

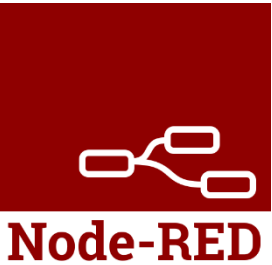
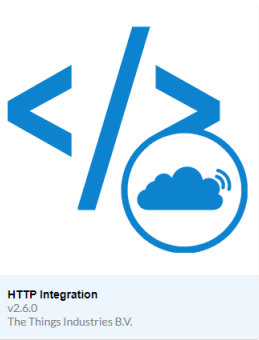
# LoRaWan

## Deploy LoRaWAN Application Using UCA Education Board

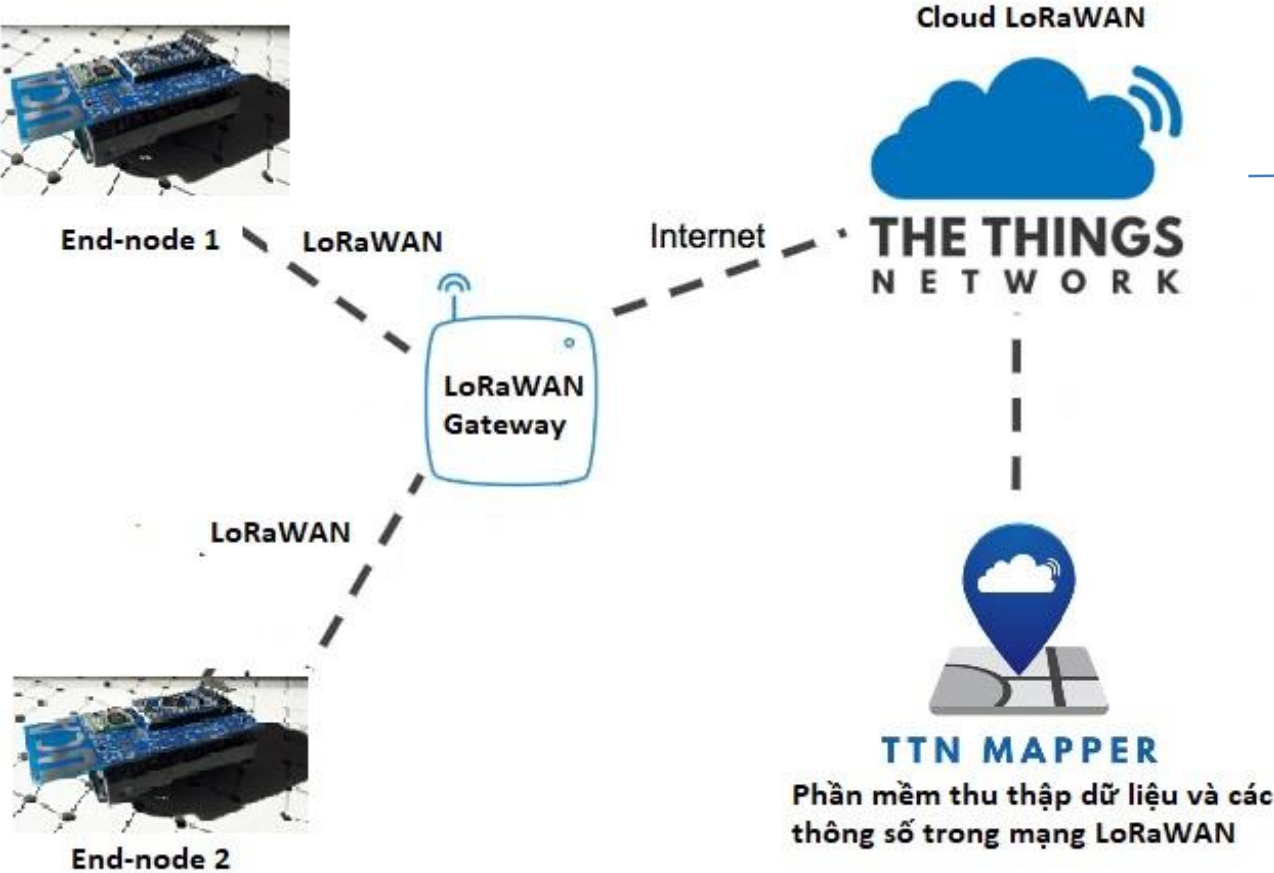
Tran Van Lic, F. Ferrero

Hue, 03/2021

# LoRaWAN Architecture



Application Server



# UCA Education Board



RFM95 Module

## Specifications:

Power:DC 1.8V ~ 3.7V

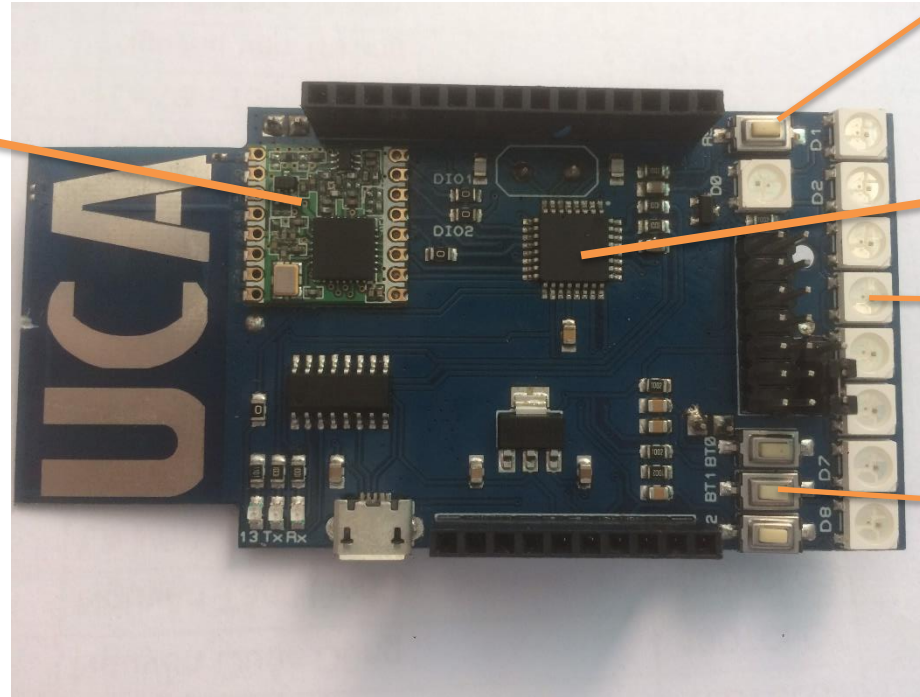
Transmit power: 20dBm

Working frequency: 868/915

Modulation mode: LoRA/FSK/GFSK/00K MHz

High sensitivity: down to -148 dBm.

