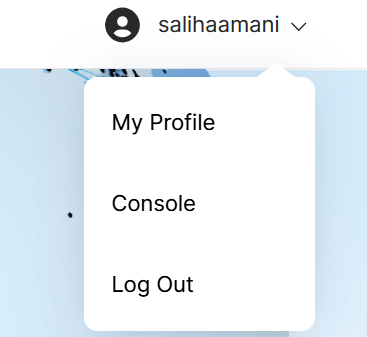
**Heltec HTCC-AB01 (V2) and WiFi LoRa 32 (V3) TTN Tutorial**

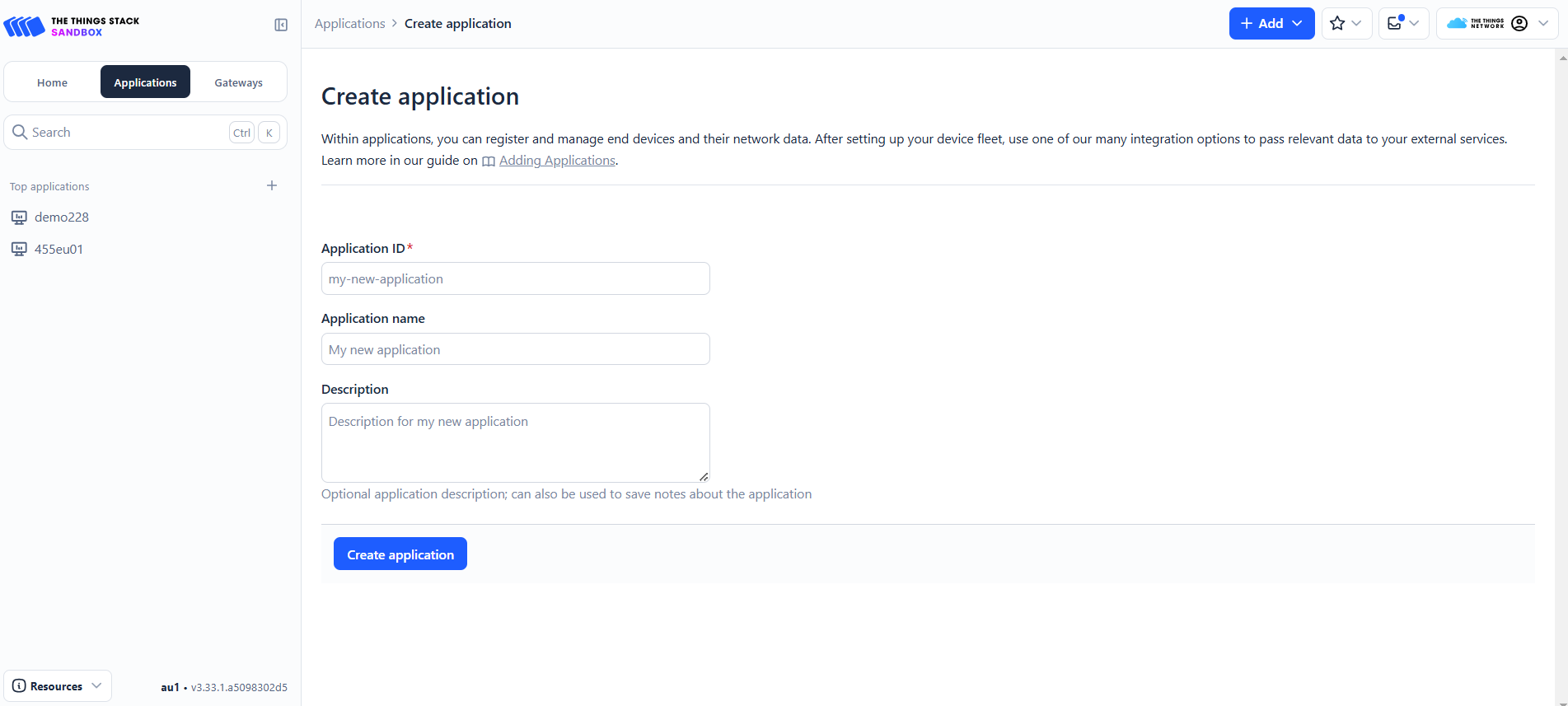
1. **Sign up at The Things Network**

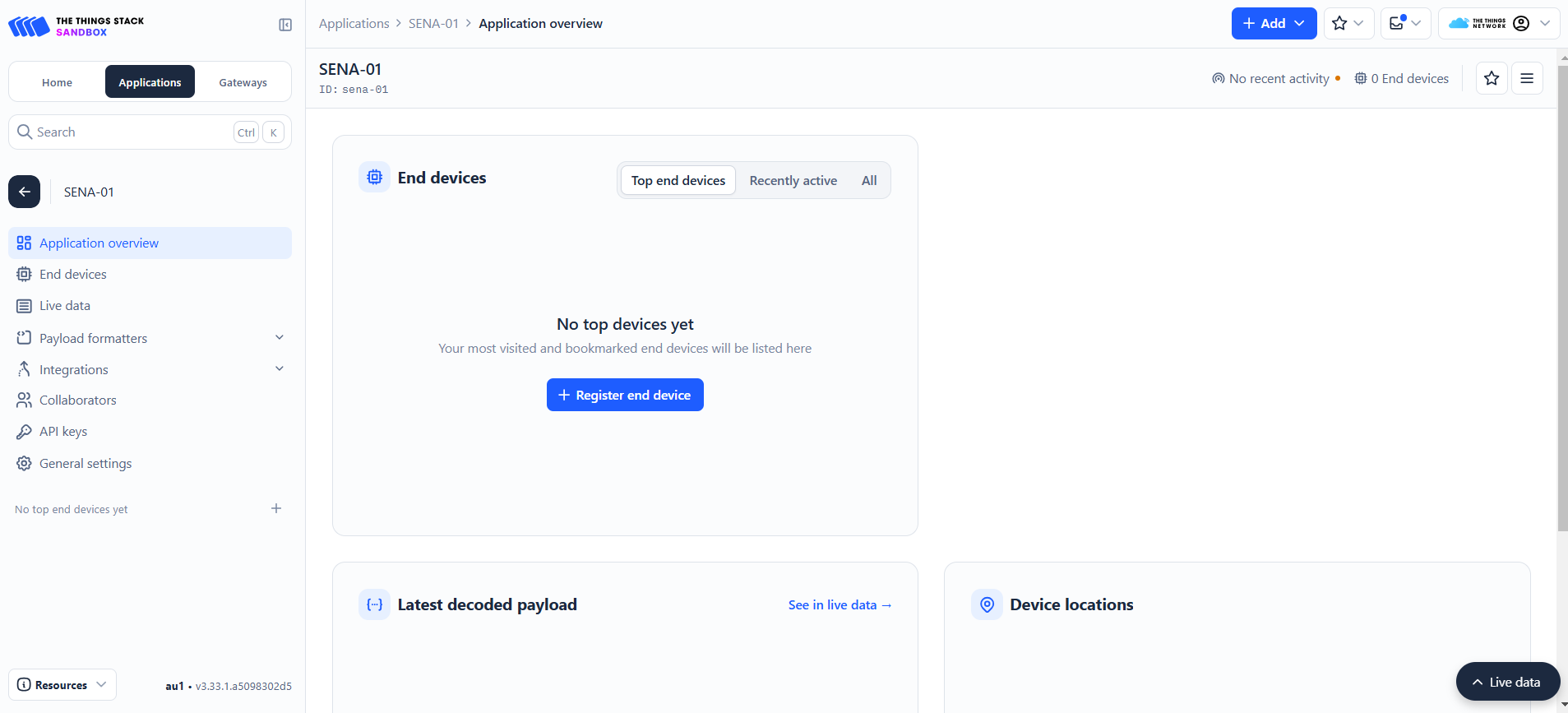
* Go to <https://www.thethingsnetwork.org/>, **sign up** -> **Join The Things Network**
* Finish sign up, select **Console** -> **Choose network cluster** -> **Australia 1 (recommended)**



