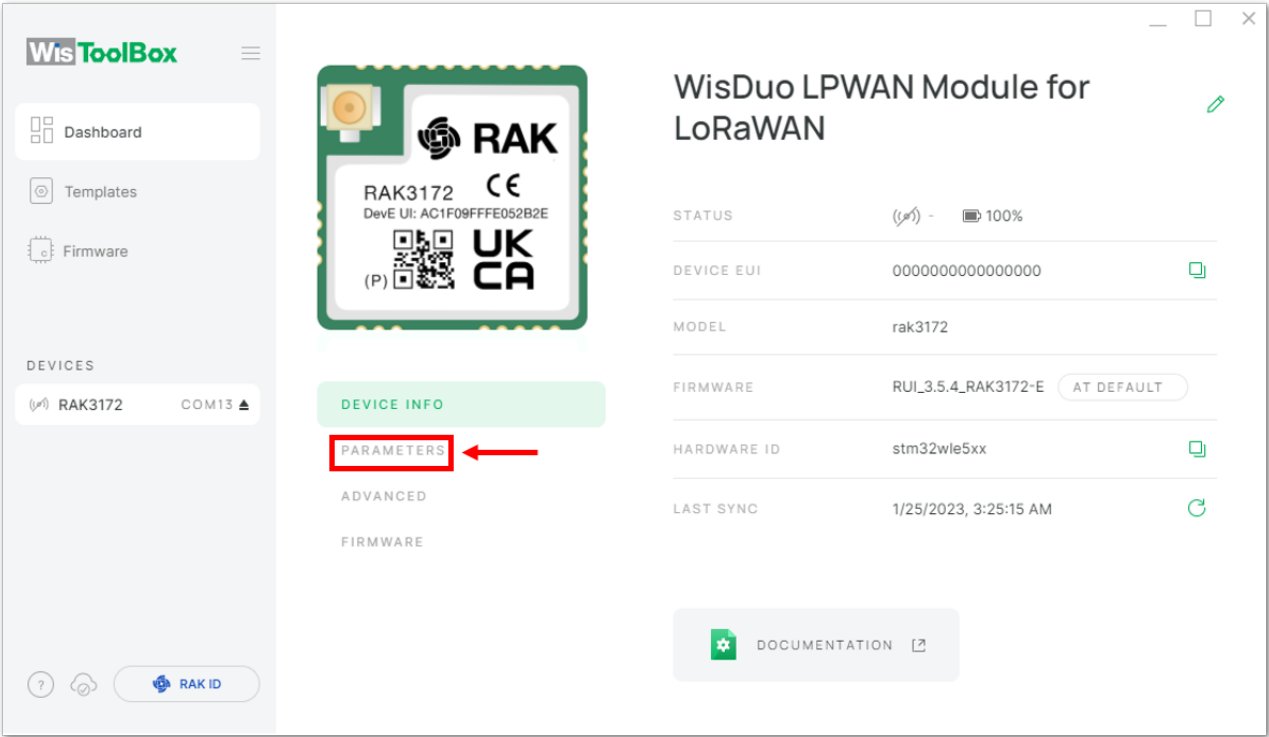


Figure 188: PARAMETERS

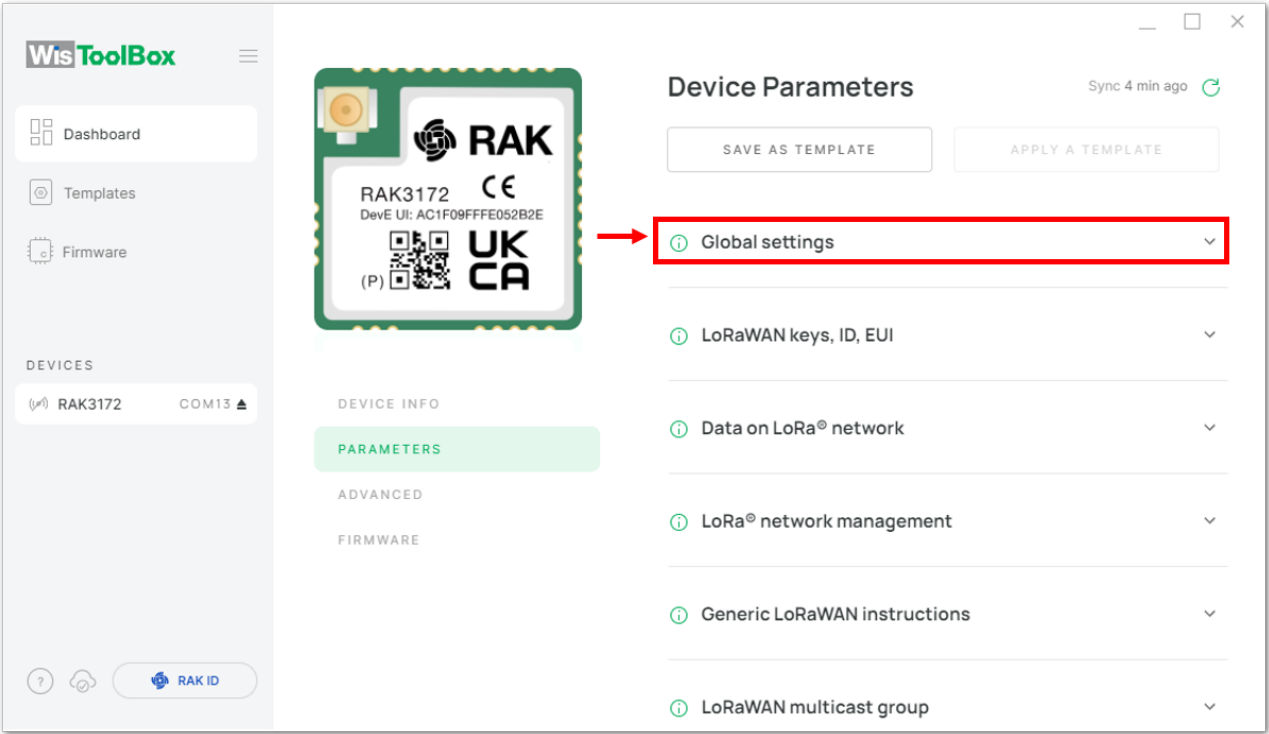


Figure 189: Global settings and LoRaWAN keys, ID, EUI

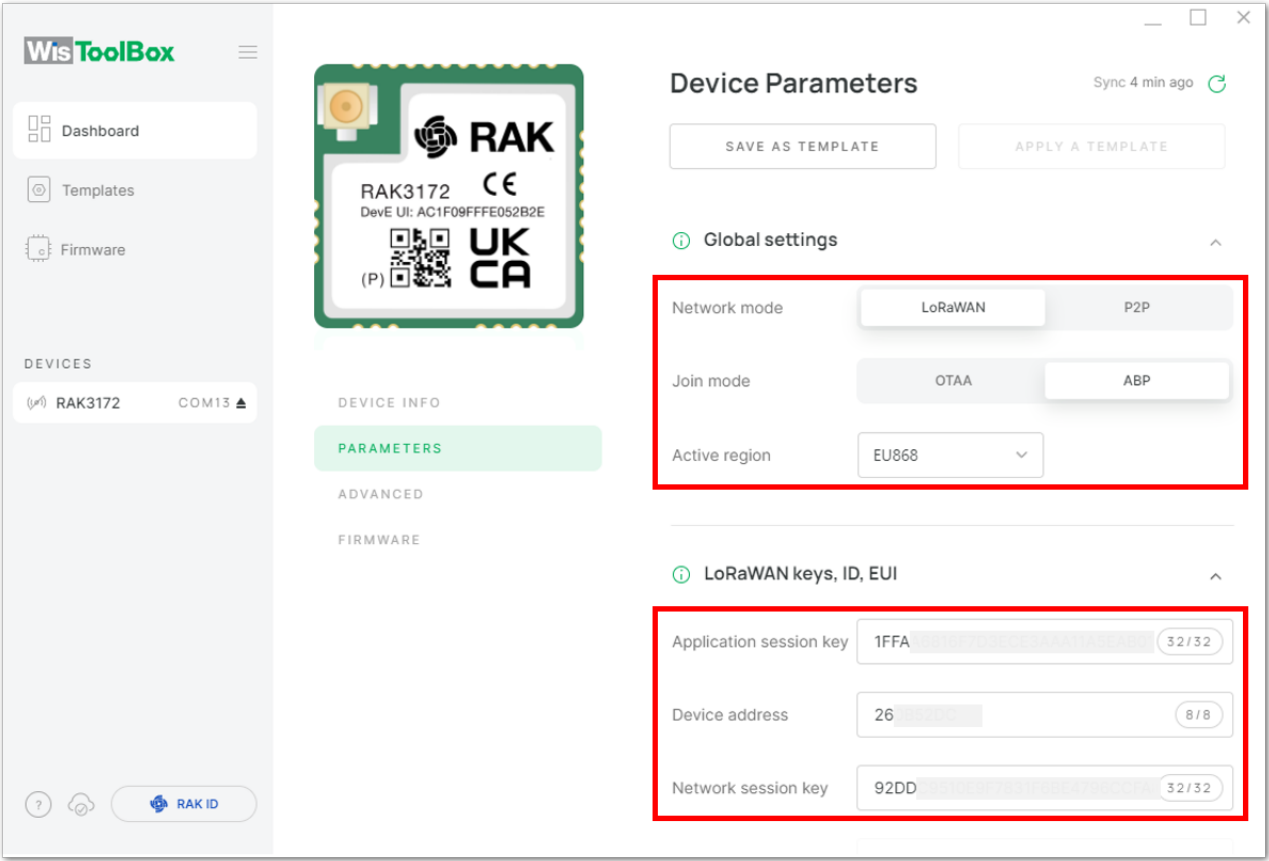


Figure 190: Global settings and LoRaWAN keys, ID, EUI details

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

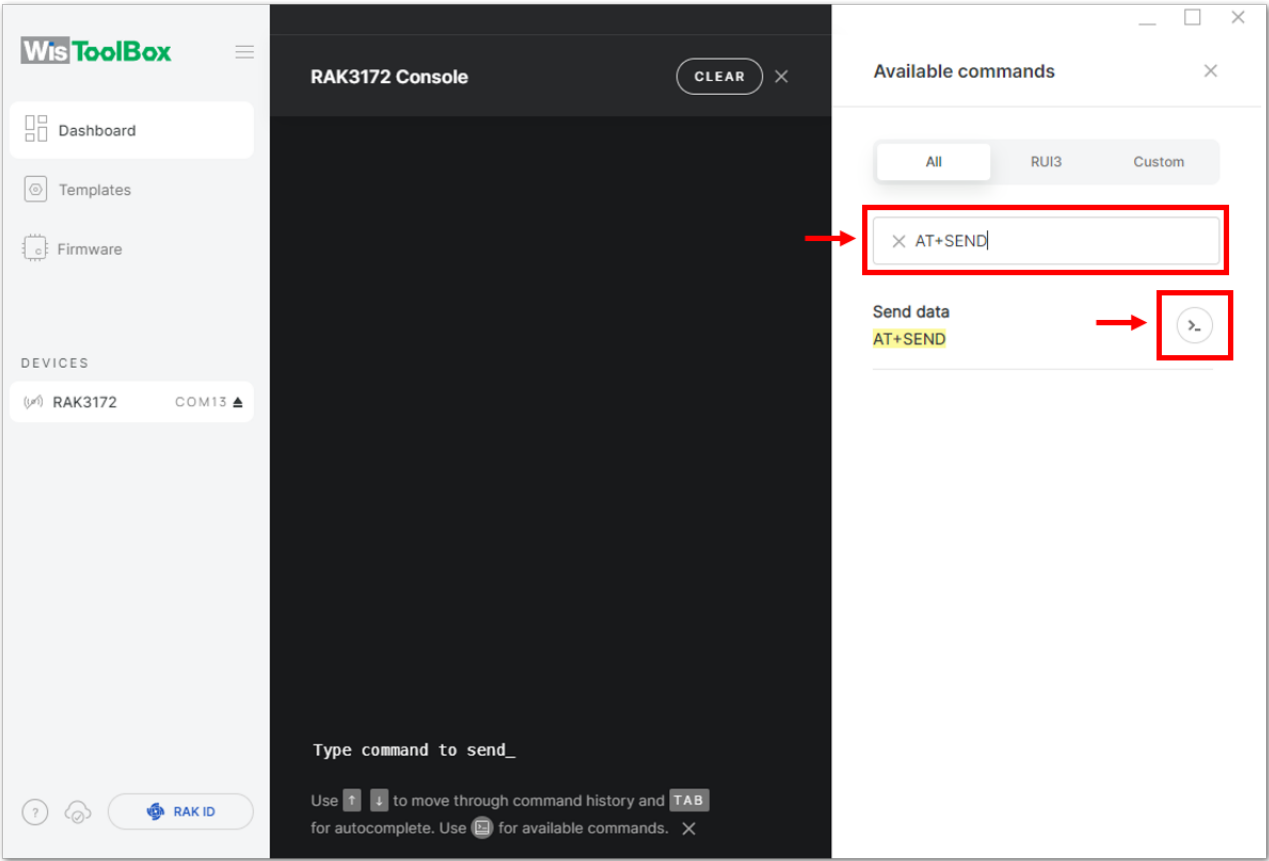


Figure 191: ABP device sending payload to the network

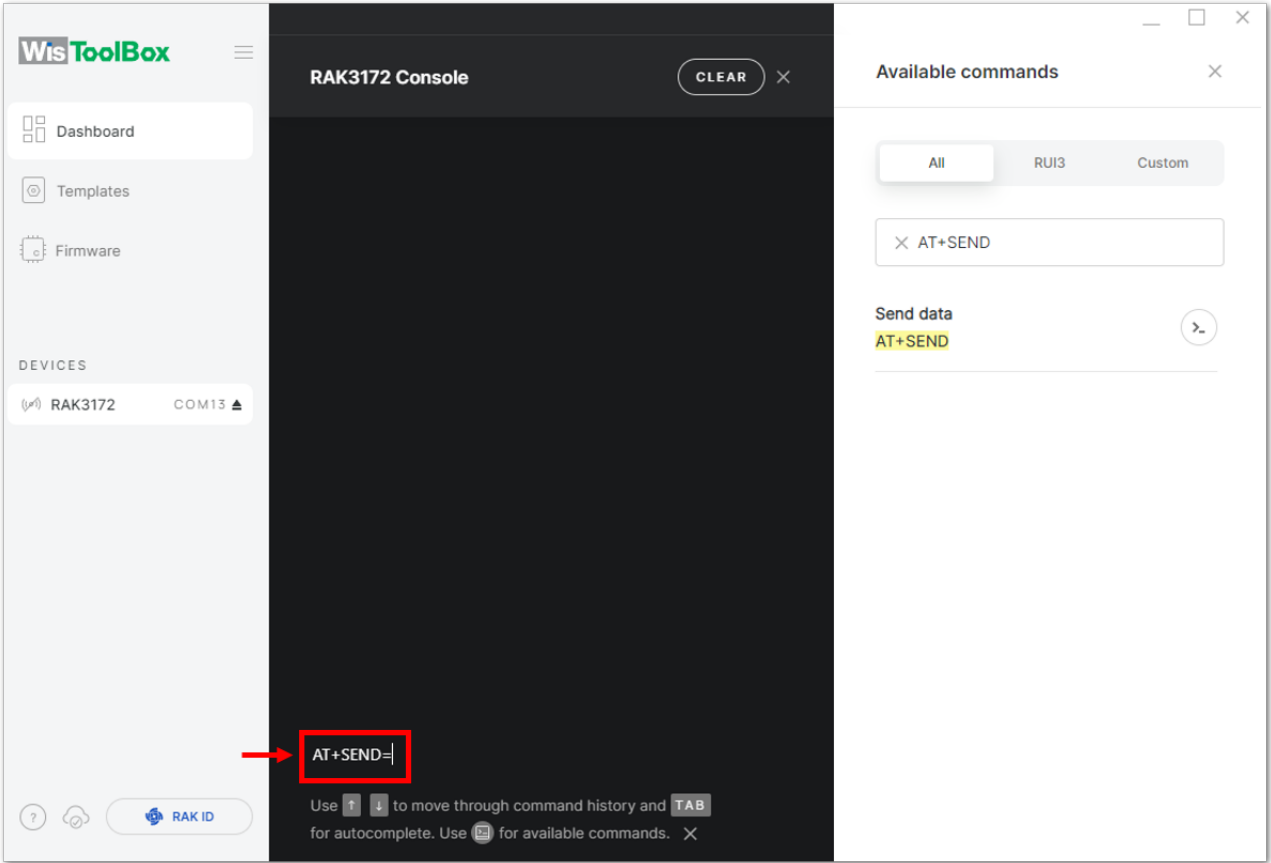


Figure 192: ABP device sending payload to the network

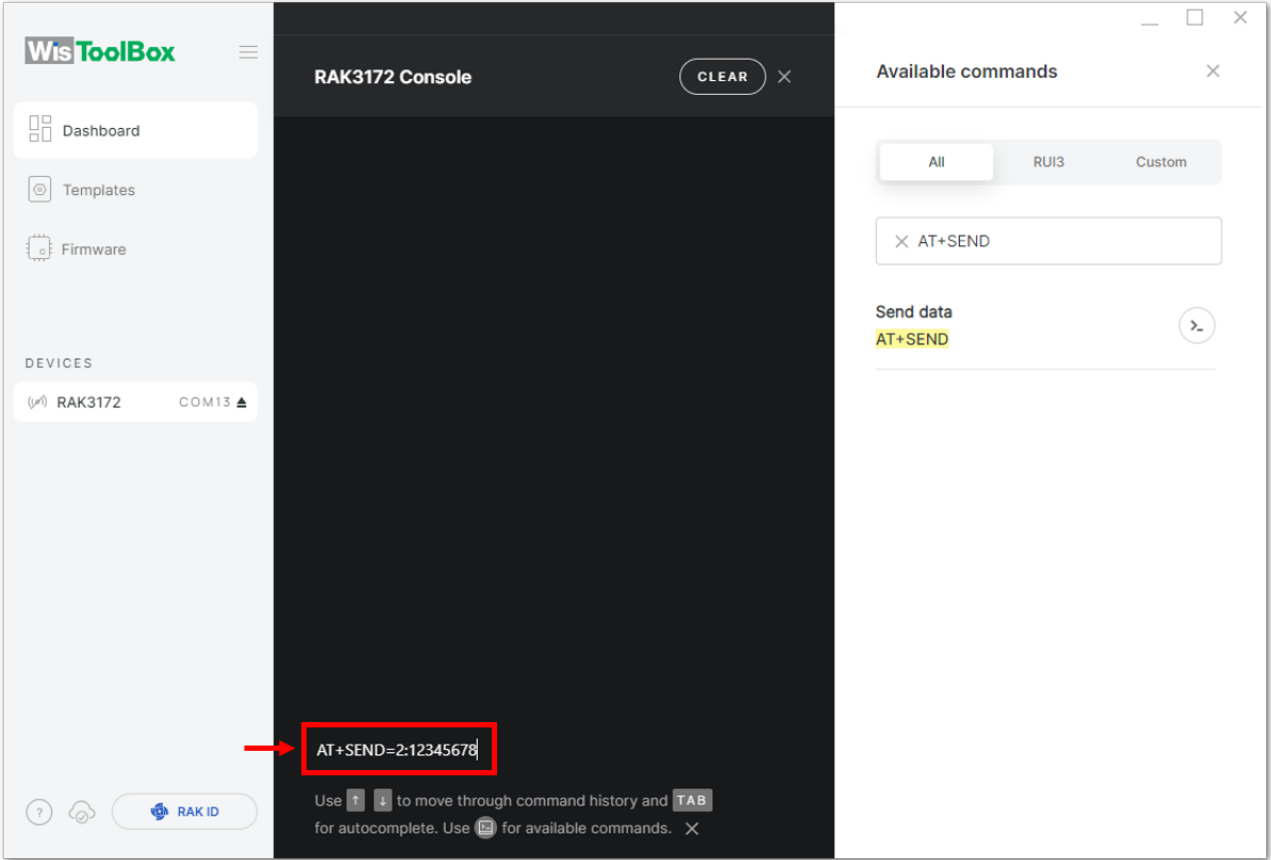


Figure 193: ABP device sending payload to the network

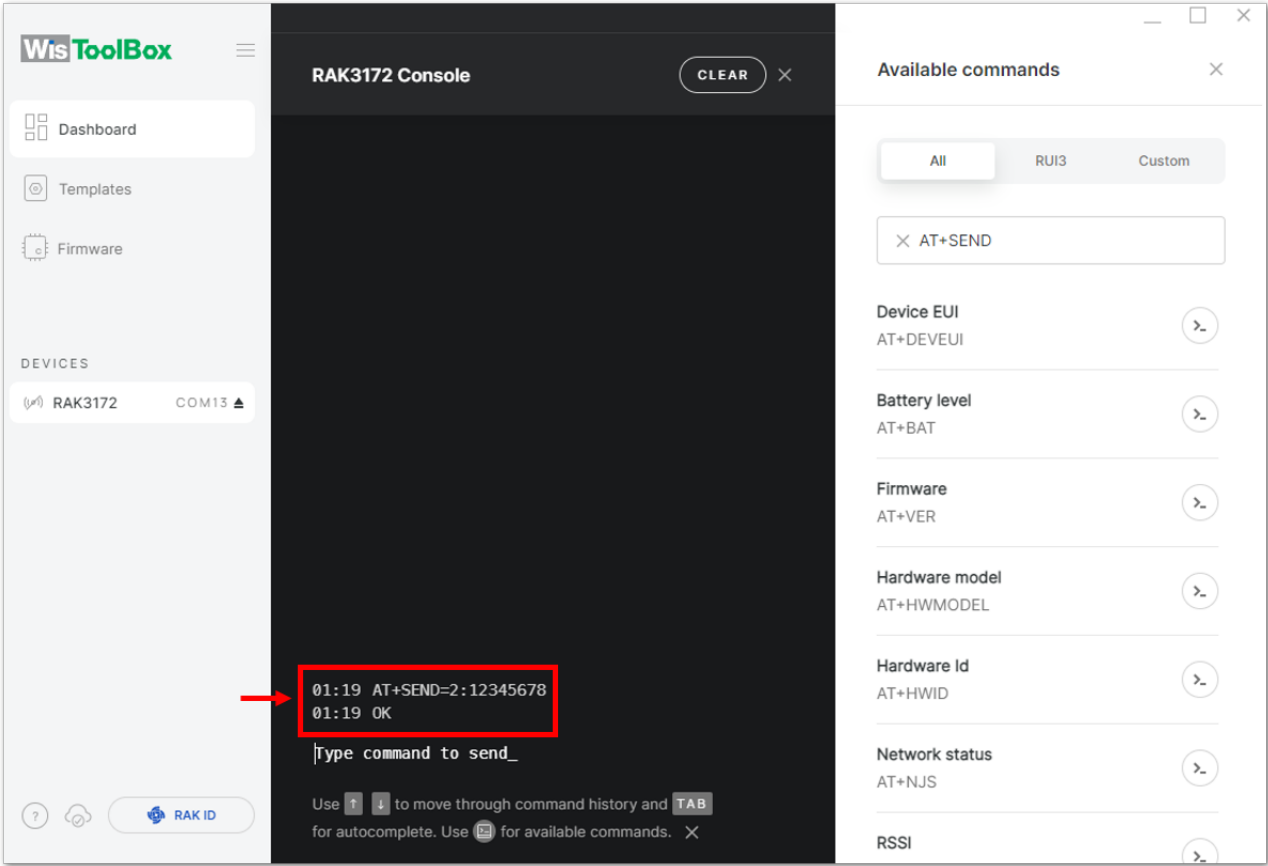


Figure 194: ABP device sending payload to the network

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.

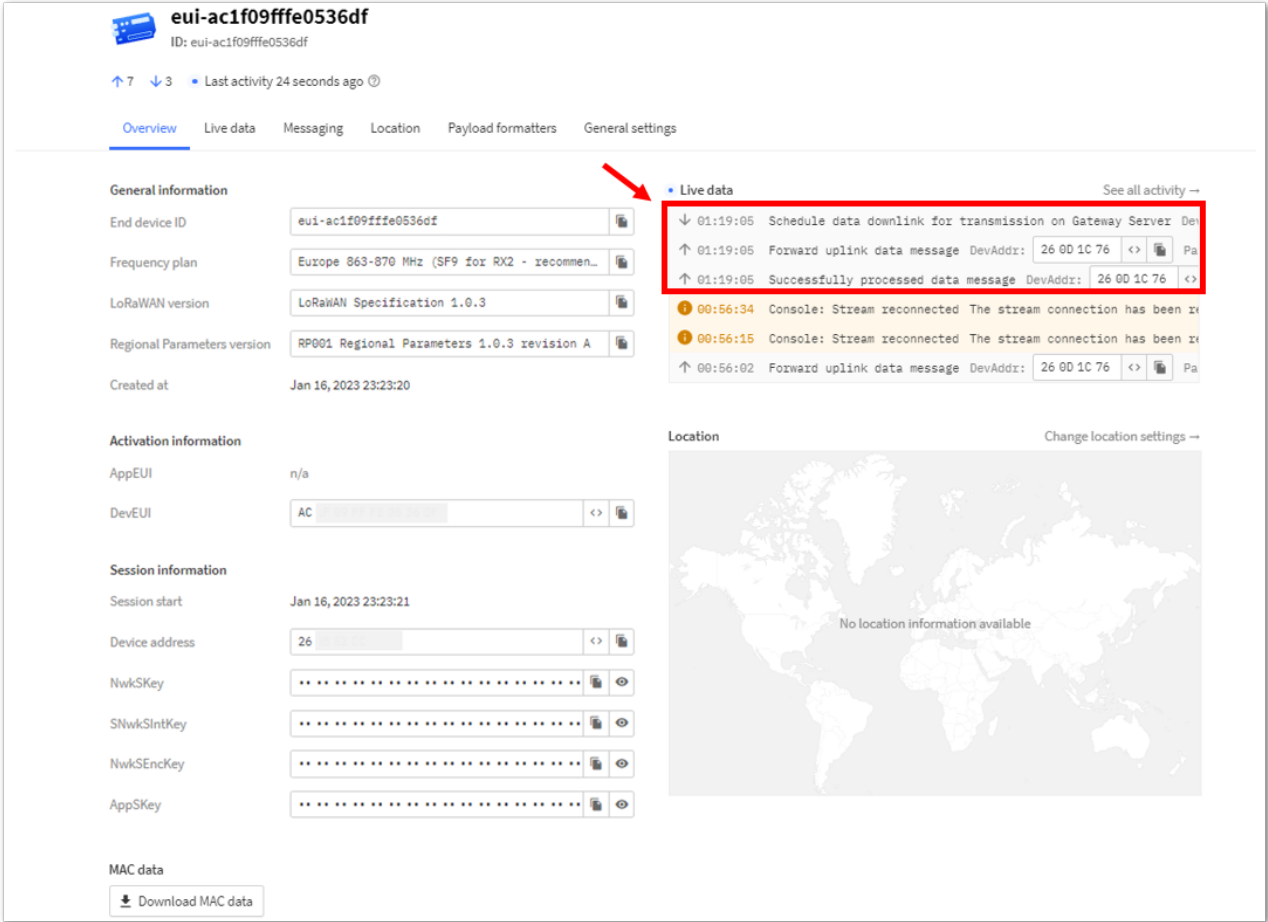


Figure 195: ABP Test Sample Data Sent Viewed in TTN

Connecting with ChirpStack

In this section, it shows how to connect the RAK3272S Breakout Board to the ChirpStack platform.

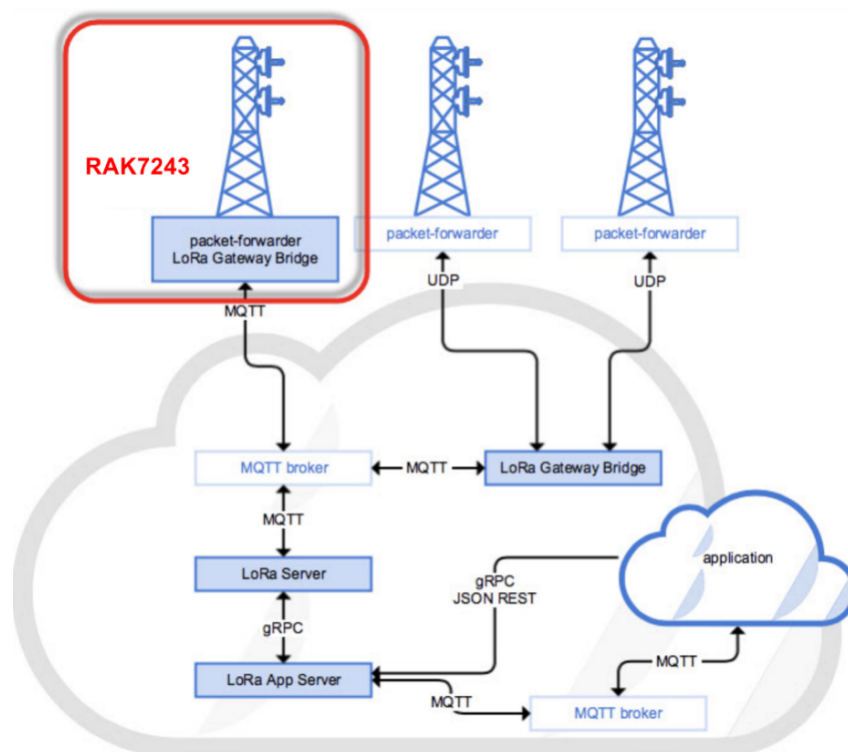


Figure 196: RAK3272S Breakout Board in the Context of the ChirpStack Platform