Data Rate:300kbps



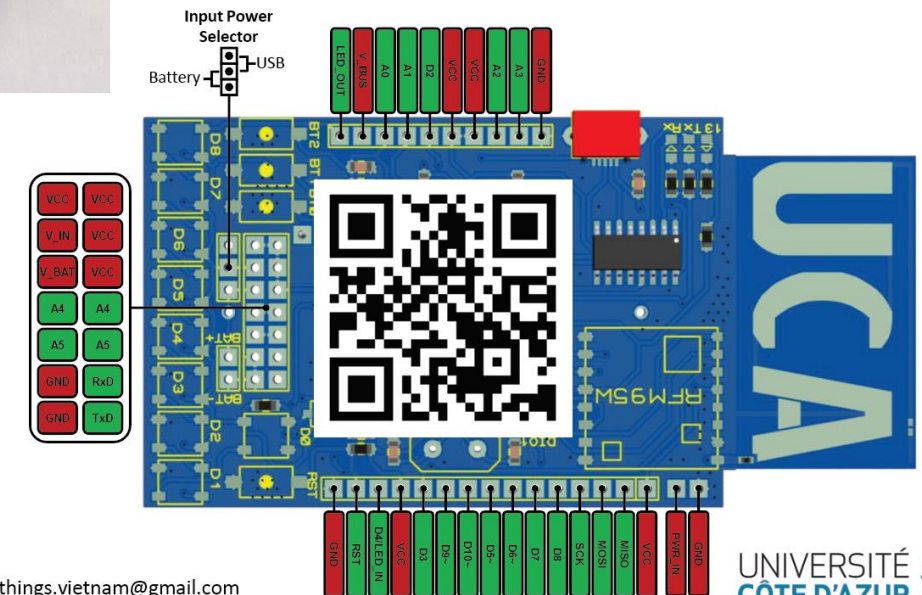
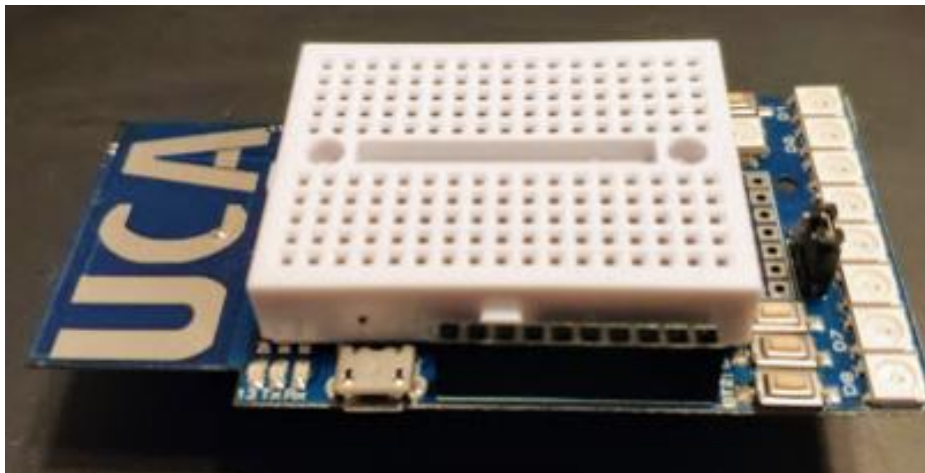
RESET button



Chip AT MEGA 328P

RGB LED

Button 1,2,3



# Preparing

- Software

- Arduino IDE: <https://www.arduino.cc/en/software>
- Codes are available on : [https://github.com/FabienFerrero/UCA\\_Education\\_Board](https://github.com/FabienFerrero/UCA_Education_Board)
- The board is using CH340C chip for USB to You may need to install the driver to use <https://sparks.gogo.co.nz/ch340.html>

- Hardware

- UCA Education Board

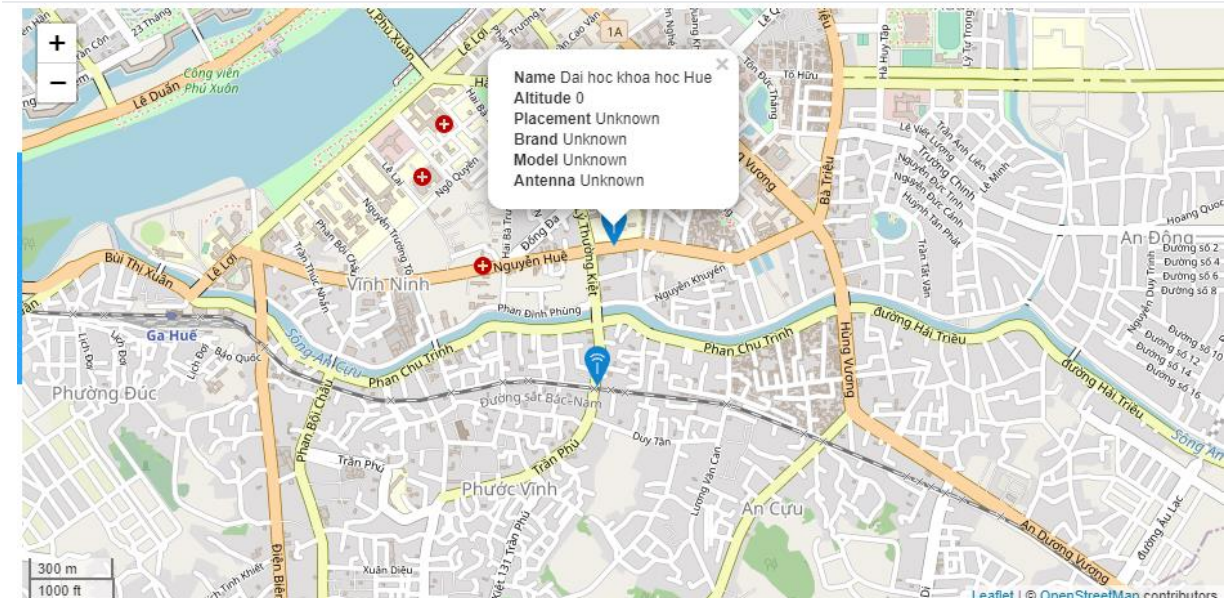




# LoRaWan with The Thing Network

At this moment, there are 17541 gateways up and running

- The Things Network is a **global**, open, crowd-sourced Internet of Things data network.
- **The Things Network Backend** route messages from Nodes to the right Application, and back
- TTN is free
- 17541 LoRa gateways are connected to TTN around the world
- Any TTN can use any GWs, it is a collaborative network



# Create a TTN account

- First, you have to [register](https://www.thethingsnetwork.org/) to <https://www.thethingsnetwork.org/>
- Then, give me your USERNAME, I will add you as a collaborator in our application – hue\_lora\_course 2021

Applications > hue\_lora\_course\_2021

Overview Devices Payload Formats Integrations Data Settings

### APPLICATION OVERVIEW

**Application ID** hue\_lora\_course\_2021 [documentation](#)

**Description**

**Created** 22 seconds ago

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

### APPLICATION EUIs

[manage euis](#)

<> 70 B3 D5 7E D0 03 FB 99

## CREATE AN ACCOUNT

Create an account for The Things Network and start exploring the world of Internet of Things with us.

### USERNAME

This will be your username — pick a good one because you will not be able to change it.



### EMAIL ADDRESS

You will receive a confirmation email, as well as occasional account related emails. If this email address is managed by a third party (such as for corporate email addresses), this third party might block emails coming from The Things Network. This email address is not public.



### PASSWORD

Use at least 6 characters.



### NEWSLETTER

Subscribe to the newsletter.



Create account

# Adding a new device

- Go to « **application** » and choose the available application – **hue\_lora\_course\_2021**
- Click on « **register device** »
- In DEVICE OVERVIEW, you get usefull information on your device. Of course, status is : « Never Seen »
- First, we will connect using ABP (Activation by Personalization )

Applications > hue\_lora\_course\_2021 > Devices

Overview **Devices** Payload Formats Integrations Data Settings

### REGISTER DEVICE

[bulk import devices](#)

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

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

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

Applications > hue\_lora\_course\_2021 > Devices > test\_registor\_device > Settings

Overview Data **Settings**

### DEVICE SETTINGS

General  
Location

### SETTINGS

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

**Device EUI**  
The serial number of your radio module, similar to a MAC address

**Application EUI**

**Activation Method**

OTAA ABP

Applications > hue\_lora\_course\_2021 > Devices > test\_registor\_device

### DEVICE OVERVIEW

**Application ID** hue\_lora\_course\_2021

**Device ID** test\_registor\_device

**Activation Method** OTAA

**Device EUI** <> 00 6F 83 61 91 18 DE E8

**Application EUI** <> 70 B3 D5 7E D0 03 FB 99

**App Key** <> .....

**Status** ● never seen

**Frames up** 0 [reset frame counters](#)

**Frames down** 0

# Downloading Arduino code on Github

- For this tutorial, you are going to use Arduino codes
- Codes are available on :  
[https://github.com/FabienFerrero/UCA\\_Education\\_Board](https://github.com/FabienFerrero/UCA_Education_Board)
- You can click on “Clone or Dowload” and “download zip”
- Then unzip it
- If you are using Github Desktop, you can use “open in Desktop”