1. **Adding application**

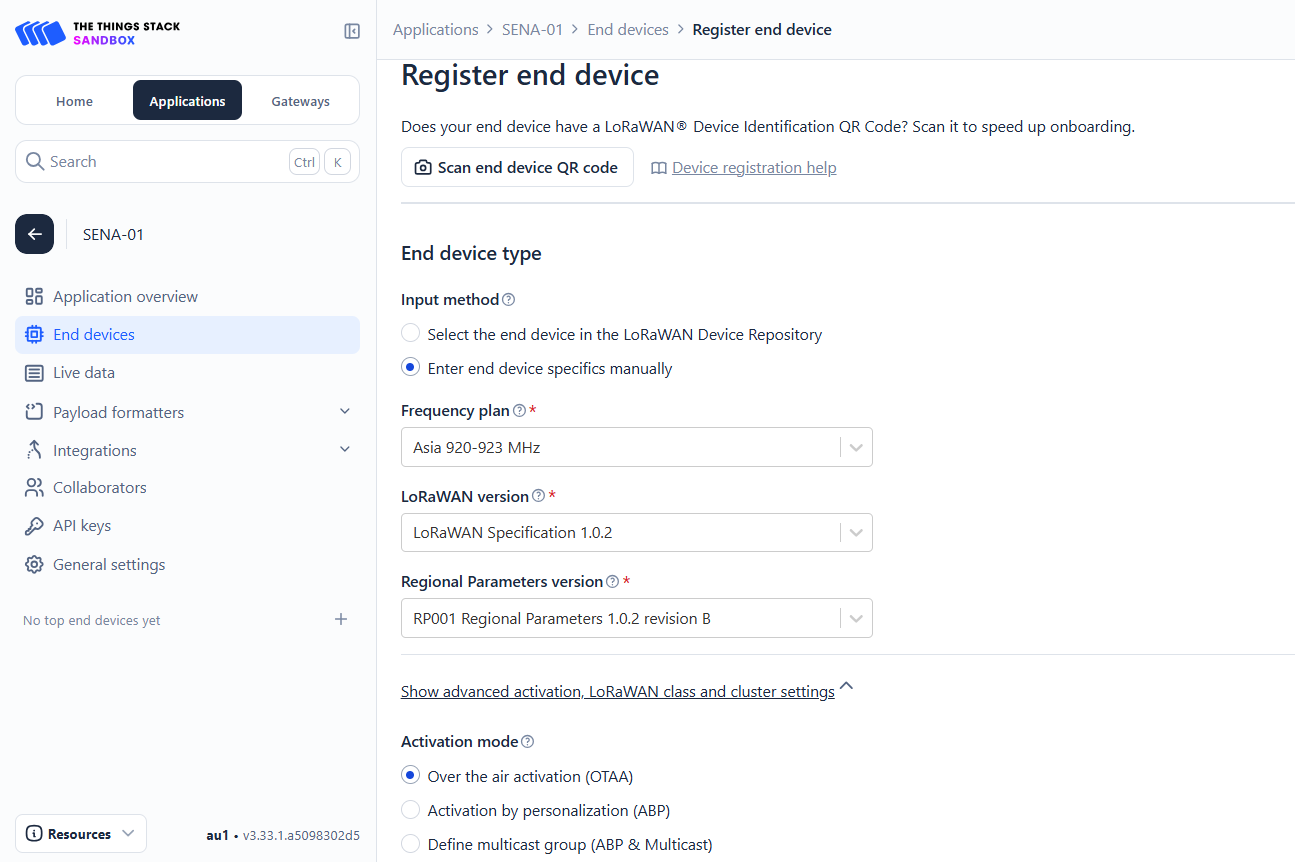
* An Application in TTN is a collection of end devices.
* Go to **Applications** in the top menu, and click **+ Add Application** to reach the application registration page. Fill the fields and click **Create Application** to create the application.