The ChirpStack or previously known as the LoRaServer project provides open-source components for building LoRaWAN networks. Like the case of TTN, the RAK3272S Breakout Board is located in the periphery and will transmit the data to the backend servers through a LoRaWAN gateway. Learn more about [ChirpStack](#) .

NOTE:

It is assumed that you are using RAK Gateway and its built-in ChirpStack. Also, the gateway with the ChirpStack must be configured successfully. For further information, check the RAK documents.

- In summary, these are the requirements:
 1. Have 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. RAK3272S Breakout Board

NOTE:

The frequency band used in the demonstration is EU868. Use a high-frequency version of RAK3272S. The product number should be “**RAK3272S (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 197**.

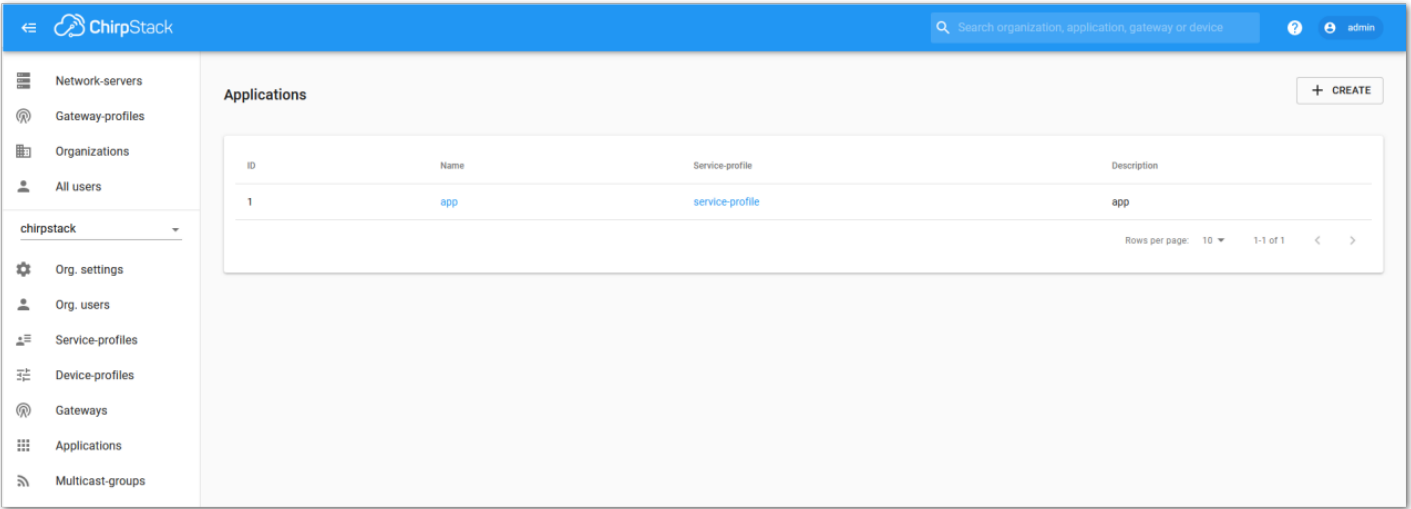


Figure 197: Application Section

3. By default, you should create a new application, although you can reuse the existing ones. For this setup, create a new Application by clicking on the “**CREATE**” button, and filling the required parameters, as shown in **Figure 198** and **Figure 199**.