Board for Education with LoRa

12 commits   1 branch   0 releases   1 contributor   MIT

Branch: master   New pull request   Find File   Clone or download

FabienFerrero add code   Latest commit ad15230 2 days ago

Antenna	code update	last month
Code	add code	2 days ago
Libraries	LED update	last month
Schematic	code update	last month
LICENSE	Initial commit	last month
README.md	Update README.md	last month

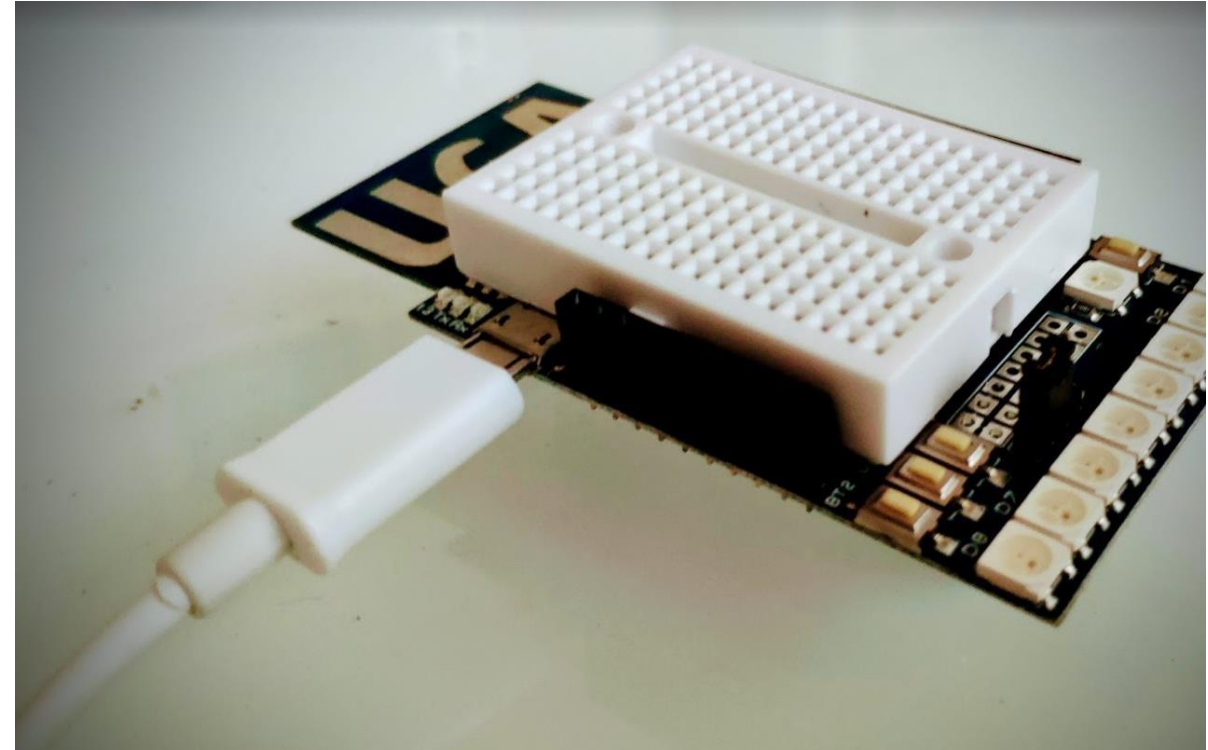


# Configuring your Arduino IDE

- After downloading the archive (.zip) and extracting the archive
- Copy the file from UCA\_Education\_Board\Libraries to /Document/Arduino/ Libraries/

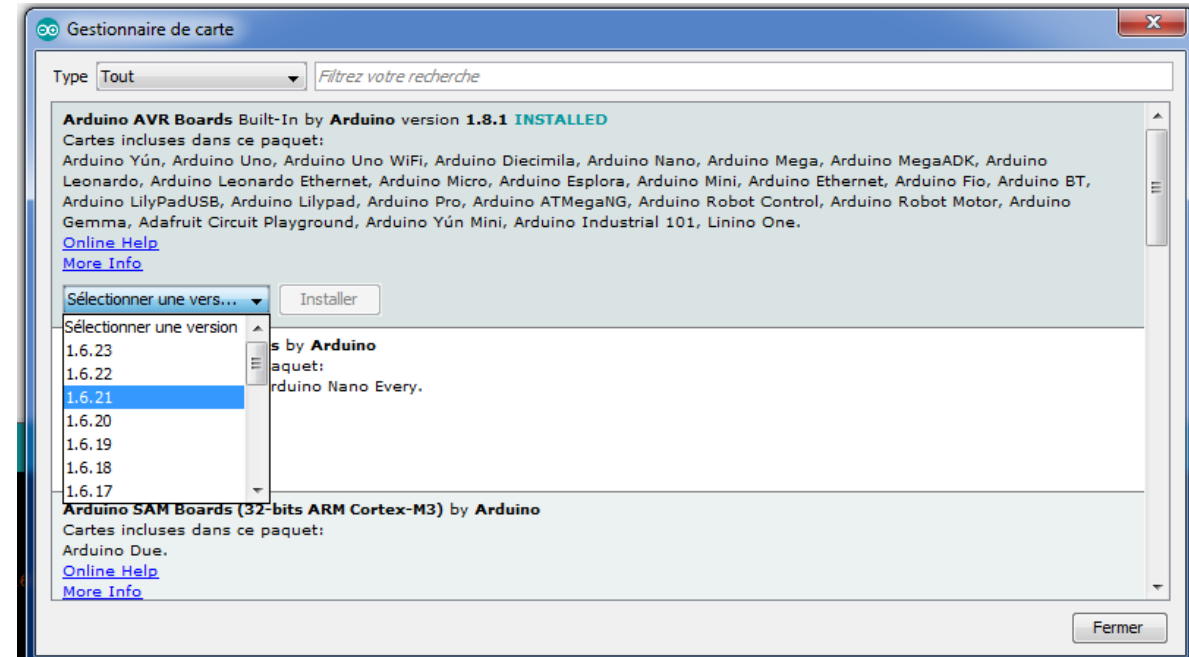
It will install the libs needed during the tutorial

- If your using Windows or Mac, your may need to install the board USB driver (CH340C) : drivers are available [here](#)
- If Arduino IDE select in Tools (Outils)
  - Board : Arduino Pro or Pro Mini
  - Processor : AT328p 8MHz 3.3V
  - Port : Select your serial port



# Configuring your Arduino IDE

- **!Important!** We have an unsolved bug in Arduino
- To avoid it, you have to go to:
  - Tools\Board\Boards Manager
  - In Arduino AVR Boards, select **1.6.21** version and **Install**
  - You will probably need to close and re-open Arduino IDE



## [UCA Education Board/Code/LORAWAN/ABP/Basic/UCA-ABP\\_Basic/](#)

```

/*****
// Region definition (will change de frequency bands
// Define only 1 country
//
// #define CFG_EU 1
#define CFG_VN 1

*****/

#include <lmic.h>
#include <hal/hal.h>
#include <SPI.h>

// LoRaWAN end-device address (DevAddr)

static const u4_t DEVADDR = 0x00000000;

// LoRaWAN NwkSKey, network session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const PROGMEM u1_t NWKSKEY[16] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };

// LoRaWAN AppSKey, application session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const u1_t PROGMEM APPSKEY[16] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };

```

The diagram consists of four orange arrows pointing from specific lines of code to labels on the right. The first arrow points from the line `#define CFG_VN 1` to the label "Switch to frequency plan in Vietnam". The second arrow points from the line `static const u4_t DEVADDR = 0x00000000;` to the label "Device address". The third arrow points from the line `static const PROGMEM u1_t NWKSKEY[16] = { ... }` to the label "Network Sesstion Key". The fourth arrow points from the line `static const u1_t PROGMEM APPSKEY[16] = { ... }` to the label "App Sesstion Key".