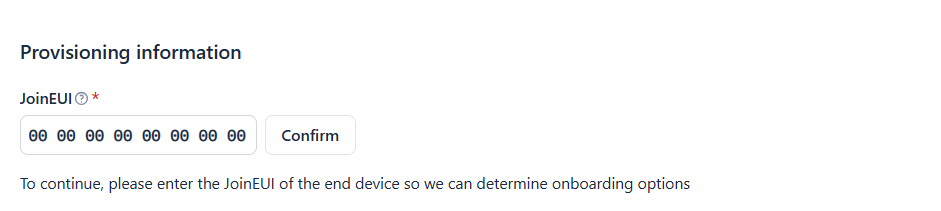




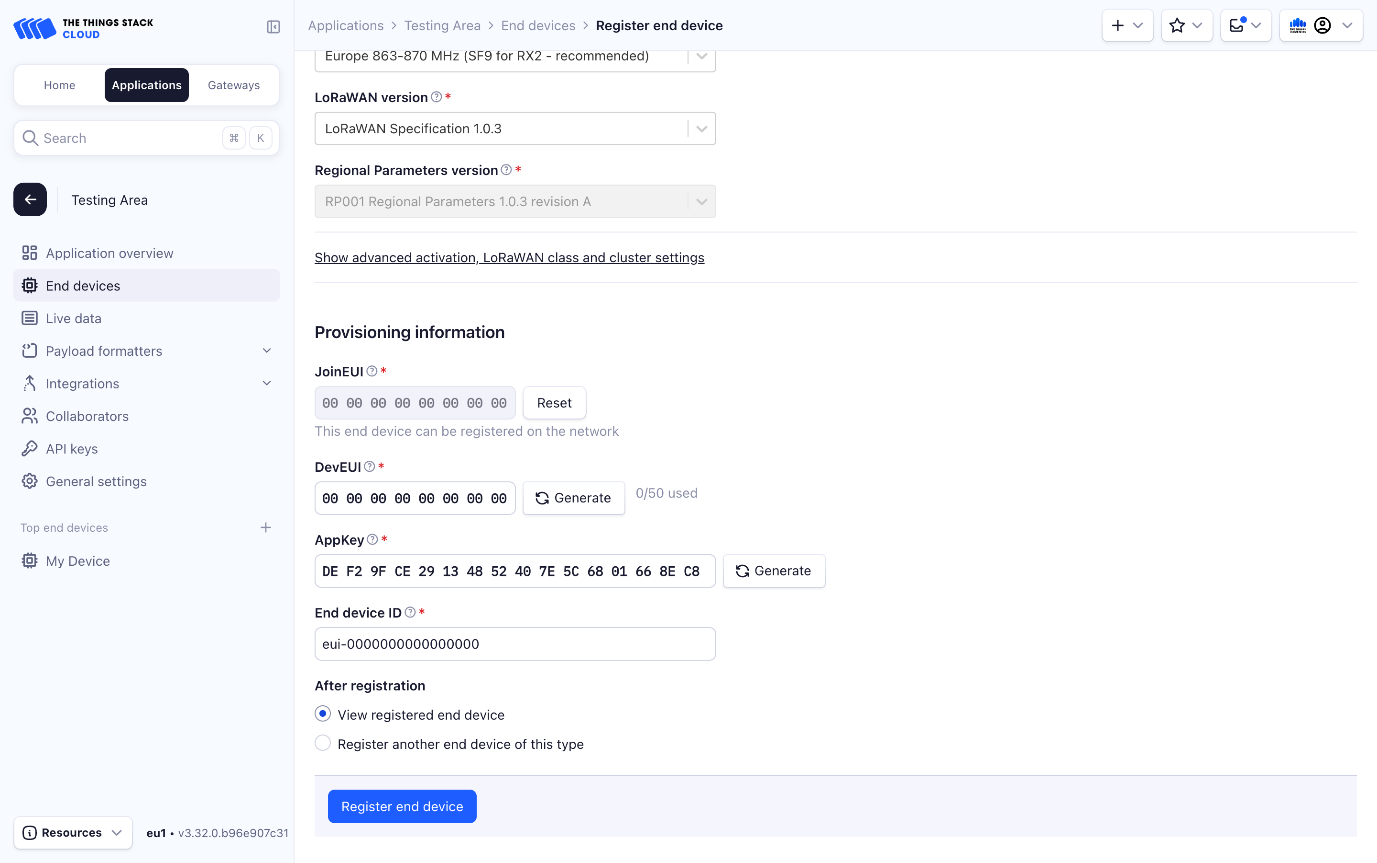
1. **Adding end device**

* To create a device, open the application you wish to add the device in. Go to **End devices** in the left menu and click on **+Register end device** to reach the end device registration page.
* **End device type**-> tick **Enter end device specifics manually** for the input method.
  + **Frequency plan** -> **Asia 923-925MHz**. Your device and gateway must use the same frequency plan to communicate.
  + **LoRaWAN version** -> **LoRaWAN Specification 1.0.2**
  + **Regional Parameters version** -> **RP001 Regional Parameters 1.0.2 revision B**