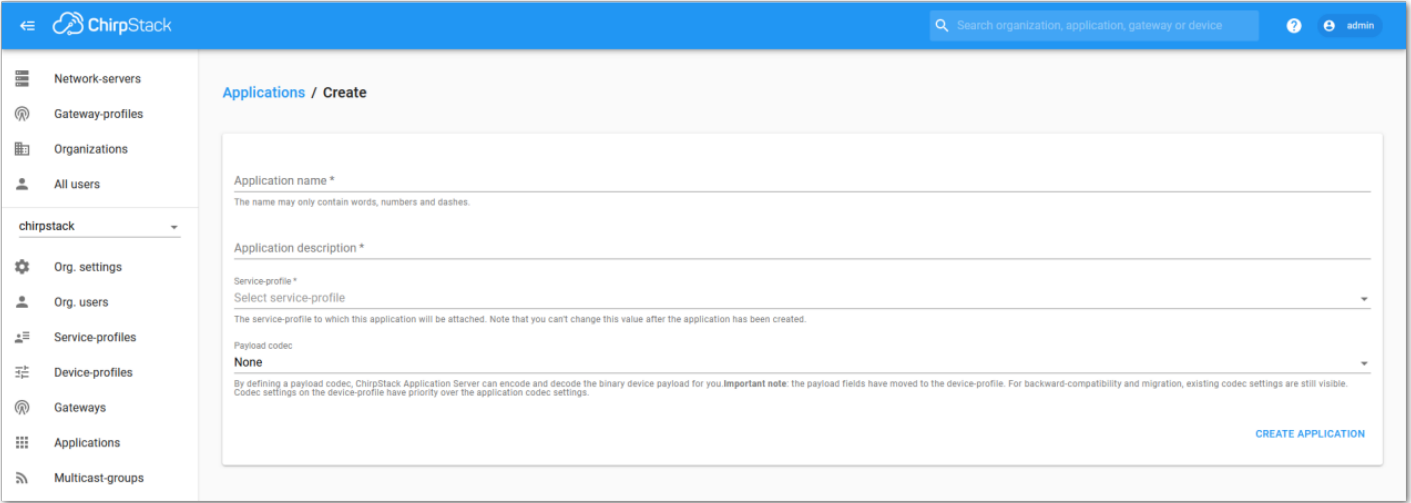


Figure 198: Creating a New Application

- 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.

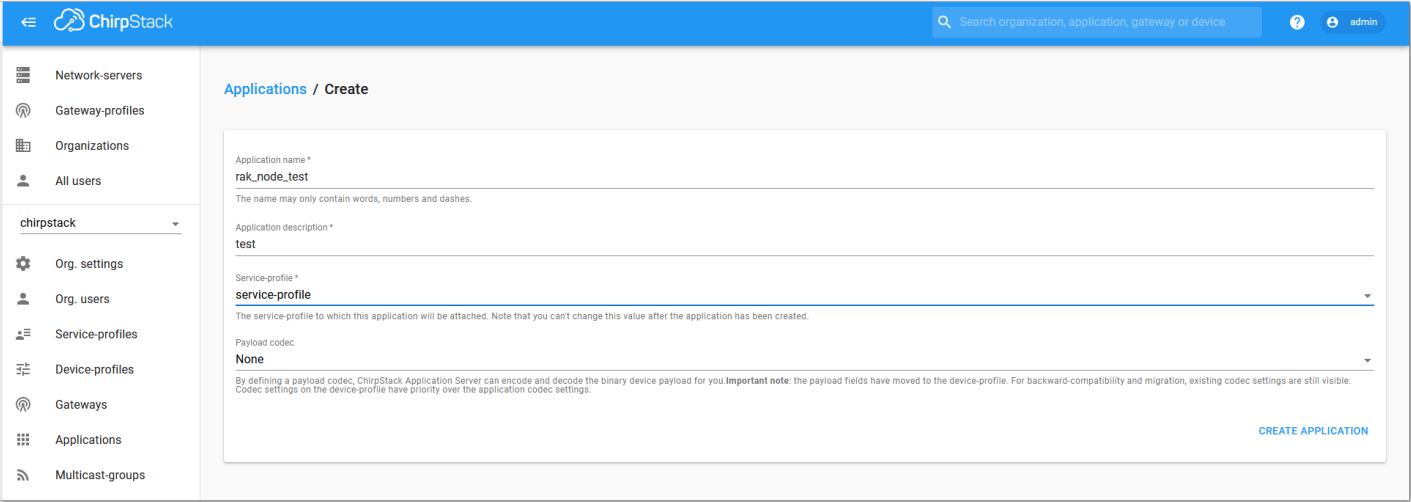


Figure 199: Filling Parameters of an Application

Register a New Device

1. Choose the **Application** created in the previous step, then select the **DEVICES** tab, as shown in **Figure 200** and **Figure 201**.
2. Once done, click “+ **CREATE**”.

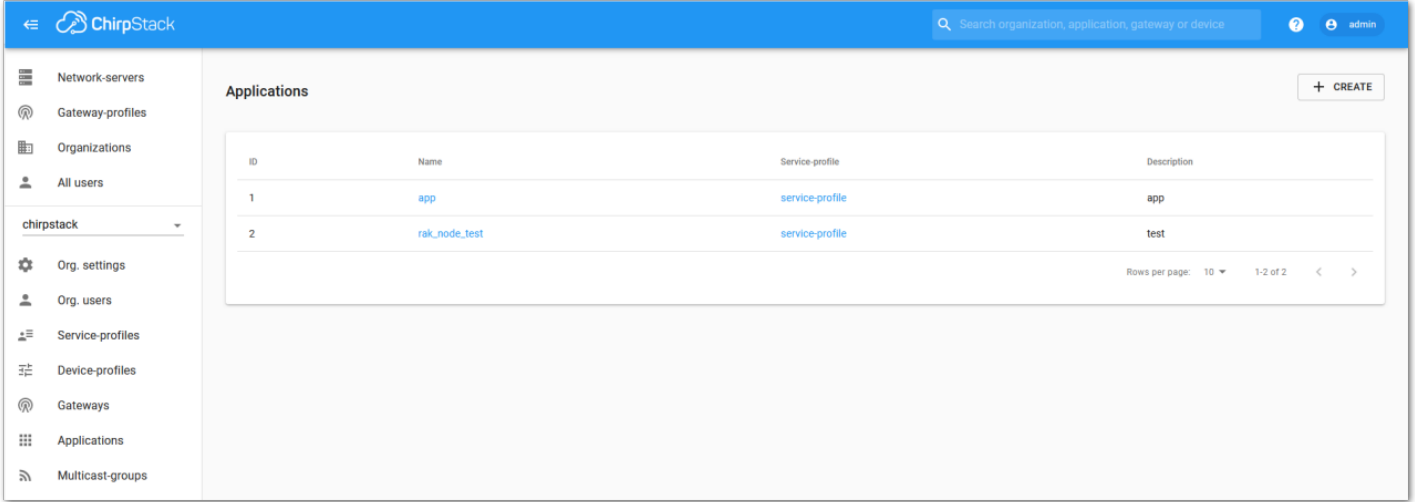


Figure 200: List of Applications Created

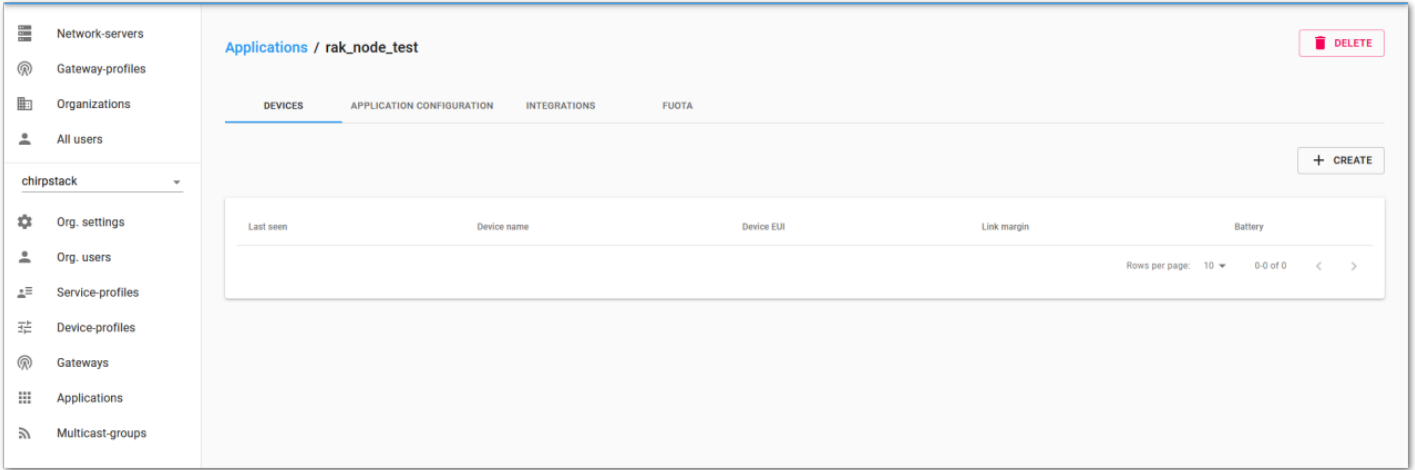


Figure 201: Device Tab of an Application

3. Once inside of the **DEVICE** tab, create a new device (LoRaWAN node) by clicking on the “+ **CREATE**” button.

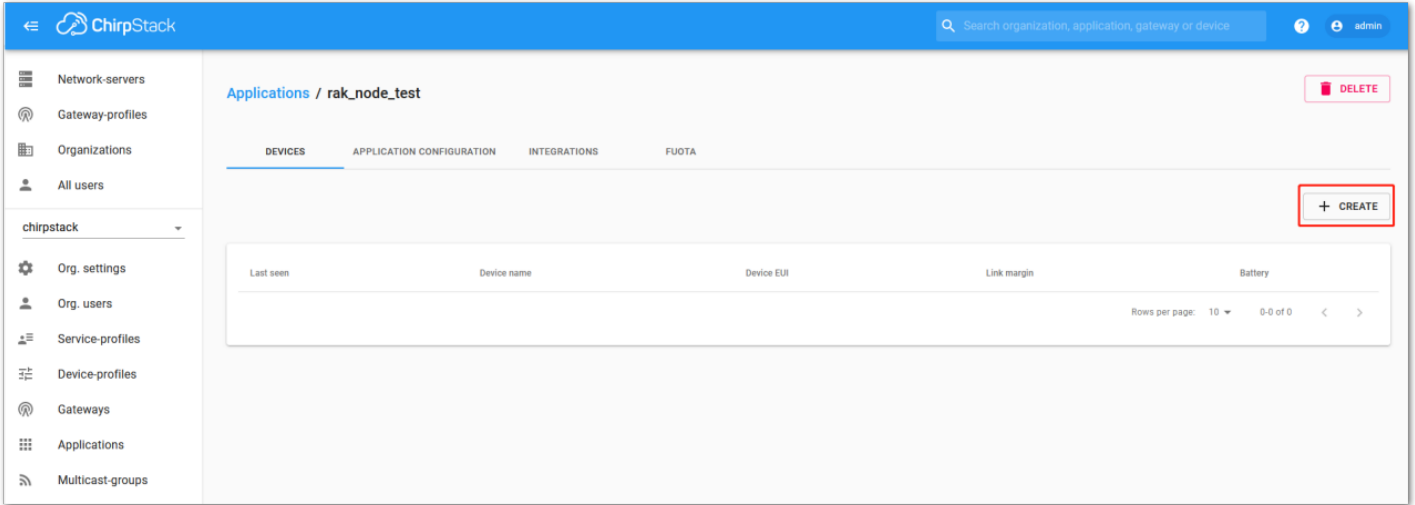


Figure 202: Add a New Device

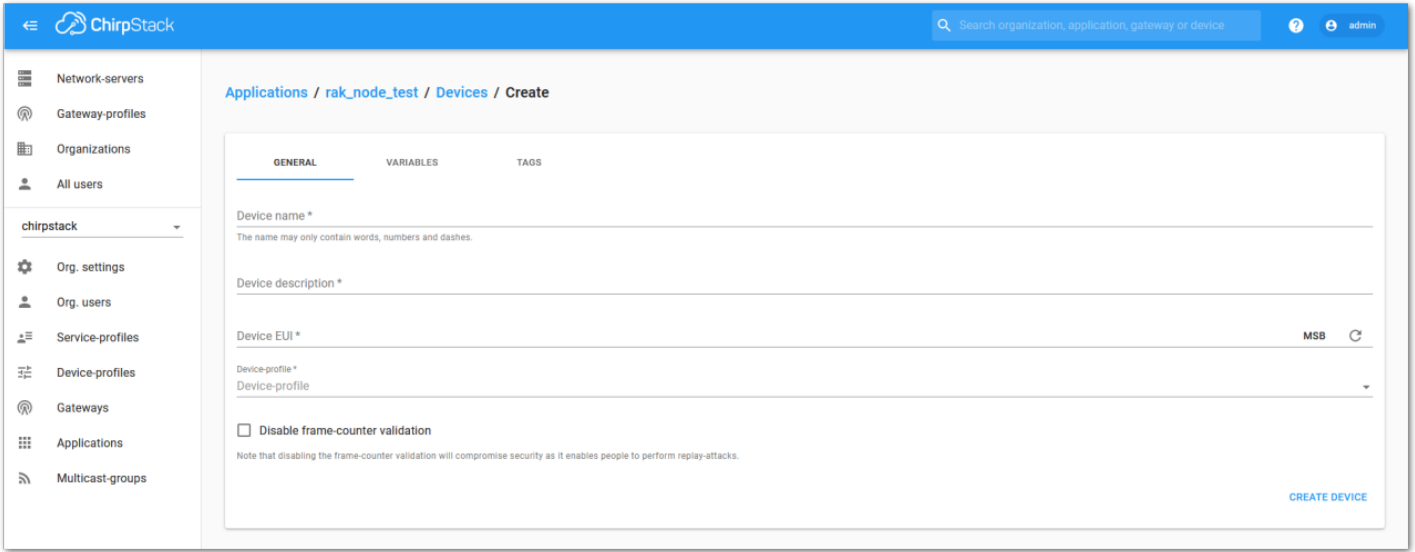


Figure 203: Chirpstack Adding Node into the RAK3272S Breakout Board

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 RAK3272S.

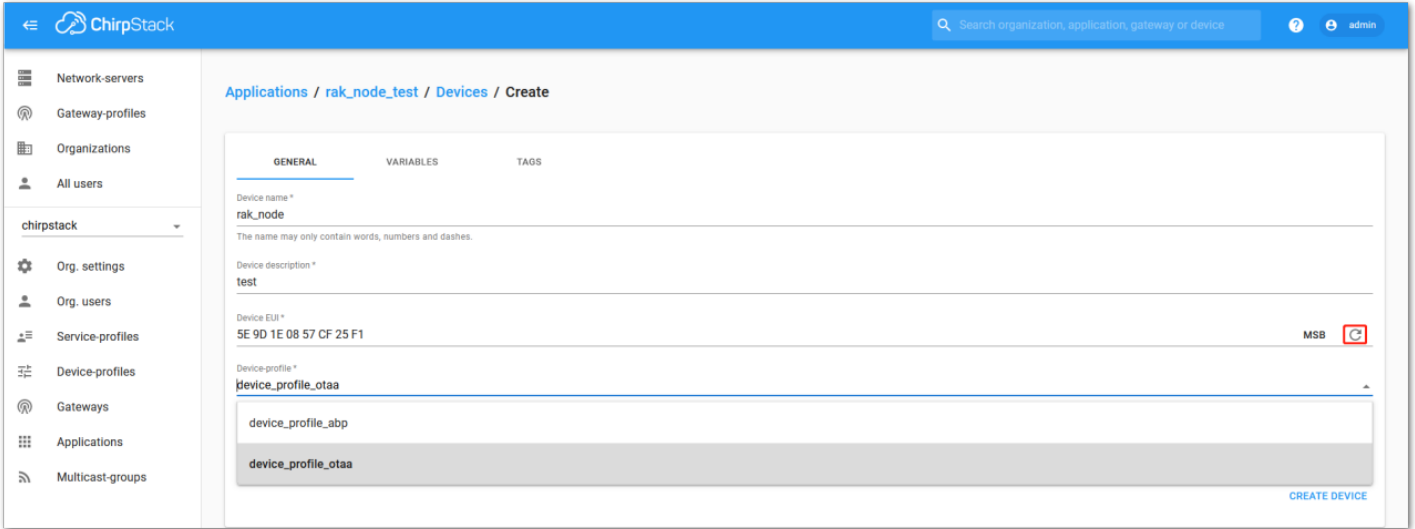


Figure 204: Generate a New Device EUI

Chirpstack OTAA Device Registration

1. If you have selected “**DeviceProfile_OTAA**”, as shown in **Figure 205**, then after the device is created, an Application Key must be also created for this device.