Switch to frequency plan in Vietnam

Device address

Network Sesstion Key

App Sesstion Key

```

#if defined(CFG_EU)
// Set up the 8 channels used
LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENTI); // g-band
LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK, DR_FSK), BAND_MILLI); // g2-band

#elif defined(CFG_VN)
// Set up the 8 channels used
LMIC_setupChannel(0, 921400000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(1, 921600000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENTI); // g-band
LMIC_setupChannel(2, 921800000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(3, 922000000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(4, 922200000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(5, 922400000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(6, 922600000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(7, 922800000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(8, 922700000, DR_RANGE_MAP(DR_FSK, DR_FSK), BAND_MILLI); // g2-band
#endif

```

# Activation by Personalization (ABP)

- Go to settings
- Select ABP and save
- Go back to Overview
- You have now the Device Address and the two 128 AES keys
- You can click on Hex-C Style to have the key in the right format

Activation Method **ABP**

Device EUI

Application EUI

Device Address

Network Session Key

App Session Key



# Activation by Personalization (ABP)

- Open the code UCA\_Education\_Board\Code\LORAWAN\ABP\Basic\UCA-ABP\_Basic\UCA-ABP\_Basic.ino
- Copy/Paste DEVADDR from your TTN window with « 0x » for Hex style
- Copy/Paste NWKSKEY and APPSKEY using C-style from your TTN window

[illegible]

# Activation by Personalization (ABP)

- Compile and download the code on your board
- Look at the TTN device overview
- Frames up should increment each half minute as your board is sending an uplink each 30s (« TX\_INTERVAL »)
- Have look on Data
- For each uplink, you can look many details as RSSI, SNR, airtime, modulation, coding rate, GW ID, etc ...
- Click on the blue triangle

Status ● 25 seconds ago

Frames up 0 [reset frame counters](#)

Frames down 0

	time	counter	port	
▲	21:45:35	3	1	payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21
▲	21:44:29	2	1	payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21
▲	21:43:22	1	1	payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21
▲	21:42:16	0	1	retry payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21

# Activation by Personalization (ABP)

## Frame counter security

- Now reset you board (click on the right button on your board)
- TTN is no more receiving the data
- Click on « reset frame counters » and reset you board again
- As you can see, frame counter is a security features to avoid replay attack (done by capturing and re-transmitting the messages)
- Frame counter can be disabled for debug test in Settings

## Downlink

- Open your serial monitor
- In TTN overview, go to downling, add a payload like « BABA » and click on send, and go to Data
- After the next uplink, you should see the number of byte received in downlink

```
Starting
Packet queued
150865: EV_TXCOMPLETE (includes waiting for RX windows)
Received
2
  bytes of payload
BABA
```

▼ 08:59:10	1	payload: BABA
▲ 08:59:08	0	1 retry payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21

# Activation by Personalization (ABP)

## Change SF, power, payload ...

- At the end of the arduino code, you can find :

LMIC\_setDrTxpow(DR\_SF12,14);

- You can change Spreading Factor(SF) from DR\_SF7 to DR\_SF12
- You can change the power from 2 dBm to 20 dBm
- Payload is in mydata[], and you can change the text.
- You can convert the payload in Hex to normal text using this online [tool](#)
- Change SF and Payload text ! What is the effect on Time on Air ?

OTAA

ABP

### Device Address

26 04 11 88

✓ 4 bytes

### Network Session Key

84 55 4B 72 82 1B CB 22 2D 41 83 96 B7 43 64 B5

✓ 16 bytes

### App Session Key

70 73 2D 4B 54 76 0C 42 A7 59 F8 D5 37 BC FA 24

✓ 16 bytes

### Frame Counter Width

16 bit

32 bit

☐ **Frame Counter Checks**

ⓘ Disabling frame counter checks drastically reduces security and should only be used for development purposes



```
static const PROGMEM ul_t NWKSEKEY[16] = { 0x84, 0x55, 0x4B, 0x72, 0x82, 0x1B, 0xCB, 0x22, 0x2D, 0x41, 0x83, 0x96, 0xB7, 0x43, 0x64, 0xB5 };
```

```
// LoRaWAN AppSKey, application session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const ul_t PROGMEM APPSKEY[16] = { 0x70, 0x73, 0x2D, 0x4B, 0x54, 0x76, 0x0C, 0x42, 0xA7, 0x59, 0xF8, 0xD5, 0x37, 0xBC, 0xFA, 0x24 };
```

```
// These callbacks are only used in over-the-air activation, so they are
// left empty here (we cannot leave them out completely unless
// DISABLE_JOIN is set in config.h, otherwise the linker will complain).
```

```
void os_getArtEui (ul_t* buf) { }
void os_getDevEui (ul_t* buf) { }
void os_getDevKey (ul_t* buf) { }
```

```
static uint8_t mydata[] = "Hello, world!";
static osjob_t sendjob;
```

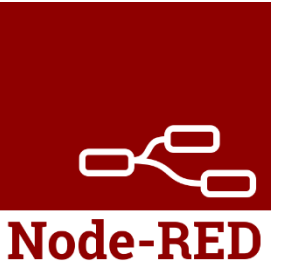
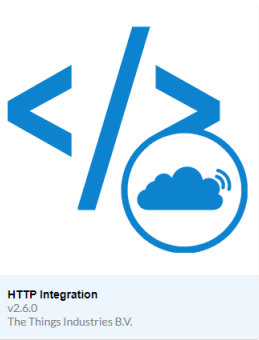
```
// Schedule TX every this many seconds (might become longer due to duty
// cycle limitations).
const unsigned TX_INTERVAL = 30;
```

```
// Pin mapping
const lmic_pinmap lmic_pins = {
    .nss = 10,
    .rxtx = LMIC_UNUSED_PIN,
    .rst = 8,
    .dio = {3, 7, 6},
};
```

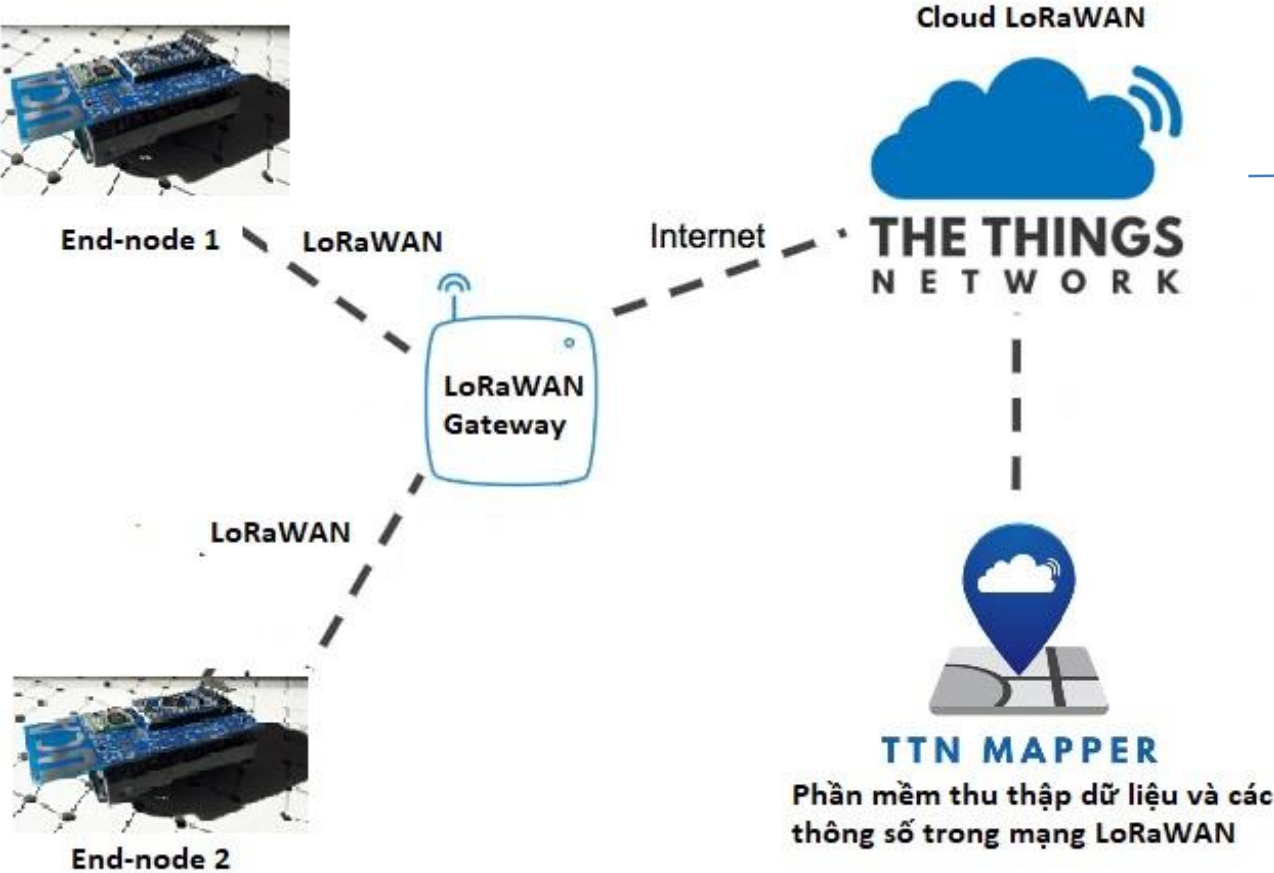
```
void onEvent (ev_t ev) {
```

Thay đổi dữ liệu gửi đi  
Ví dụ: mydata[] = "Hello, Tôi là nhóm abc";