  + Show advanced activation, LoRaWAN class and cluster settings, tick on the **Over the air activation (OTAA)** for the activation mode.
  + JoinEUI -> all zeros





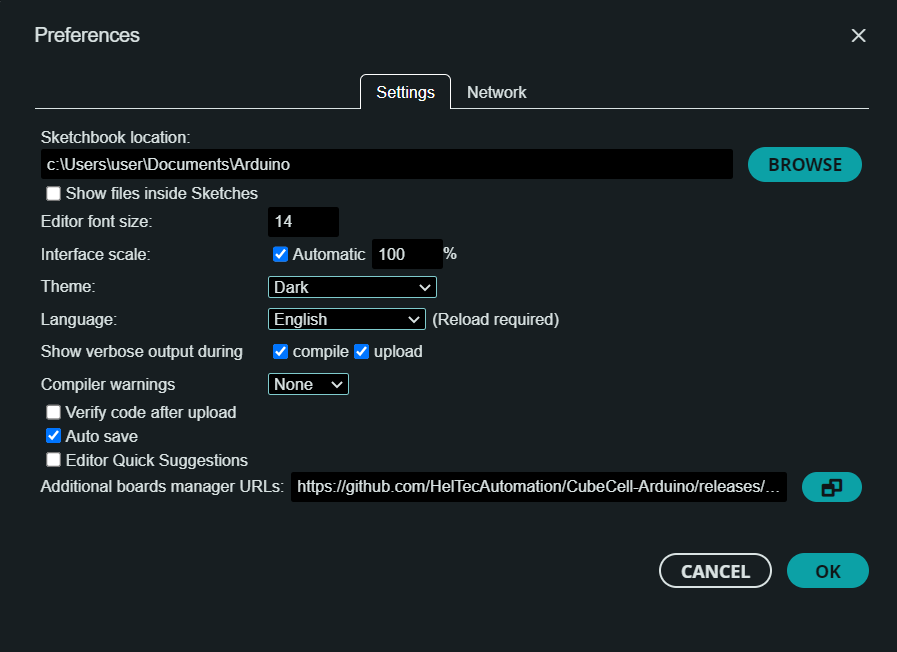
* Choose **Over the air activation (OTAA)**
* Since the Heltec HTCC-AB01 and WiFi LoRa 32 (V3) are programmable, the **EUI** and **AppKey** can be generated using the **Generate button**, and can be programmed to the device.
* Give your device a unique **End device ID**
* After configuring your device, select the **Register end device** button. Next, proceed to Arduino IDE configuration.