ChirpStack

Search organization, application, gateway or device

admin

Network-servers

Gateway-profiles

Organizations

All users

chirpstack

Org. settings

Org. users

Service-profiles

Device-profiles

Gateways

Applications

Multicast-groups

Applications / rak_node_test / Devices / Create

GENERAL

VARIABLES

TAGS

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

MSB

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.

CREATE DEVICE

Figure 205: 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 206**.

ChirpStack

Search organization, application, gateway or device

admin

Network-servers

Gateway-profiles

Organizations

All users

chirpstack

Org. settings

Org. users

Service-profiles

Device-profiles

Gateways

Applications

Multicast-groups

Applications / rak_node_test / Devices / rak_node

DELETE

DETAILS

CONFIGURATION

KEYS (OTAA)

ACTIVATION

DEVICE DATA

LORAWAN FRAMES

FIRMWARE

Application key *

F9 21 D5 0C D7 D0 2E E3 C5 E6 14 21 54 F2 74 B2

MSB

For LoRaWAN 1.0 devices. In case your device supports LoRaWAN 1.1, update the device-profile first.

Gen Application key

MSB

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.

SET DEVICE-KEYS

Figure 206: 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 207**, a new device should be listed in the DEVICES tab. The most important parameters, such as the Device EUI are shown in the summary.

ChirpStack

Search organization, application, gateway or device

admin

Network-servers

Gateway-profiles

Organizations

All users

chirpstack

Org. settings

Org. users

Service-profiles

Device-profiles

Gateways

Applications

Multicast-groups

Applications / rak_node_test

DELETE

DEVICES

APPLICATION CONFIGURATION

INTEGRATIONS

FUOTA

+ CREATE

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

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

Figure 207: 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 208**.

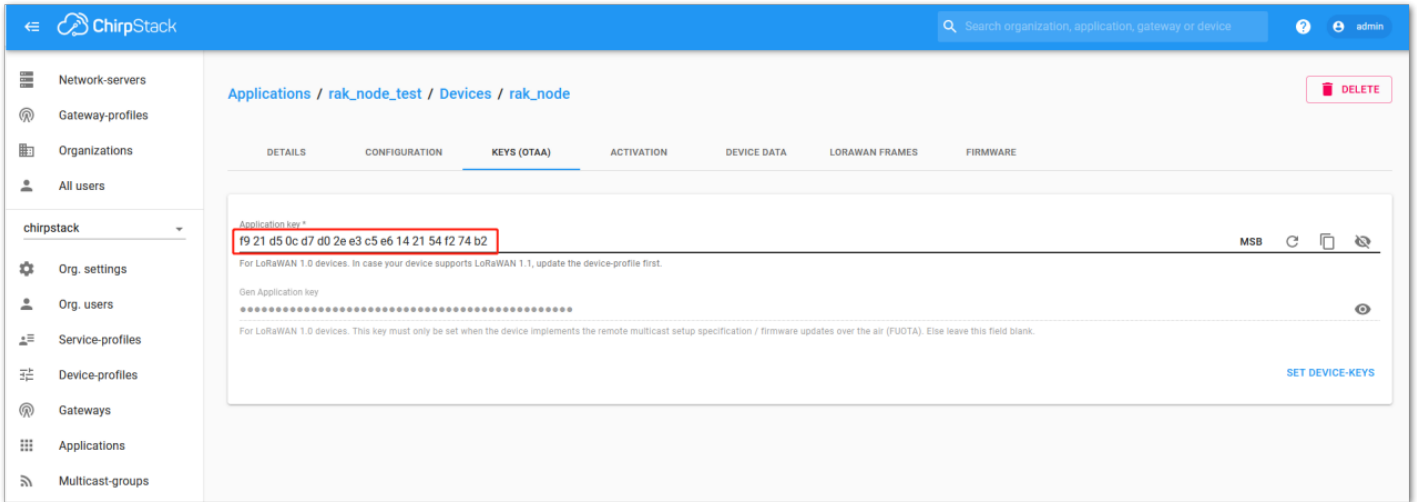


Figure 208: Application Key Associated with the New Device


NOTE:

Standard OTAA mode requires the **Device EUI**, **Application Key**, and **Application EUI**, but in the 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 RAK3272S Breakout Board supports a series of AT commands to configure its internal parameters and control the functionalities of the board.

1. To set up the RAK3272S Breakout Board to join the Chirpstack using OTAA, start by connecting the RAK3272S Breakout Board to the computer (see [Figure 21](#)) 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 board 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 209**.

 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.

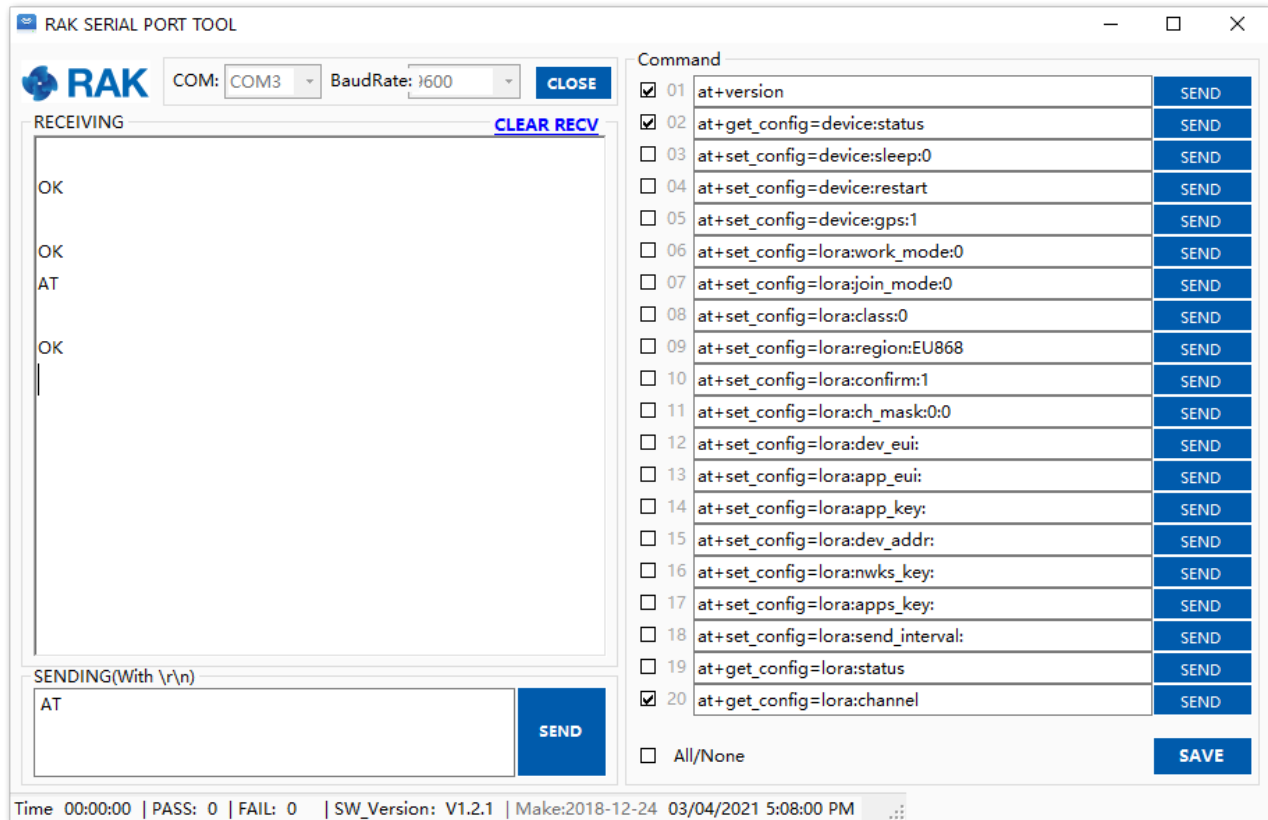


Figure 209: at+version command response

2. The next step is to configure the OTAA LoRaWAN parameters in RAK3272S:

- 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 RAK3272S 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 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