# LoRaWAN Architecture



Application Server



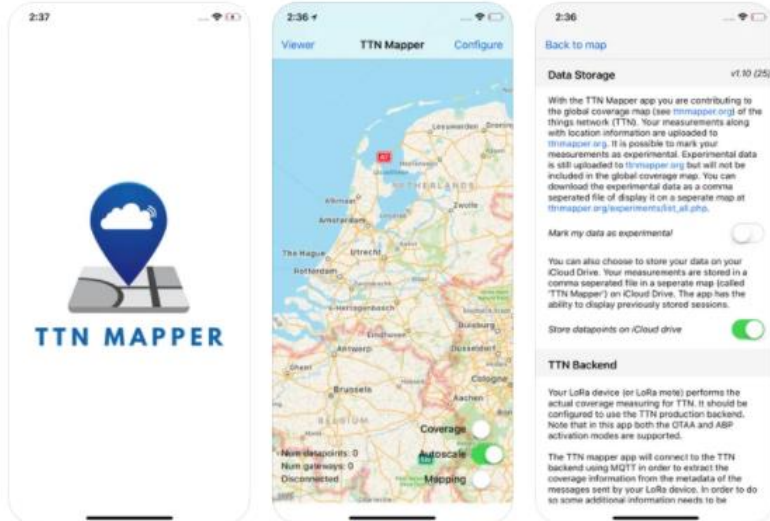
# Install TTN Mapper

## App Store Preview



**TTN Mapper** 4+  
Timothy Sealy  
Designed for iPhone  
★★★★★ 2.3 • 3 Ratings  
Free

## iPhone Screenshots



Mobifone 09:17 70%

[Back to map](#)

additional information needs to be configured.

Or scan a custom QR code to update these settings.

[How to create a QR code.](#)

Scan

Handler region (automatically populated):

asia-se.thethings.network

Please enter your Device ID:

yourDeviceID

Please enter your Application ID:

hue\_lora\_course\_2021

Please enter your Application Access Key:

ttn-account-v2.JgeWlRpTZ2TIS2e39Y...

Cancel

Test config

Save



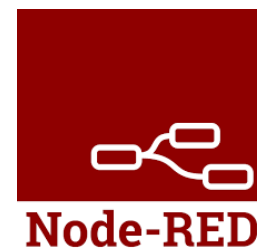
Scan HERE

QR CODE – Configure TTN mapper for  
hue\_lora\_course\_2021

# Test TTN Mapper



# Application server - Node Red



- Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.
- It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.
- Built on Node.js
  - The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.



# Install Nodejs and Node-Red

- Install nodejs: <https://nodejs.org/en/download/>
- Install git : <https://git-scm.com/downloads>
- Then install Node Red, follow this tutorial :
  - <https://nodered.org/docs/getting-started/windows>

## 2. Install Node-RED

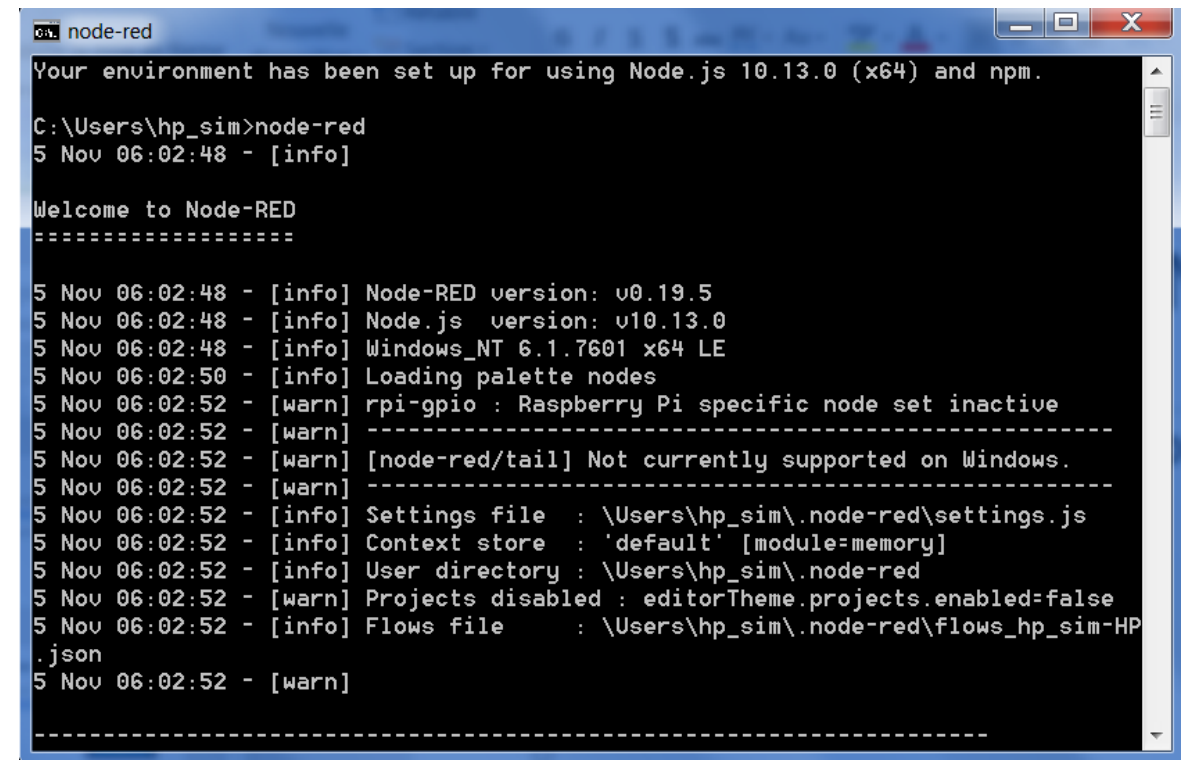
Installing Node-RED as a global module adds the command `node-red` to your system path. Execute the following at the command prompt:

```
npm install -g --unsafe-perm node-red
```

- Install package in Node Red :
  - node-red-contrib-ttn
  - cd \$HOME/.node-red
  - npm install node-red-contrib-ttn