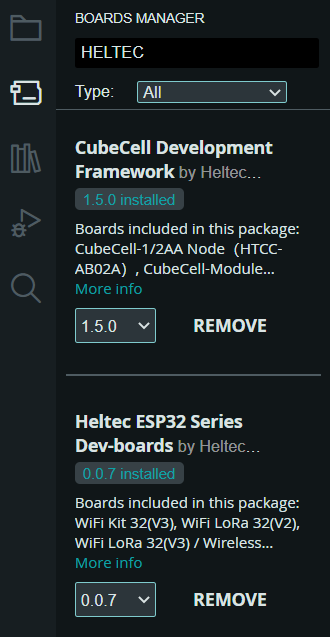
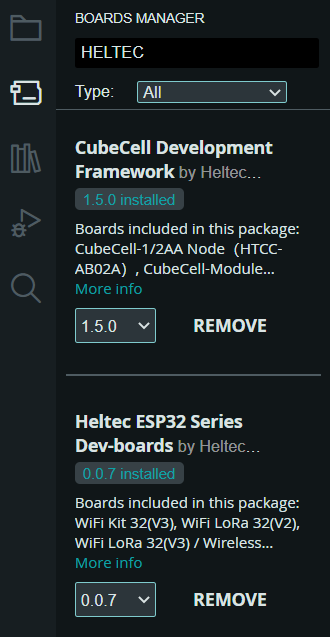
**Heltec HTCC-AB01 (V2) and WiFi LoRa 32 (V3) Arduino IDE Tutorial**

1. Download the latest release of the [Arduino Software (IDE)](https://www.arduino.cc/en/Main/Software) at <https://www.arduino.cc/en/software/> , [install](https://www.arduino.cc/en/Guide) it on your operating system and run it.
2. Open Arduino IDE, and click **File**->**Preferences**->**Settings** and click the button next to **Additional Boards Manager URLs**.

* **For** **HTCC-AB01 (v2):** <https://github.com/HelTecAutomation/CubeCell-Arduino/releases/download/V1.5.0/package_CubeCell_index.json>
* **For Wifi LoRa32 (v3) :** <https://github.com/Heltec-Aaron-Lee/WiFi_Kit_series/releases/download/0.0.7/package_heltec_esp32_index.json>



1. Select **Tools**->**Board**->**Boards Manager**, then search **Heltec** in the search field, select the latest releases for HTCC-AB01 or LoRa32 and click **Install**.

1. Connect the board to your computer. Go to the **Tools** menu and select the right COM port and board. In the **Tools** menu, there will be some options need to be selected.

* LORAWAN\_CLASS – **Class A**
* LORAWAN\_DEVEUI – **CUSTOM**
* LORAWAN\_NETMODE – **OTAA**
* LORAWAN\_REGION – **AS923 (AS1)**

1. Copy the **devEui**, **appEui**, and **appKey** from the TTN to be pasted into the flood\_monitoring-code.ino. The .ino code for flood monitoring can be downloaded here:

<https://github.com/amani-ibrahim/sena-github/blob/49e47a400a731522054feeac9c368d10848d3b0e/flood_monitoring-code/flood_monitoring-code.ino>.

Paste the **devEui**, **app Eui**, and **appKey** to the **OTAA parameters**. Upload the code to the device.

/\* OTAA para\*/

uint8\_t devEui[] = { 0x22, 0x32, 0x33, 0x00, 0x00, 0x88, 0x88, 0x02 };

uint8\_t appEui[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };

uint8\_t appKey[] = { 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x66, 0x01 };

1. In TTN website, in the **Application** -> **End Devices** -> **Payload Formatters** -> **Formatter Type** -> **Custom Javascript Formatter**. Change the payload formatter according to data received from the End devices. The custom javascript for flood monitoring can be copied here:

<https://github.com/amani-ibrahim/sena-github/blob/49e47a400a731522054feeac9c368d10848d3b0e/flood_monitoring-code/Payload-formatter.js>