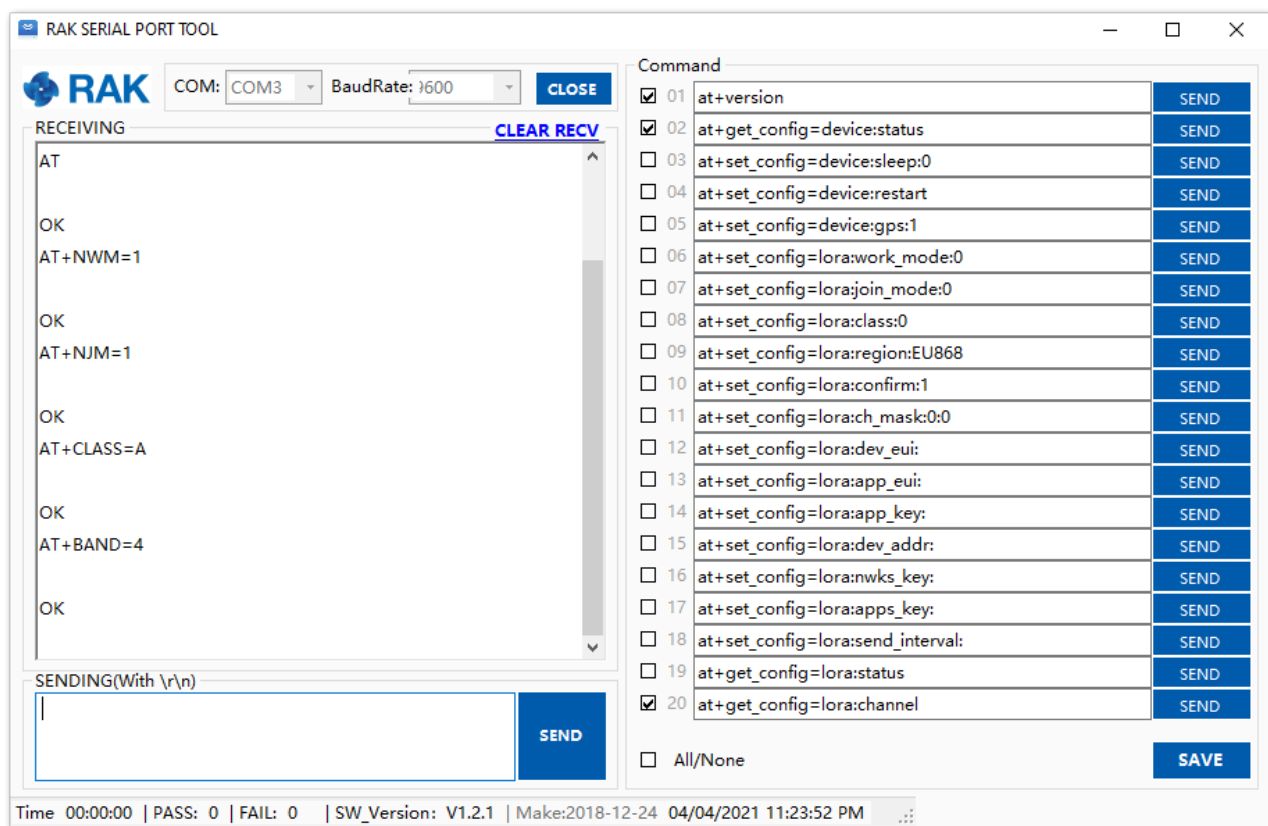


Figure 210: Configuring LoRa Parameters

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
```

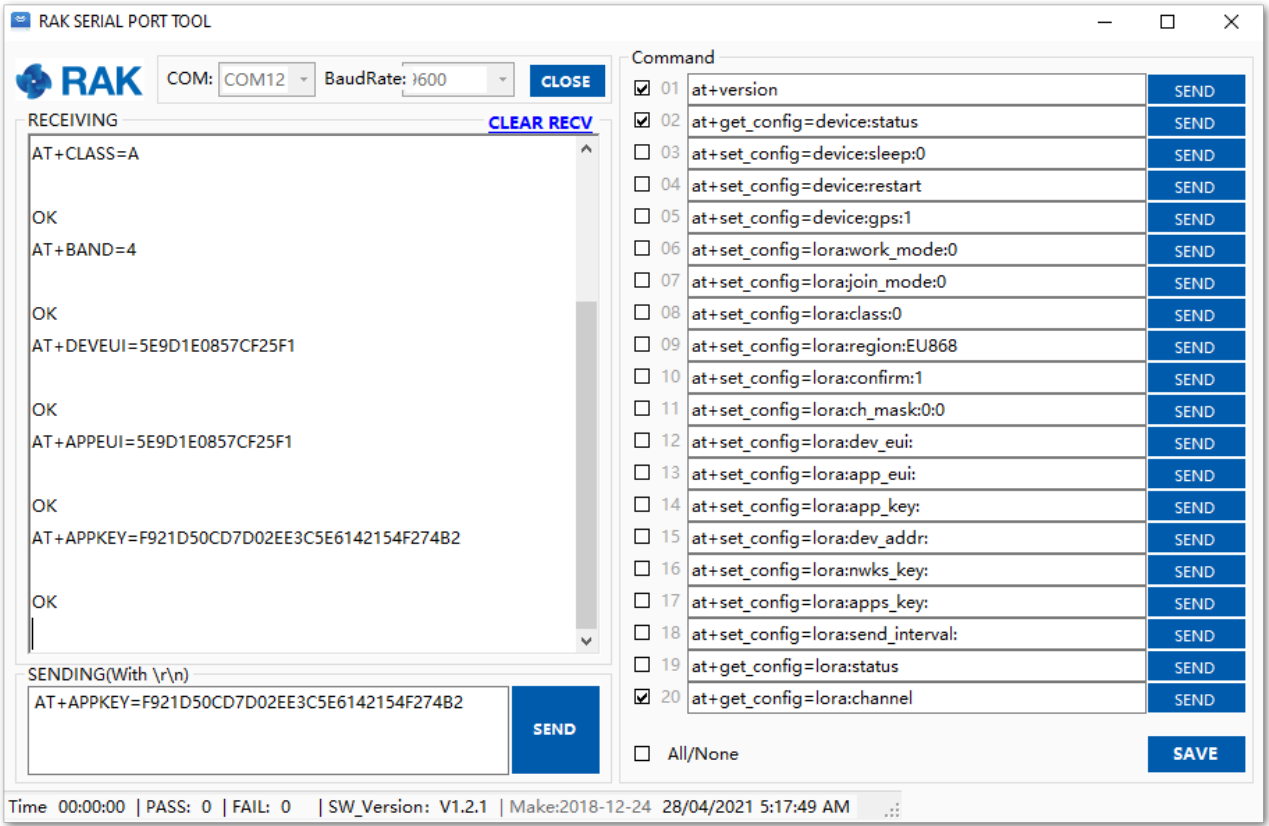


Figure 211: Configuring LoRa Parameters

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

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

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.

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

 **NOTE:**

- If the OTAA device join failed, 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 by using `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.

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

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

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

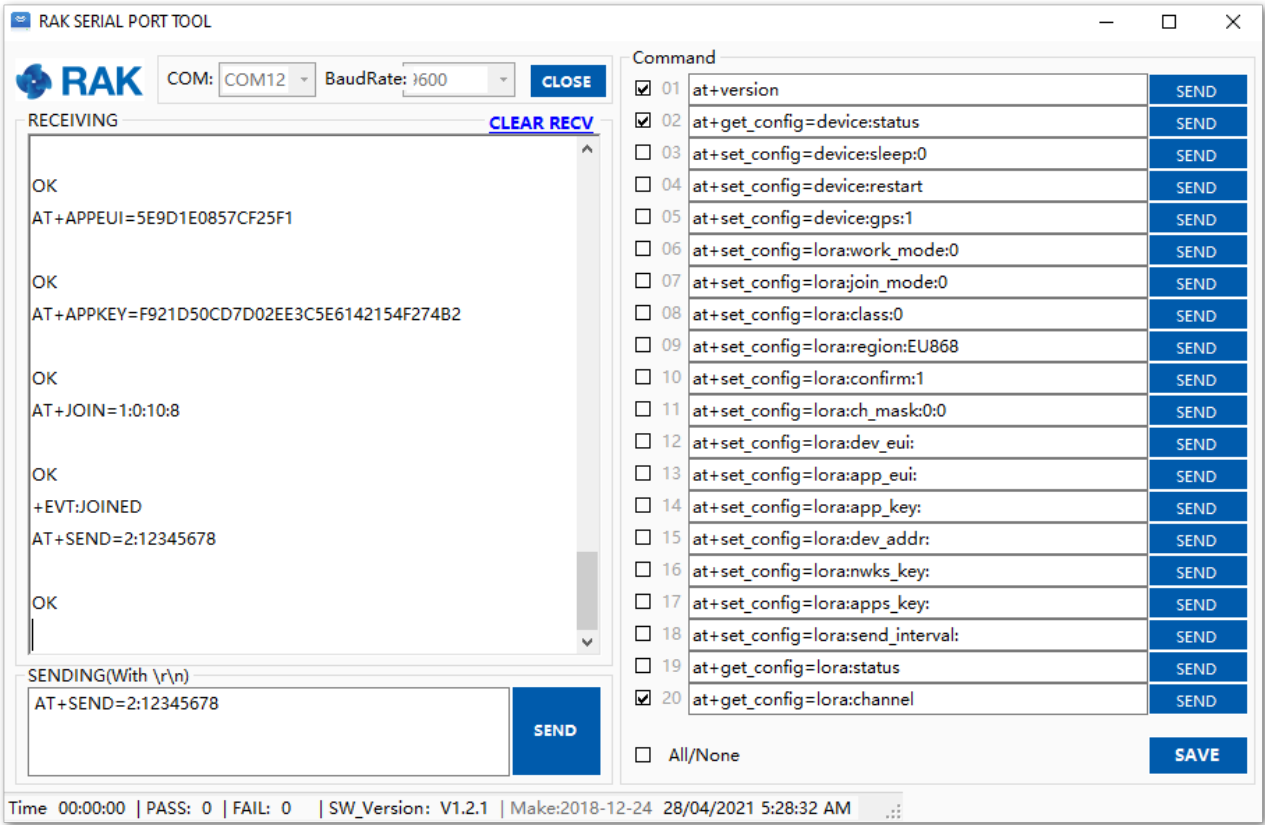


Figure 212: OTAA Test Sample Data Sent via RAK Serial Port Tool

7. On the ChirpStack platform, you should see the join and uplink messages in the LORAWAN FRAMES tab, as shown in **Figure 213**. By convention, messages sent from nodes to gateways are considered as **Uplinks** while messages sent by gateways to nodes are considered as **Downlinks**.

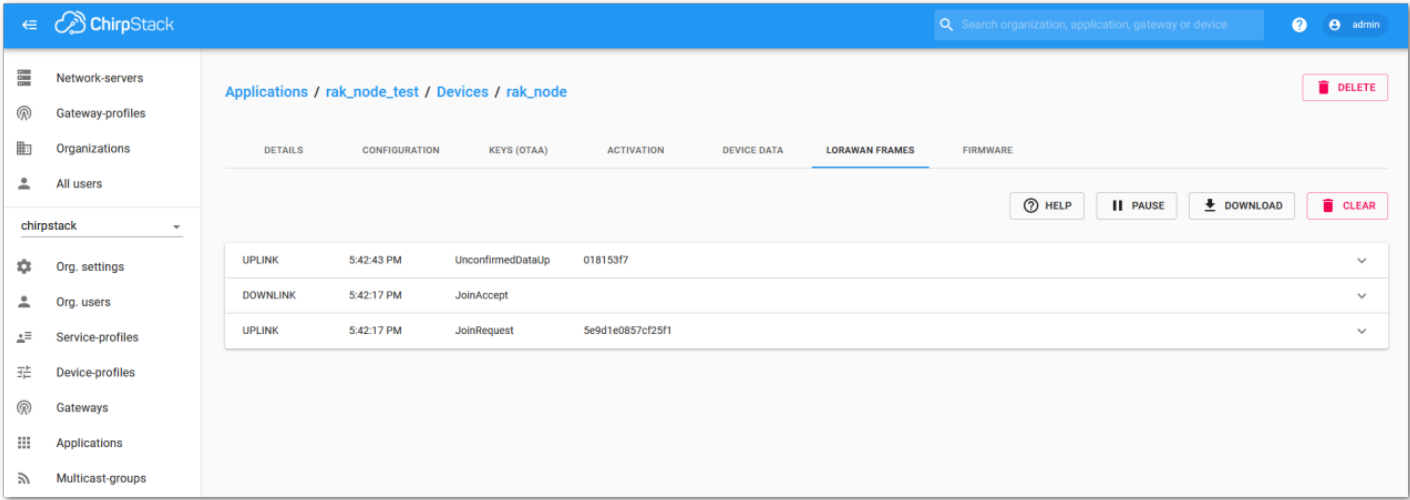


Figure 213: Chirpstack Data Received Preview

Chirpstack ABP Device Registration

1. During the registration of a new device, if you select “**DeviceProfile_ABP**”, as shown in **Figure 214**, then the ChirpStack platform will assume that this device will join the LoRaWAN network using the ABP mode.

NOTE:

Check “**Disable counting frame verification**”. During the test, when the board is restarted, the frame counting number will be also be restarted from zero. This would cause a synchronization problem with the ChirpStack server treating it as a replay attack. For the testing purpose, it is safe to disable this feature, but remember to activate it in a production environment.

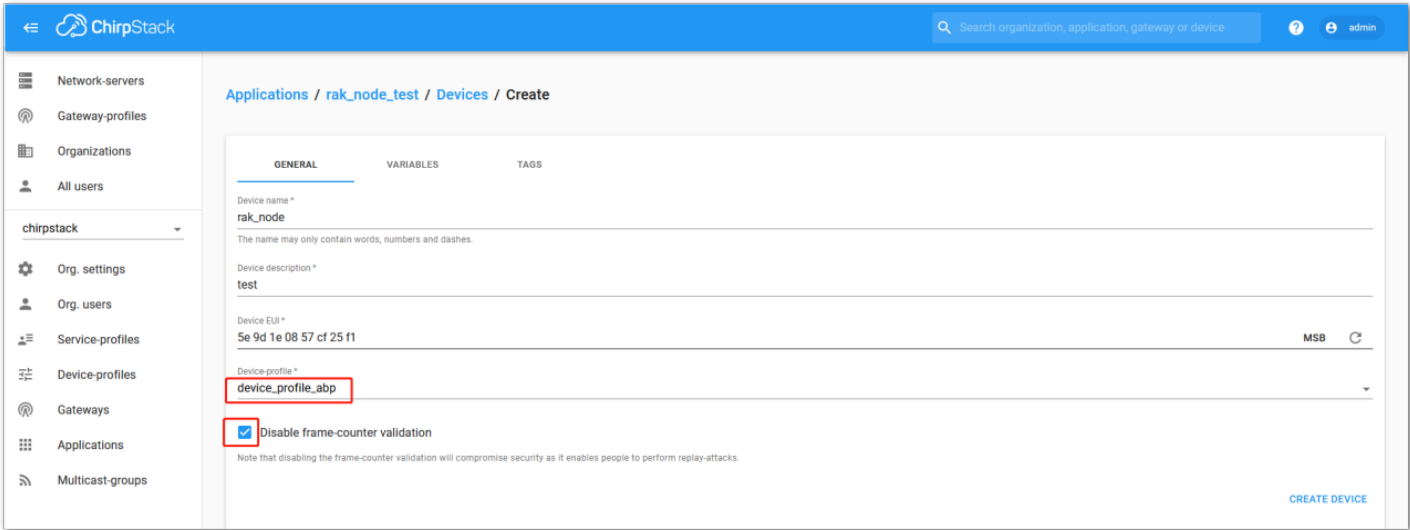


Figure 214: ChirpStack Console, Configuring a Device

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

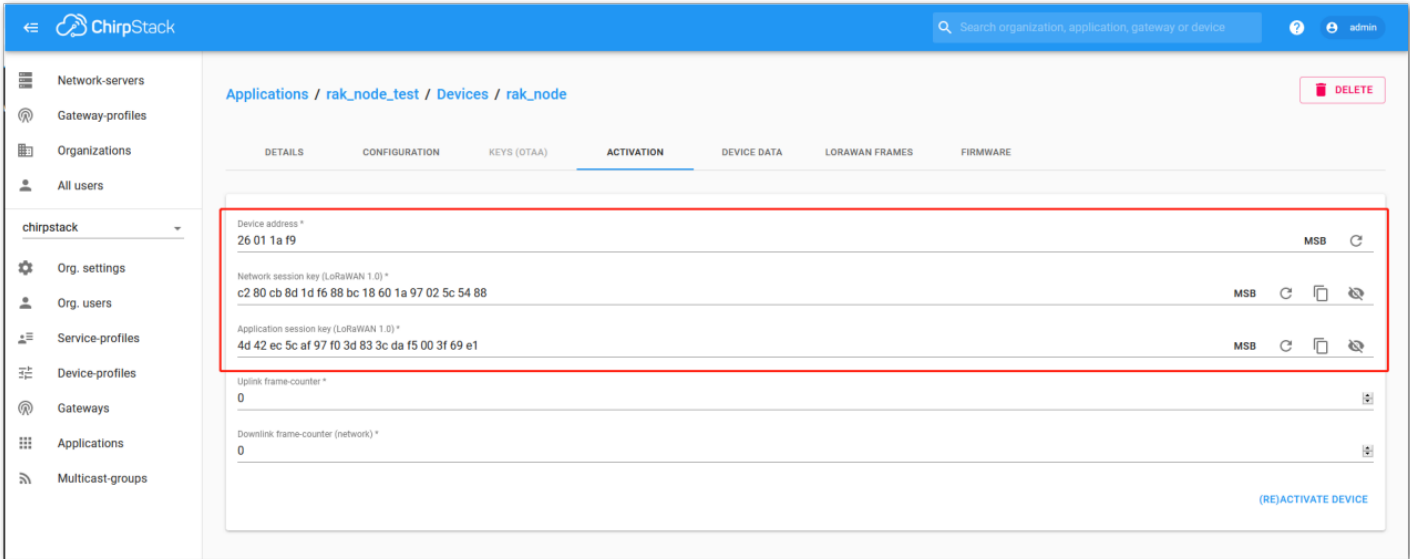


Figure 215: Chirpstack ABP Activation Parameters Needed

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 RAK3272S Breakout Board to join the Chirpstack using ABP, start by connecting the RAK3272S Breakout Board to the computer (see [Figure 21](#)) 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 board 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 216**.

 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.

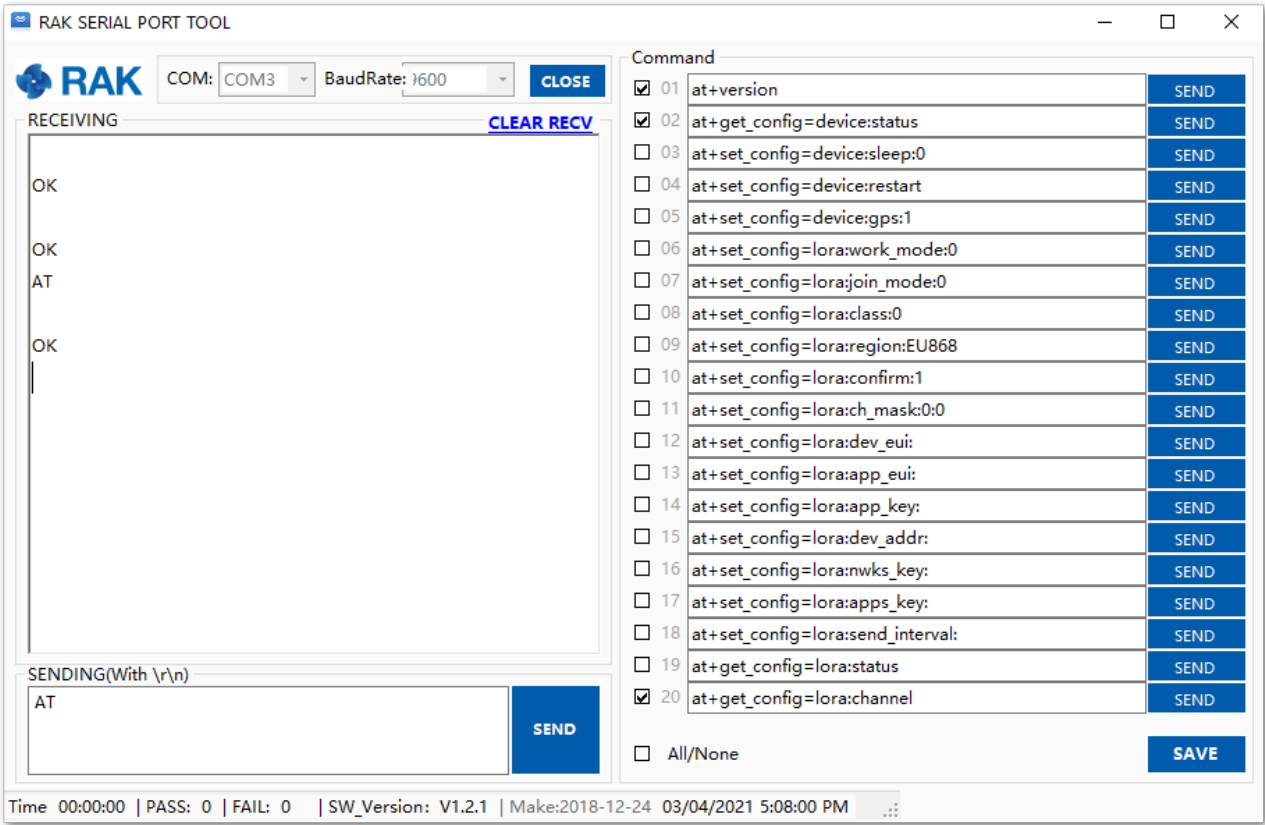


Figure 216: at+version command response

2. The next step is to configure the ABP LoRaWAN parameters in RAK3272S:

- 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+NM=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 RAK3272S 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 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

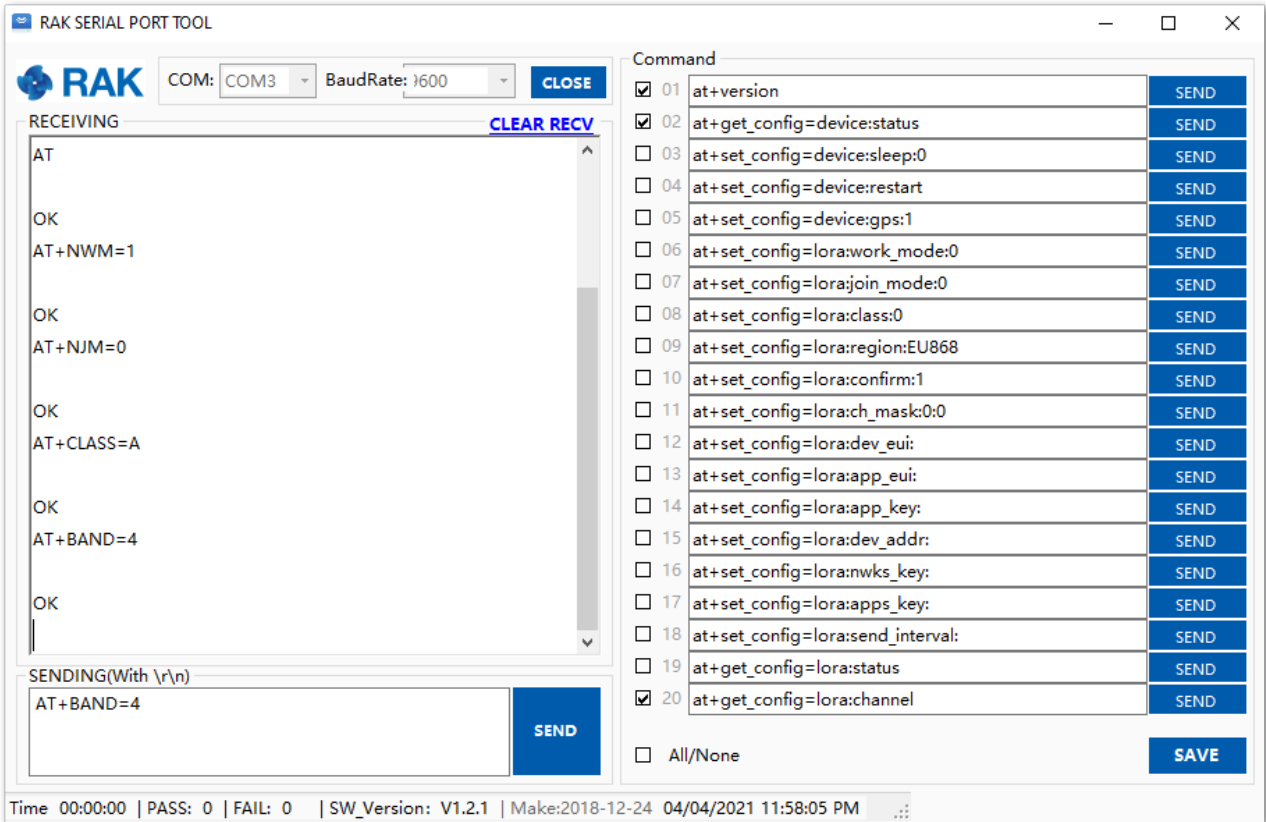


Figure 217: Configuring LoRa Parameters

3. After the configuration of the LoRaWAN parameters, the next step is to set up the device address and session keys. You need to 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
```