# Connecting to TTN

- Start NODE.js command prompt
- Run : node-red
- Open your web browser and go to <http://127.0.0.1:1880>



```
node-red
Your environment has been set up for using Node.js 10.13.0 (x64) and npm.

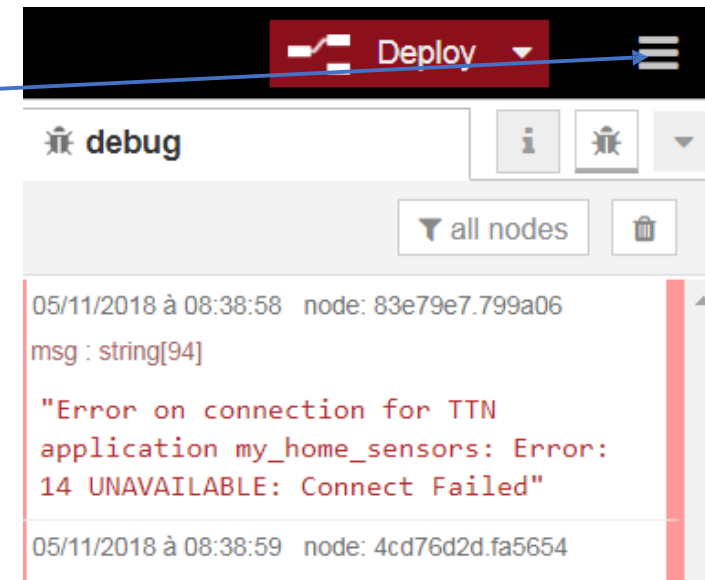
C:\Users\hp_sim>node-red
5 Nov 06:02:48 - [info]

Welcome to Node-RED
=====

5 Nov 06:02:48 - [info] Node-RED version: v0.19.5
5 Nov 06:02:48 - [info] Node.js version: v10.13.0
5 Nov 06:02:48 - [info] Windows_NT 6.1.7601 x64 LE
5 Nov 06:02:50 - [info] Loading palette nodes
5 Nov 06:02:52 - [warn] rpi-gpio : Raspberry Pi specific node set inactive
5 Nov 06:02:52 - [warn] -----
5 Nov 06:02:52 - [warn] [node-red/tail] Not currently supported on Windows.
5 Nov 06:02:52 - [warn] -----
5 Nov 06:02:52 - [info] Settings file   : \Users\hp_sim\.node-red\settings.js
5 Nov 06:02:52 - [info] Context store  : 'default' [module=memory]
5 Nov 06:02:52 - [info] User directory : \Users\hp_sim\.node-red
5 Nov 06:02:52 - [warn] Projects disabled : editorTheme.projects.enabled=false
5 Nov 06:02:52 - [info] Flows file     : \Users\hp_sim\.node-red\flows_hp_sim-HP.json
5 Nov 06:02:52 - [warn] -----
```

- On the editor, click here
- And go to palette editor
- Install :

- node-red-contrib-ttn
- OR npm install node-red-contrib-ttn
- OR npm i node-red-contrib-ttn@1.0.0.



Deploy

User Settings

Close

View

Nodes

Install

Keyboard

sort:

a-z

recent

node-red-contrib-ttn

1 / 3176

node-red-contrib-ttn

The Things Network Node-RED Application Nodes

2.0.5 2 years, 4 months ago

installed

info

Search flows

Flows

Flow 1

Subflows

Global Configuration Nodes

Flow 1

Flow"8f8cc7f3.8bc628"

Enable or disable these tips from the option in the settings

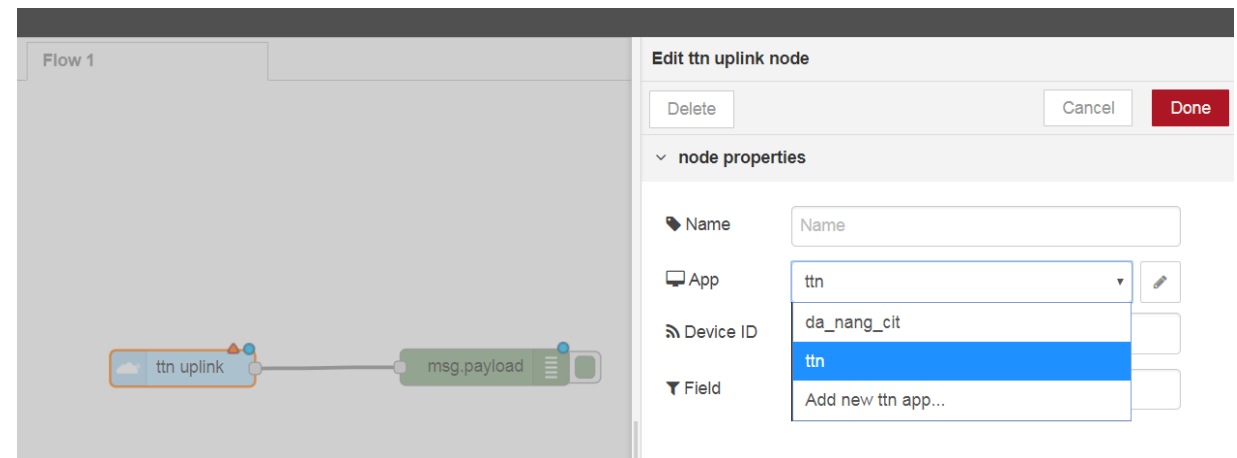
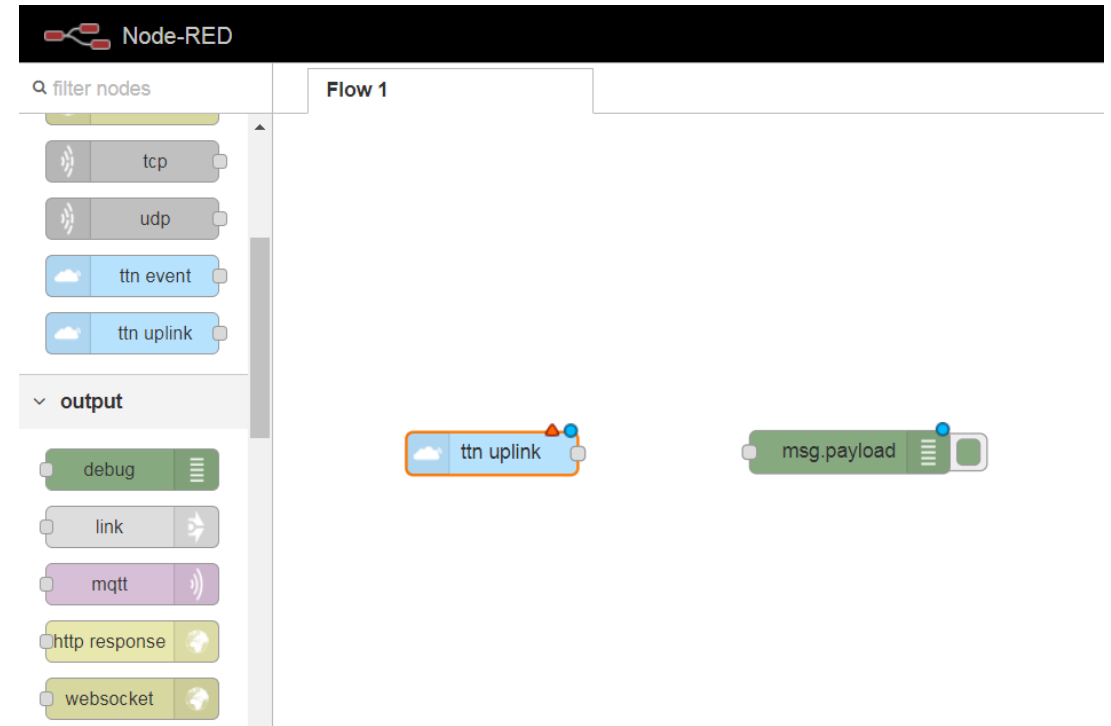
# Connecting to TTN

- You have the graphical Node-red editor
- Add ttn uplink and a debug output
- Edit TTN uplink
- Choose « Add new ttn app ...» in App and click on edit

App ID

Access Key

Discovery address



# Connecting to TTN

- You need :

- App ID :
- Access Key :
- Discovery adress :

discovery.thethingsnetwork.org:1900

- Go to you application in TTN
- Copy paste the Application ID and Access Key

Edit ttn uplink node > Edit ttn app node

Delete Cancel Update

App ID 70B3D57ED0013EAD

Access Key .....