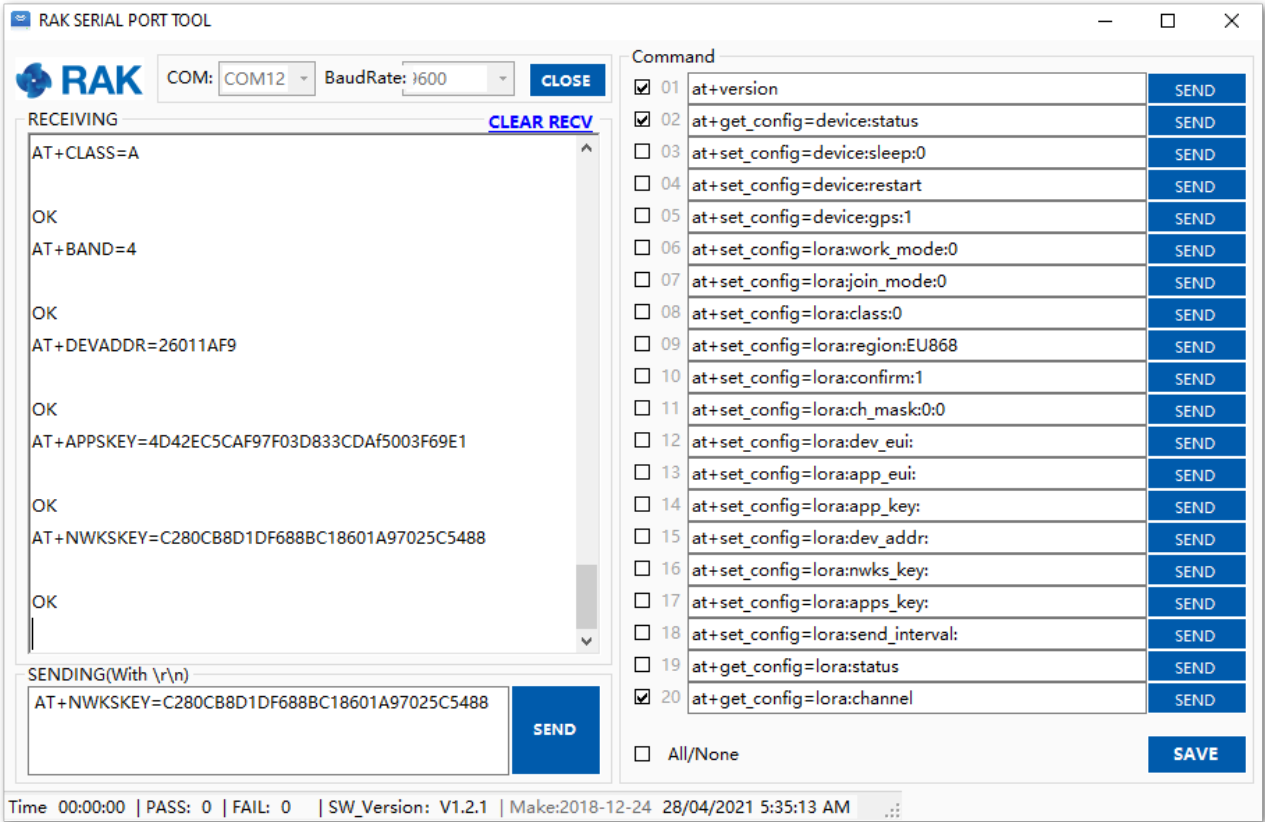


Figure 218: Configuring LoRa Parameters

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

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

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.

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

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

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

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

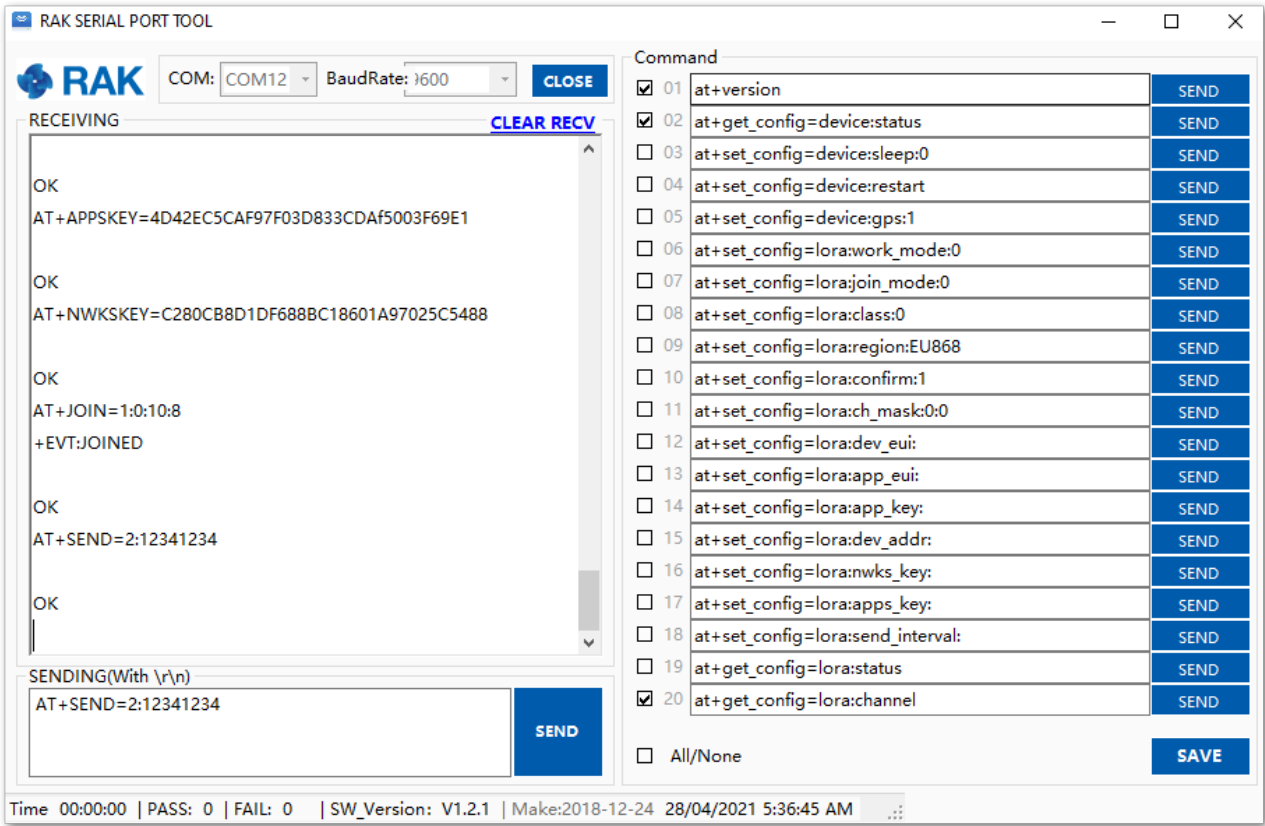


Figure 219: ABP Test Sample Data Sent via RAK Serial Port Tool

LoRa P2P Mode

This section will show you how to set up and connect two RAK3272S units to work in the LoRa P2P mode. The configuration of the RAK3272S units is done by connecting the two modules to a general-purpose computer using a USB-UART converter. The setup of each RAK3272S 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.

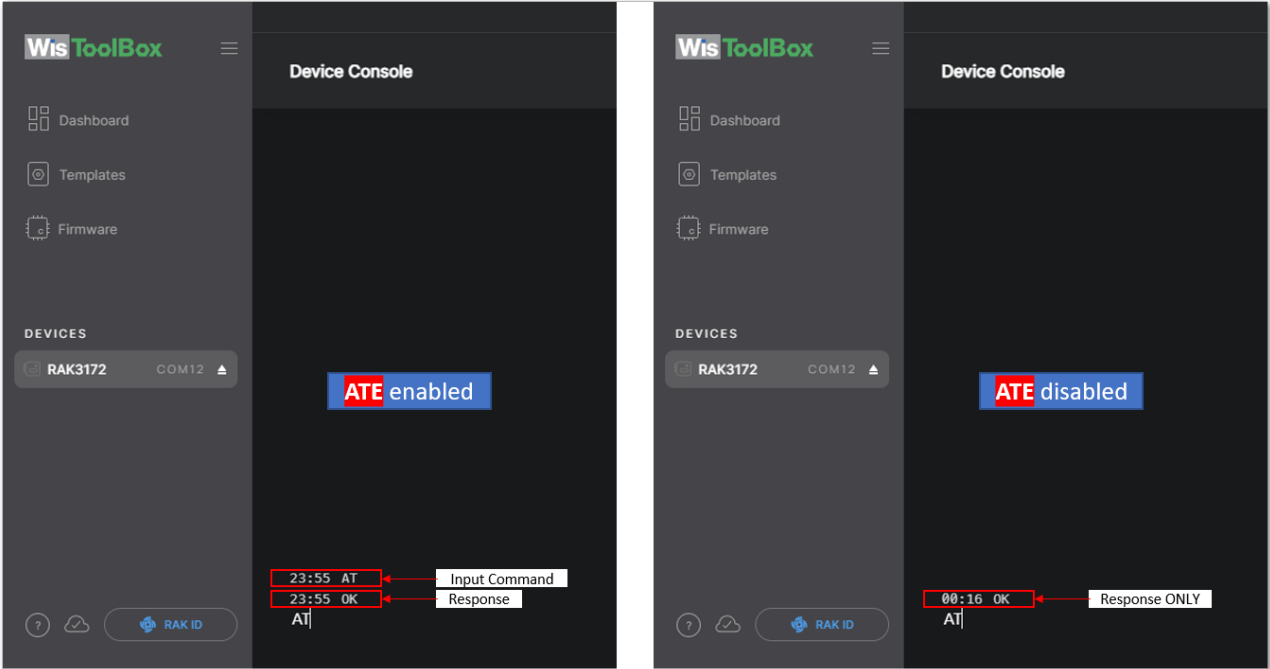


Figure 220: at+version command response

1. In setting up the RAK3272S to work in LoRa P2P mode, you need to change the LoRa network work mode command on both RAK3272S Breakout Board.

```
AT+NWM=0
```

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

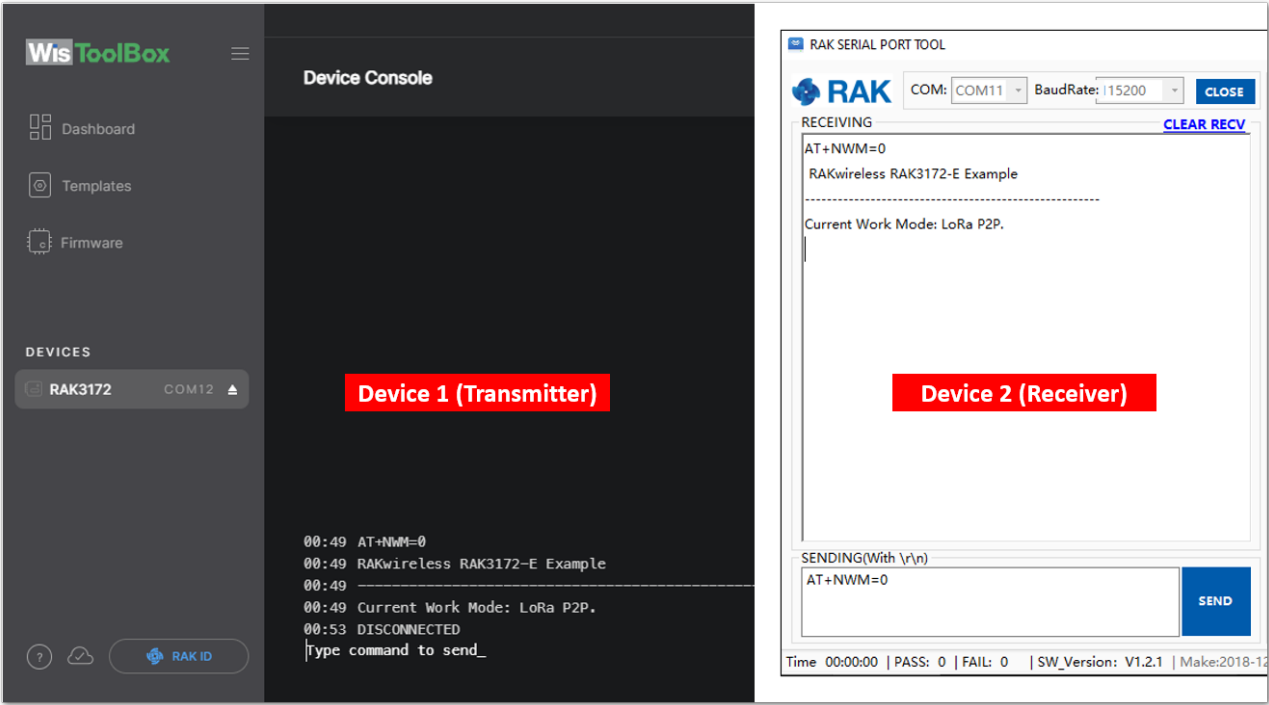


Figure 221: P2P Mode


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 RAK3272S Breakout Board. The parameters should be exactly the same on 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 parameter changed.

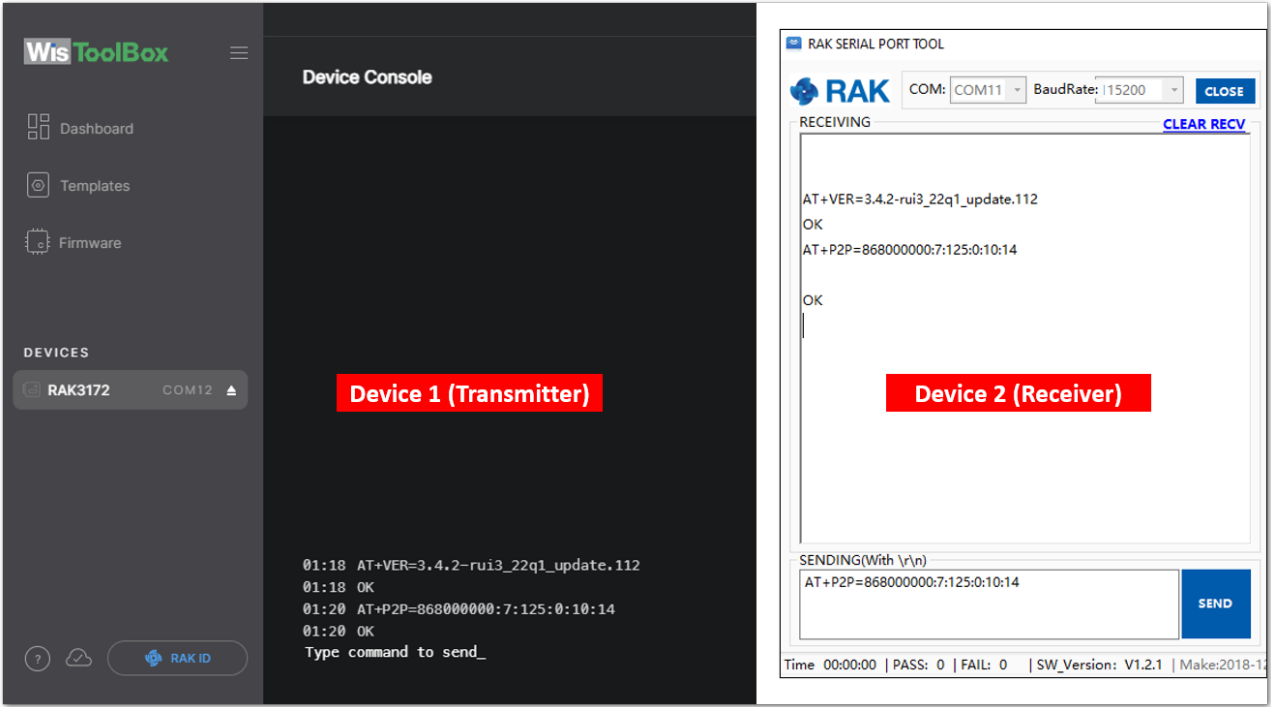


Figure 222: Configuring P2P in both RAK3272S Breakout Board

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 done 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. The 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 switch 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 wrong or malformed value.
- `AT_BUSY_ERROR` is returned if the device is still in RX mode and you try to send or reconfigure 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`, therefore the AT command should be `AT+PSEND = 03AA32`.