Discovery address discovery.thethingsnetwork.org:1900

Applications > my\_home\_sensors

Overview Devices Payload Formats Integrations Data Settings

APPLICATION OVERVIEW

Application ID my\_home\_sensors [documentation](#)

Description My Home sensors

Created 7 months ago

Handler ttn-handler-eu (current handler)

APPLICATION EUIs [manage euis](#)

<> 70 B3 D5 7E D0 00 81 84

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

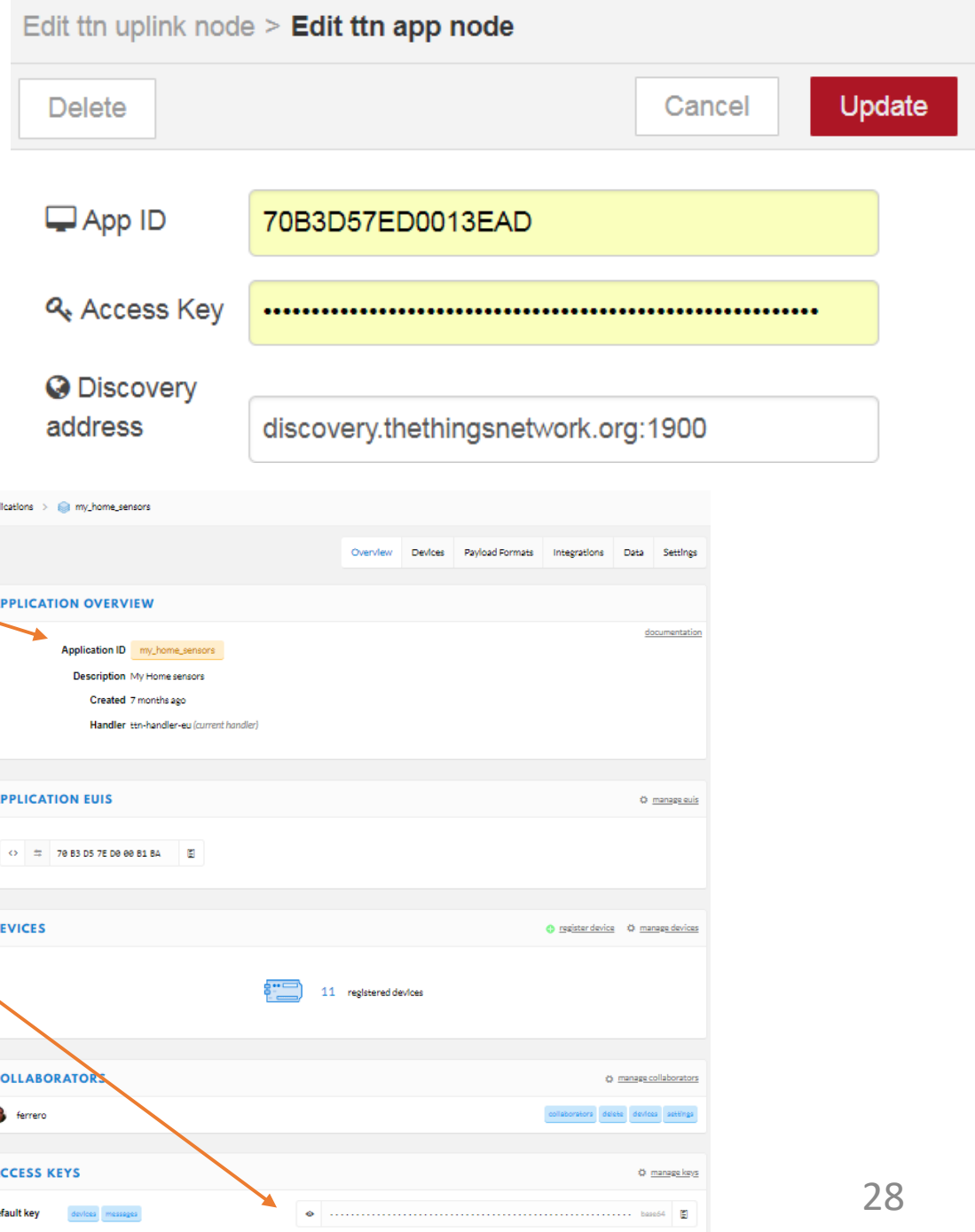
11 registered devices

COLLABORATORS [manage collaborators](#)

ferro [collaborators](#) [delete](#) [devices](#) [settings](#)

ACCESS KEYS [manage keys](#)

default key [delete](#) [messages](#) ..... base64





# Connecting to TTN

- Click on Deploy
- You uplink TTN should be connected
- Click on debug window
- You will receive the packet of the application
- If you want to filter only your device, add your device ID
- Click here :

The screenshot displays the TTN (The Things Network) console interface. At the top, a black bar contains a red 'Deploy' button. Below this, a 'debug' window is open, showing a log entry for a node with ID 'ba3a9ad9.e50308'. The log entry is dated '05/11/2018 à 06:17:40' and contains a JSON payload: `{ analog_in_3: 5.14, digital_out_4: 0, relative_humidity_2: 79, temperature_1: 21.2 }`. A blue arrow points from the 'debug' window to the 'Click here' bullet point in the list.

In the center, a diagram shows a node labeled 'my\_home\_sensors' connected to a 'msg.payload' block. A green status indicator below the node says 'connected'. A blue arrow points from the 'Click on debug window' bullet point to the 'debug' window.

At the bottom left, the 'DEVICE OVERVIEW' section shows the following details:

- Application ID: my\_home\_sensors
- Device ID: 50ff1a0000010004
- Description: Light test
- Activation Method: OTAA

A blue arrow points from the 'Device ID' field to the 'Click here' bullet point in the list.

At the bottom right, the 'Edit ttn uplink node' dialog is open, showing the following fields:

- Name: my\_home\_sensors
- App: my\_home\_sensors
- Device ID: 50ff1a0000010004
- Field: (empty)

A blue arrow points from the 'Device ID' field in the dialog to the 'Device ID' field in the 'DEVICE OVERVIEW' section.

# Connecting to TTN

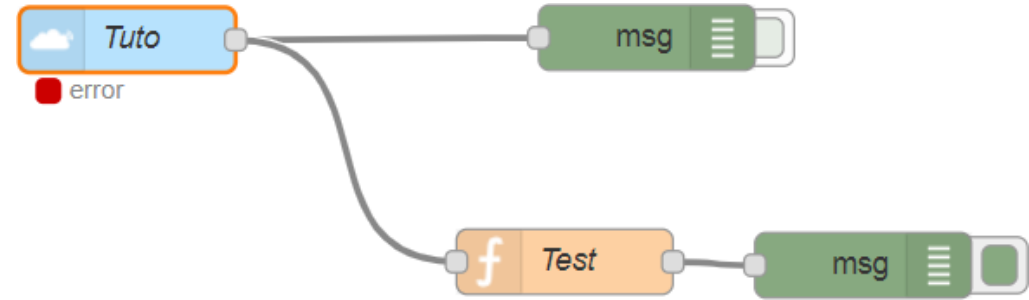
- Click here :
- Choose « complete msg object »
- And Deploy
- You have now more information of your uplink

The screenshot displays the TTN console interface. At the top, a black bar contains a 'Deploy' button. Below this, a node configuration is shown with a blue box labeled 'my\_home\_sensors' (cloud icon) connected to a green box labeled 'msg.payload' (list icon). A blue arrow points from the 'msg.payload' box to the 'complete msg object' option in the 'Edit debug node' dialog. The dialog has buttons for 'Delete', 'Cancel', and 'Done'. Under 'node properties', the 'Output' dropdown is set to 'complete msg object', and the 'To' checkbox for 'debug window' is checked. The 'Name' field is empty. On the left, a code editor shows the debug output for node 'ba3a9ad9.e50308' at 05/11/2018 06:27:44. The output is a JSON object with fields: 'app\_id', 'dev\_id', 'hardware\_serial', 'port', 'counter', 'payload\_raw', 'payload\_fields', 'metadata', 'payload', and '\_msgid'.

```
05/11/2018 à 06:27:44 node: ba3a9ad9.e50308
msg : Object
  object
    app_id: "my_home_sensors"
    dev_id: "my_cellar_sensor"
    hardware_serial: "50FF1A0000010003"
    port: 1
    counter: 32781
    payload_raw: buffer[14]
    payload_fields: object
    metadata: object
    payload: object
    _msgid: "fc884851.257408"
```