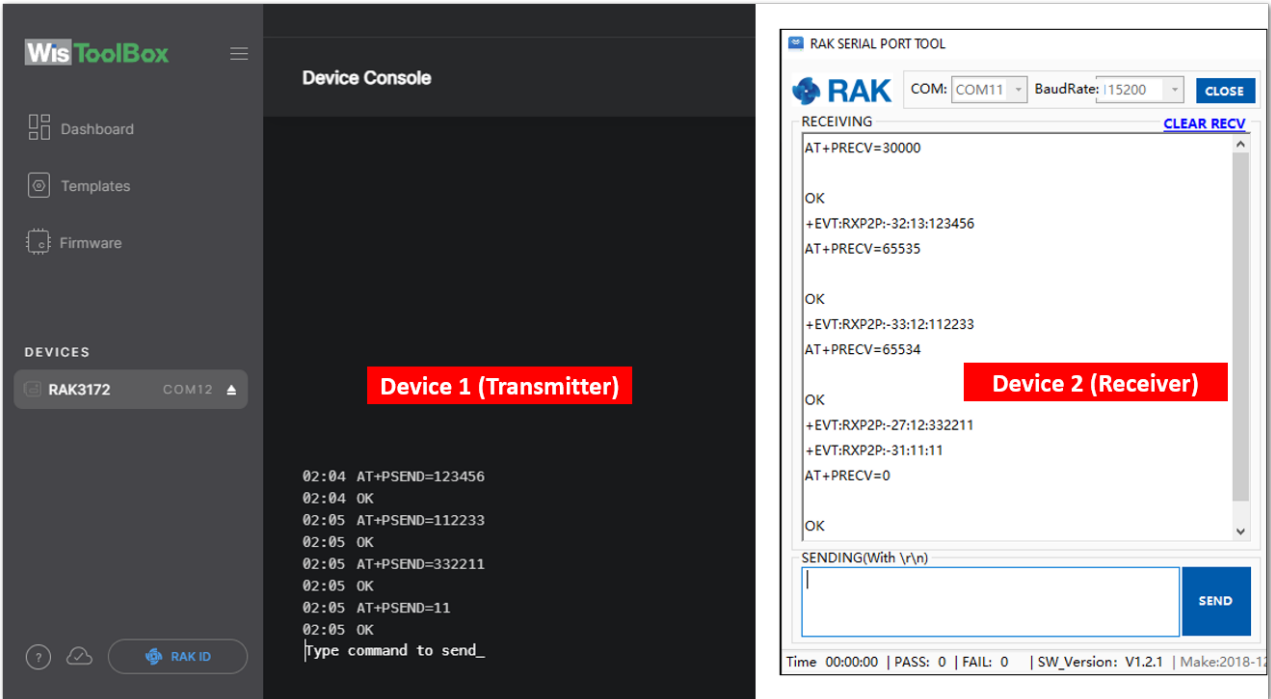


Figure 223: Configuring P2P in both RAK3272S Module

Miscellaneous

Upgrading the Firmware

If you want to upgrade to the latest version firmware of the module, you can follow this section. The latest firmware can be found in the software section of [RAK3272S Datasheet](#).

 NOTE:

What if the RAK3272S board stops responding to AT commands and firmware update?

You can recover your device by using the `.hex` file in the datasheet and upload it using STM32CubeProgrammer. The guide on updating STM32 firmware using STM32CubeProgrammer can be found in the [Knowledge Hub 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 USB interface.

 NOTE:

RAK3272S 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 RAK3272S.
 - [RAK3272S Firmware](#)
2. Download the RAK Device Firmware Upgrade (DFU) tool.
 - [RAK Device Firmware Upgrade \(DFU\) Tool](#)
3. Connect the RAK3272S Breakout Board to the computer via a USB-Serial adapter. Refer to [Figure 21](#).
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 fail, check your current baudrate setting using `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.

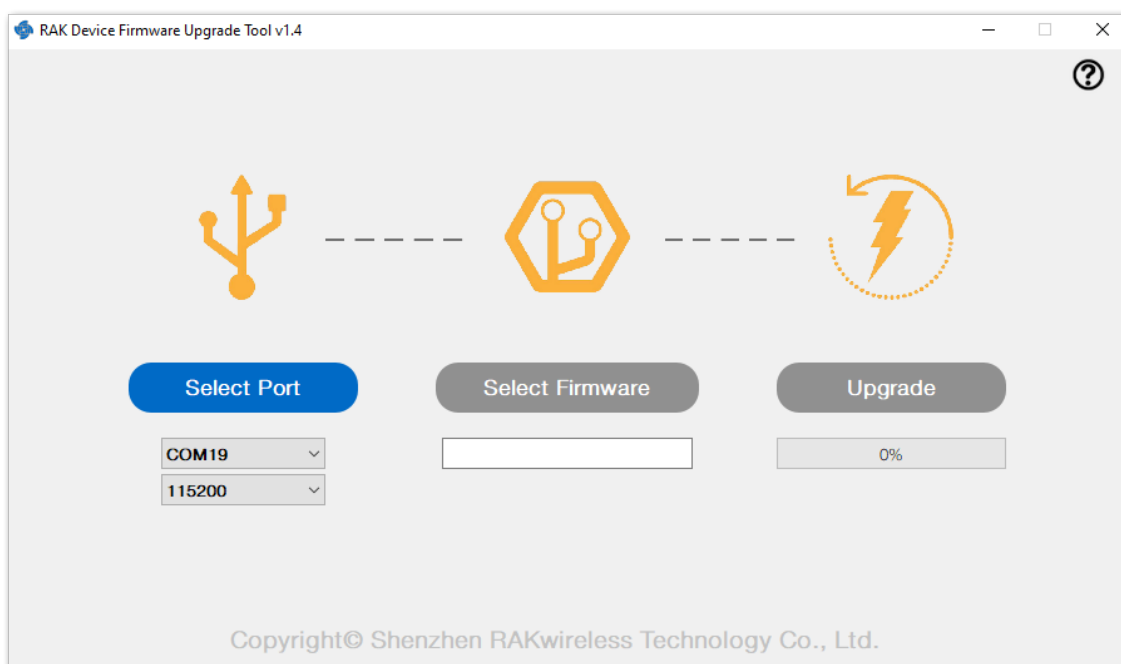


Figure 224: Device Firmware Upgrade Tool

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

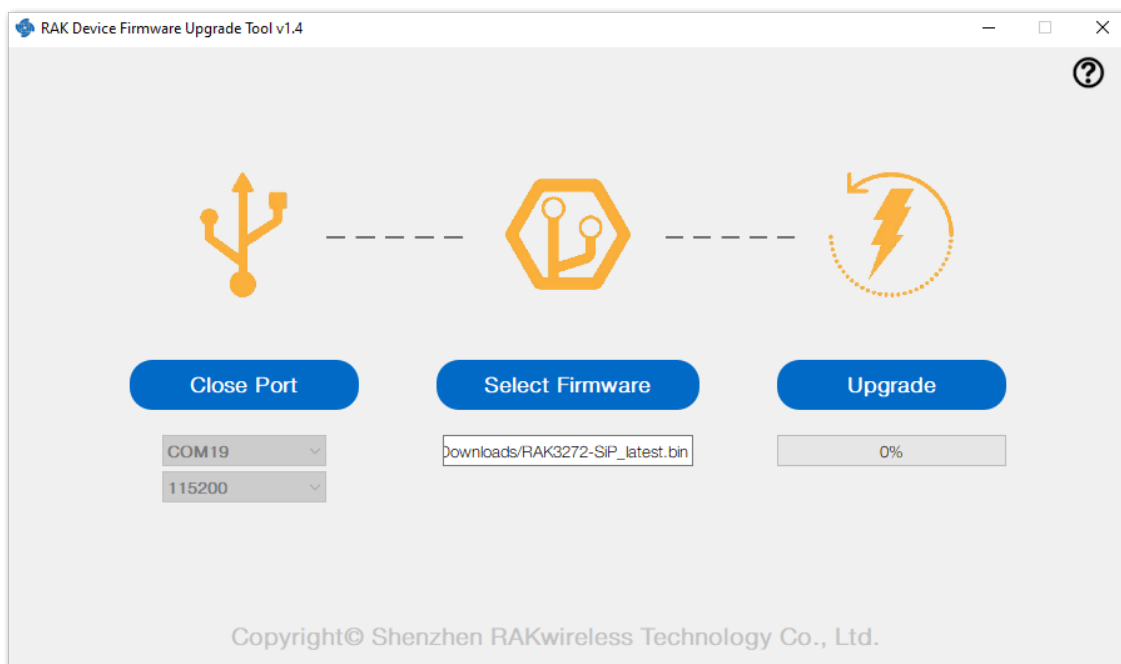


Figure 225: Select firmware

6. Click the "**Upgrade**" button to upgrade the device. After the upgrade is complete, the RAK3272S will be ready to work with the new firmware.

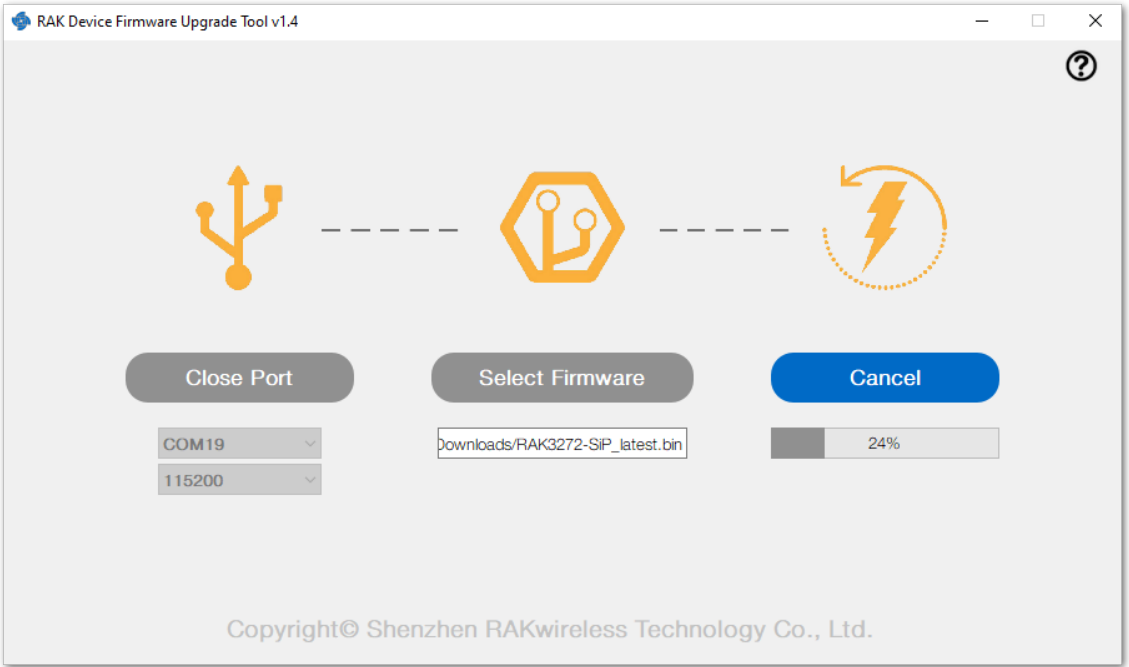


Figure 226: Firmware upgrading

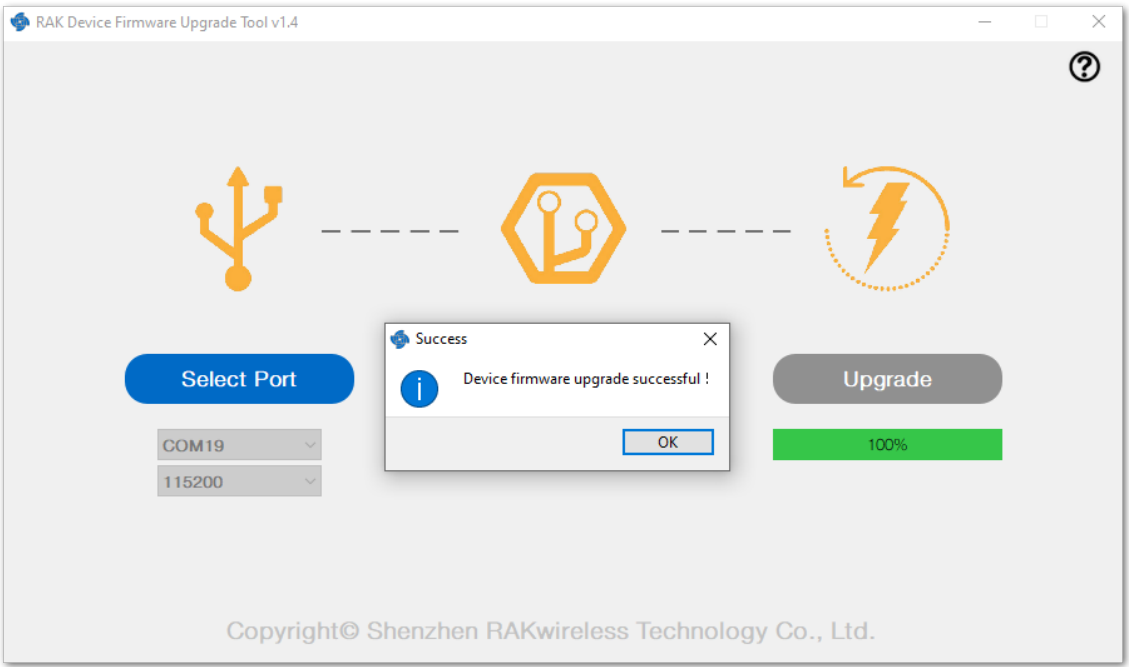


Figure 227: Upgrade successful

Arduino Installation

Go to the [Arduino official website](#) and download the Arduino IDE. You can see the multiple versions available for Windows, Linux, and Mac OS X. Choose the correct version of Arduino IDE and download it.



Figure 228: Arduino IDE latest version

For Windows

 **NOTE**

For Windows 10 users: Do **NOT** install the Arduino IDE from the Microsoft App store. Install the original Arduino IDE from the Arduino official website instead, since the Arduino app from the Microsoft App Store has problems using third-party Board Support Packages.

1. Install the Arduino IDE, which you just downloaded, on your Windows PC.
2. Click **I Agree** then **Next** to proceed.

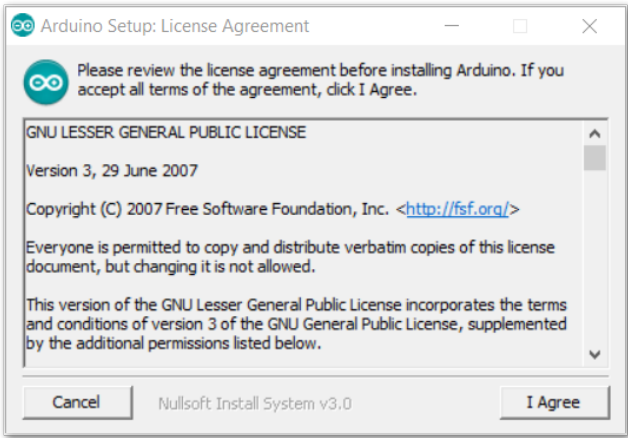


Figure 229: Arduino setup license agreement

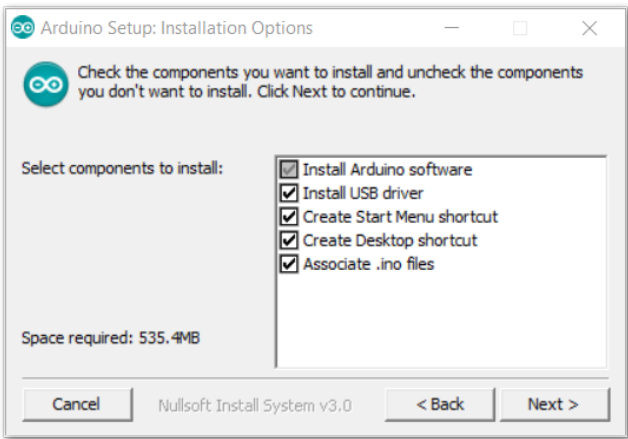


Figure 230: Arduino setup installation options

3. Click **Install**.

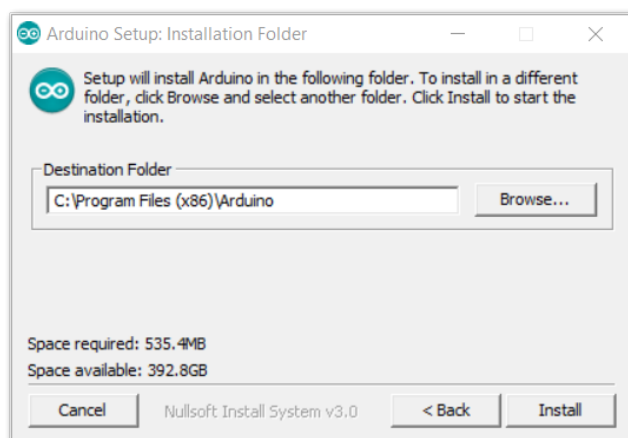


Figure 231: Installing Arduino IDE

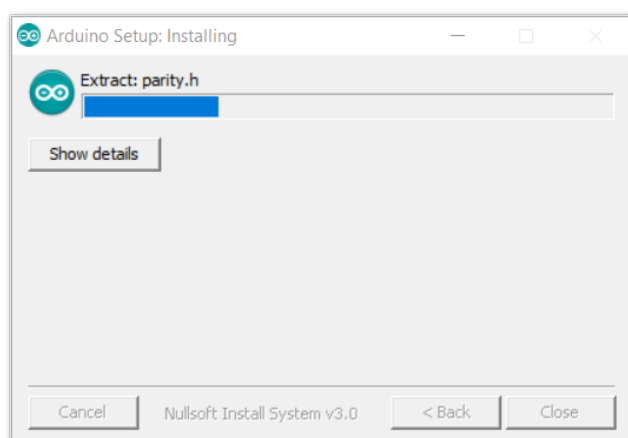


Figure 232: Ongoing installation

After 100% progress, the Arduino IDE has been installed successfully.

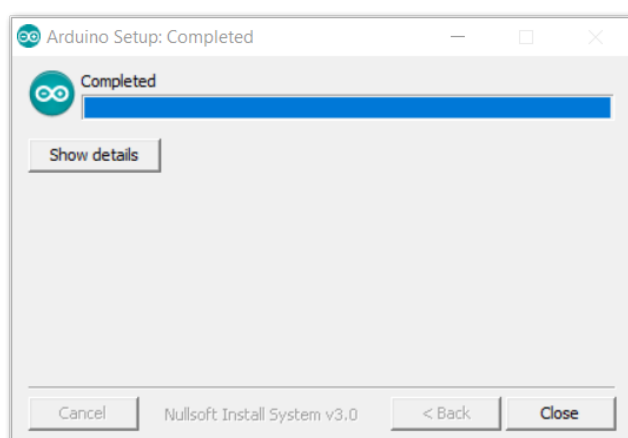


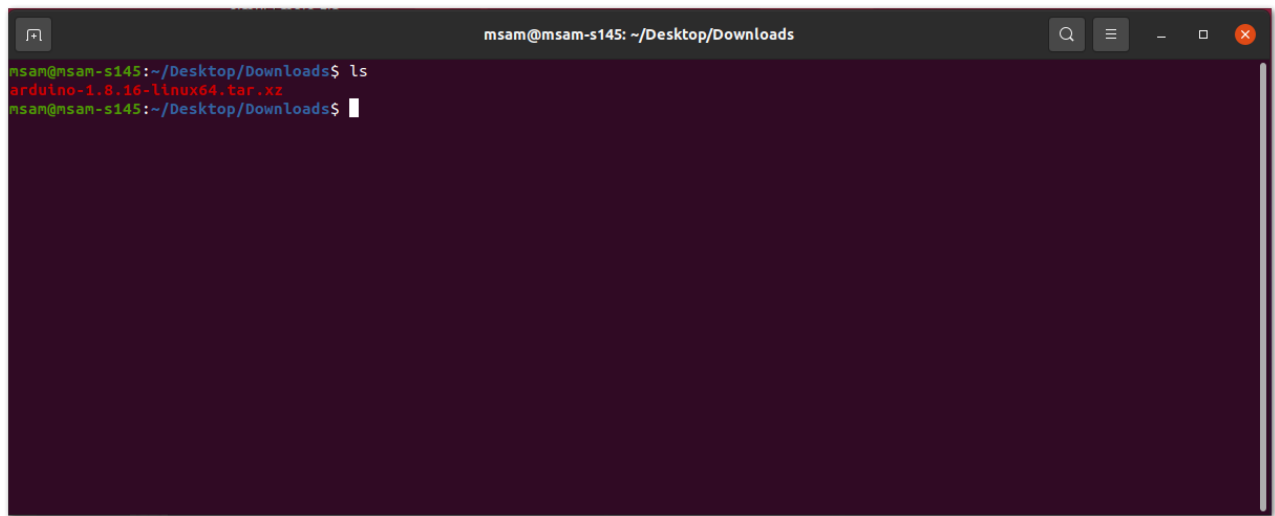
Figure 233: Successful installation

For Linux

First, you need to check the compatibility with your system and choose between the 32-bit, 64-bit, and ARM versions of the Arduino IDE for Linux.

Installing via a Tarball

After downloading the correct Arduino version, open a terminal, then run `ls` to check the installation file on the download folder.

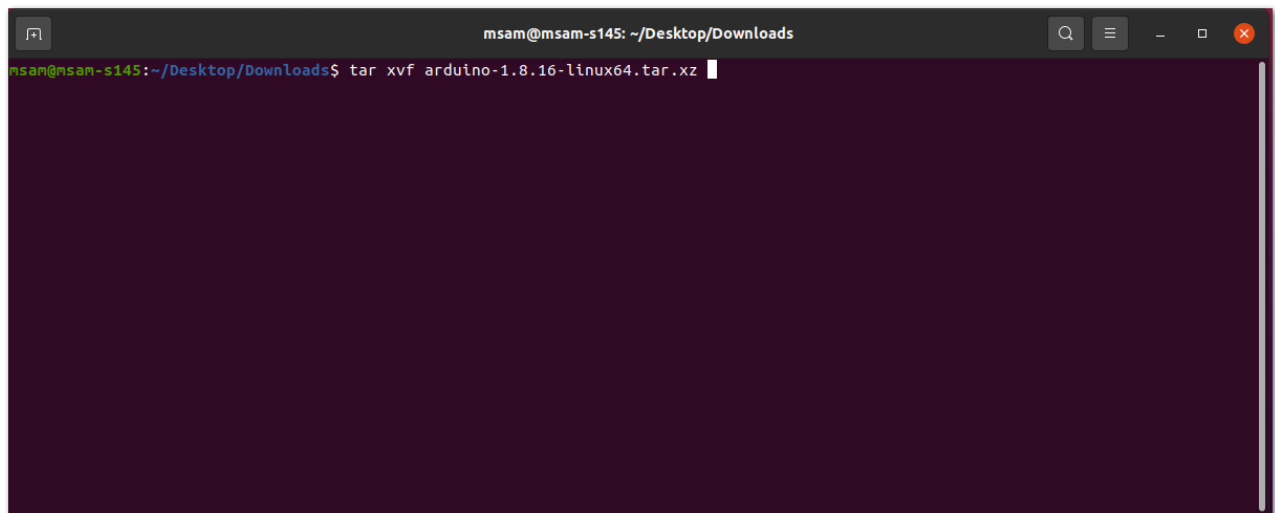


```
msam@msam-s145: ~/Desktop/Downloads
msam@msam-s145:~/Desktop/Downloads$ ls
arduino-1.8.16-linux64.tar.xz
msam@msam-s145:~/Desktop/Downloads$
```

Figure 234: Check the download folder

A tarball is a type of compressed folder, like a `.zip` file, commonly used to distribute software in Linux. To extract the files from the tarball, change the directory to where the downloaded tarball is, then run:

```
tar xvf arduino-version.xz
```



```
msam@msam-s145: ~/Desktop/Downloads
msam@msam-s145:~/Desktop/Downloads$ tar xvf arduino-1.8.16-linux64.tar.xz
```

Figure 235: Tarball extract command

When the tar command finishes, run `ls` again. A folder named **arduino-version** will be created.

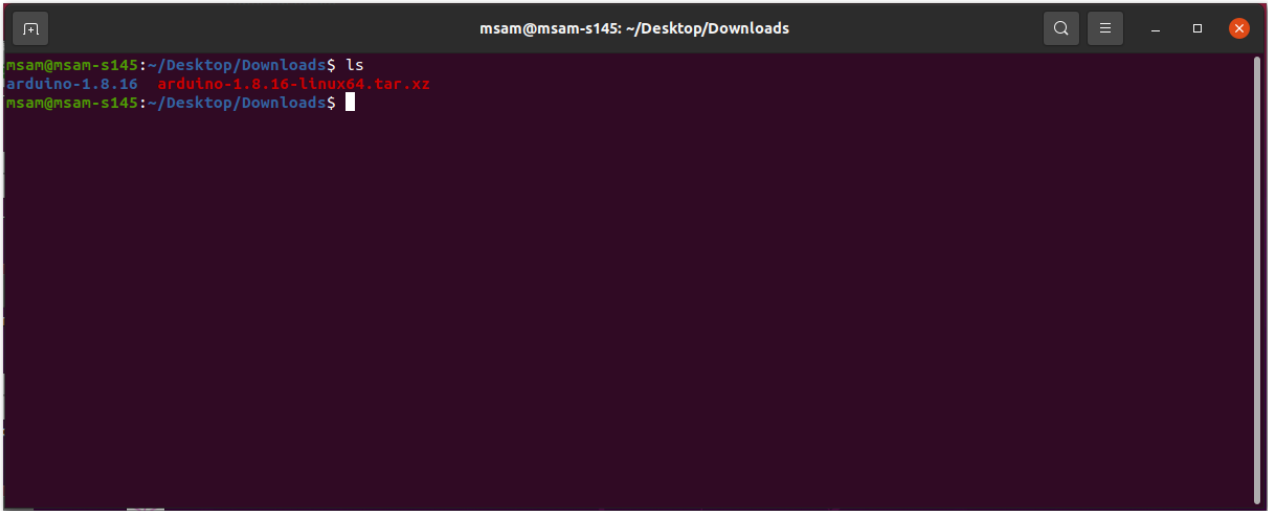


Figure 236: Arduino install folder created

Change the current directory and go to the newly created folder directory. There will be a file named `install.sh` in the folder. Execute `sudo ./install.sh` to install the Arduino IDE.

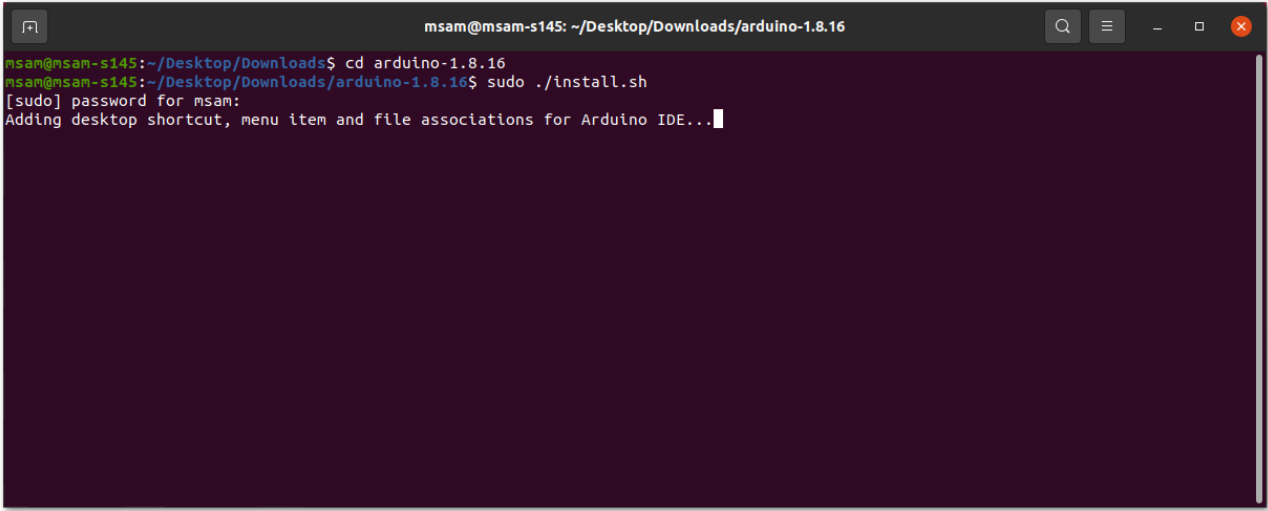


Figure 237: Arduino install script running

The `sudo` command temporarily elevates privileges allowing the installer to complete sensitive tasks without logging in as the root user.

For Mac OS X

In Mac OS X, the same as Linux, there is no installation process. It is just a process of decompression, then you can open Arduino IDE successfully.

Arduino IDE Parts Guide

Figure 238 shows the five (5) parts of Arduino IDE.

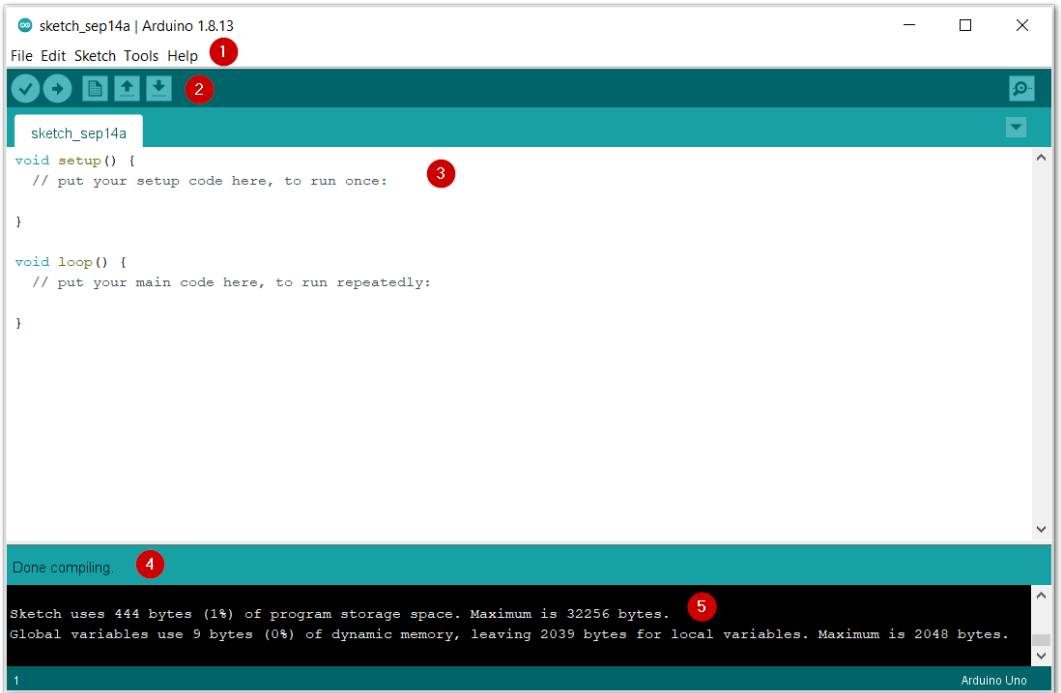


Figure 238: Arduino IDE

1. IDE Option Menu

You can configure some general parameters such as the serial port, the board information, the libraries, the edit parameters, and so on.

2. Operating Buttons

The operating buttons have five operations:

- **Verify/Compile** the source code;
- **Upload** the compiled code into WisBlock;
- **Open** a **New** Arduino IDE window or existing application;
- **Save** the current application.

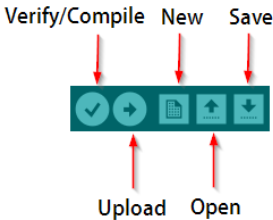


Figure 239: Operating buttons

3. Code Area

You can edit the source code, which will be compiled and uploaded into WisBlock later in this area.

4. State Area

5. **Output Message Area** You can see the output message in this area, whether it's failure or success information.