# Connecting to TTN

- If you want to extract only 1 data,
- As an exemple the RSSI (received signal Strength indicator
- Use a function to extract the wanted data



```
var gateways = msg.metadata.gateways;

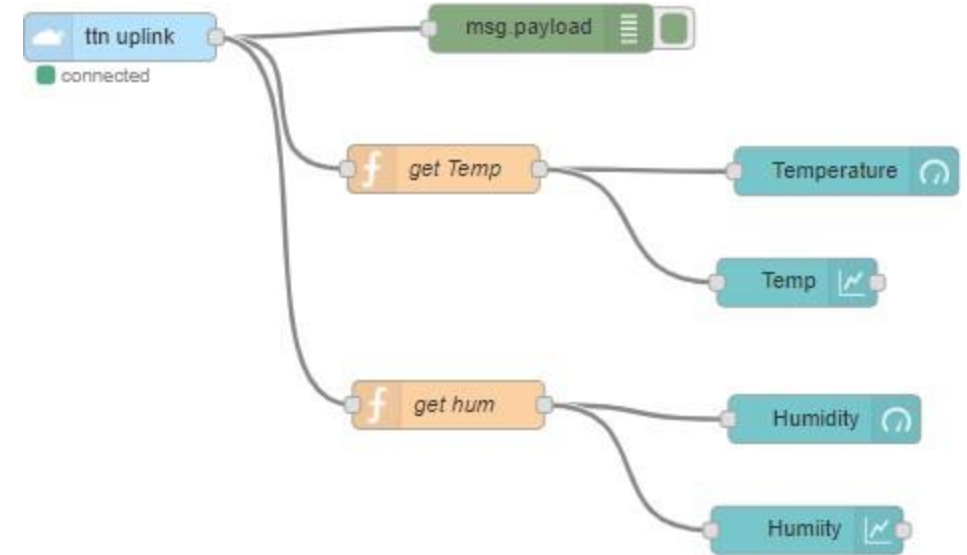
return {
  // Some fields from the metadata freq:
  msg.metadata.frequency,
  cr: msg.metadata.cr,
  dr: msg.metadata.dr,

  // Combine RSSI and SNR of all gateways into two arrays:
  rssi: gateways.map(gw => gw.rssi),
  snr: gateways.map(gw => gw.snr),

};
```

# Add a Dashboard

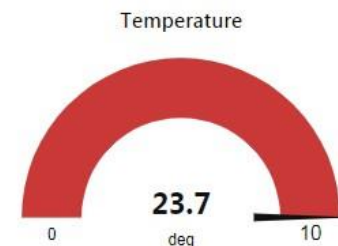
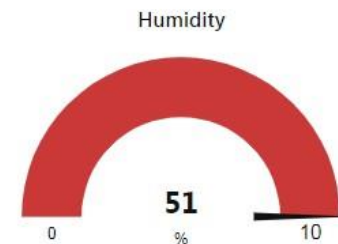
- Go to Manage Palette, select Install
- Install : node-red-dashboard
- Add a function to extract sensor values (Temp, Hum, luminosity...)
- Add Gauge and Graph for Dashboard section
- Go to : <http://127.0.0.1:1880/ui/>



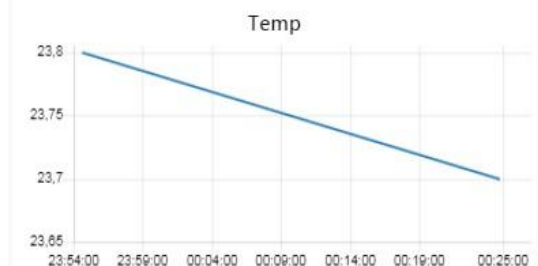
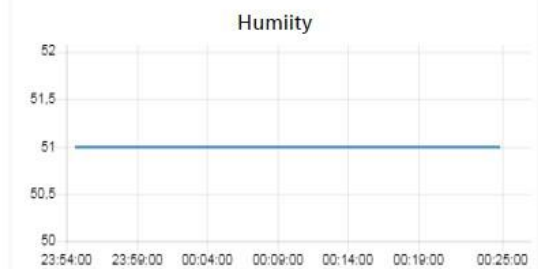
## Function

```
1 var temp = {}  
2 temp.payload = msg.payload.temperature_4;  
3 return temp;
```

test



Test

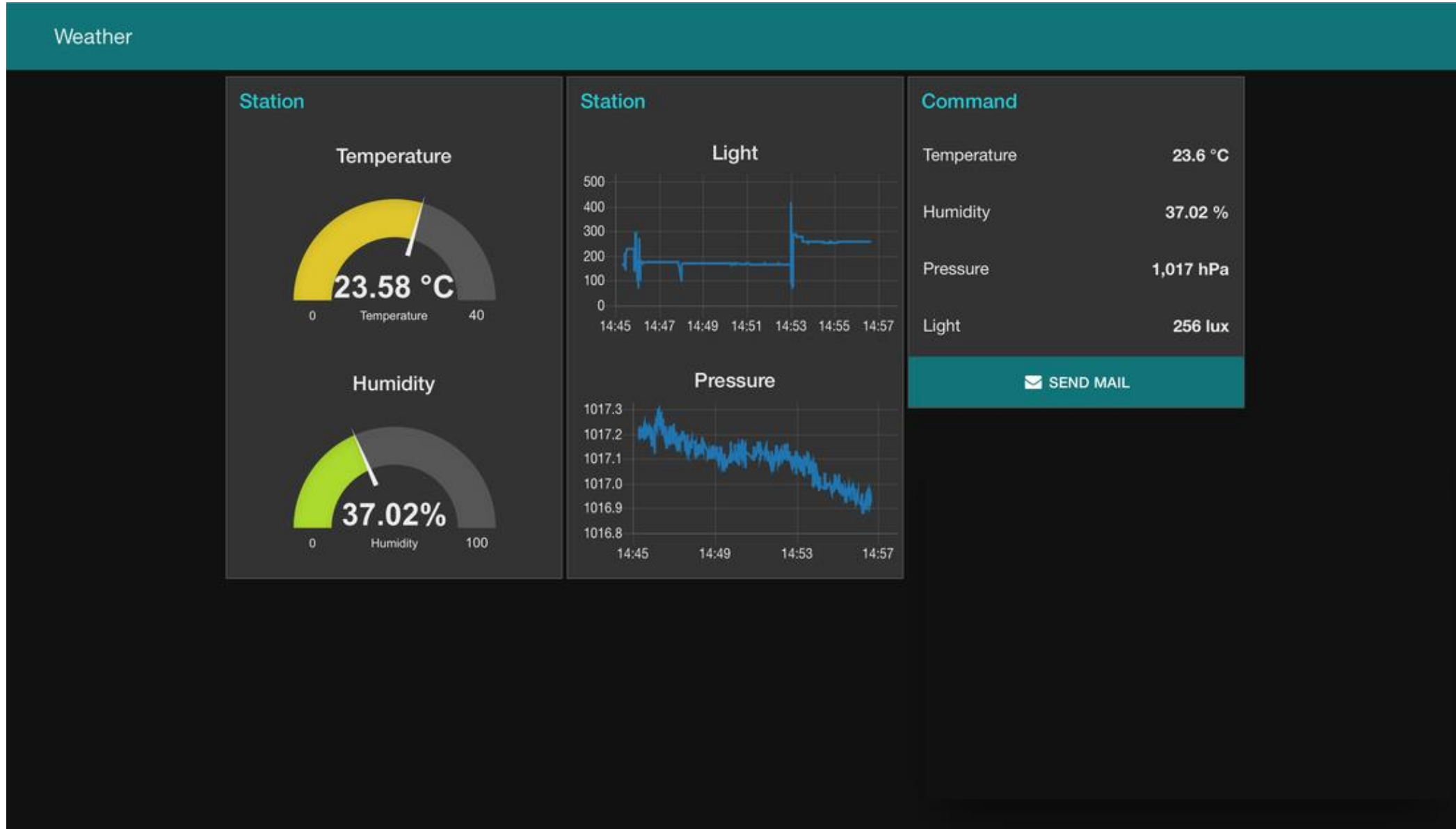


# UCA Education Board with sensor

- Depending on the sensor available from your instructor, you may use :
  - BME280 : T°C, Humidity and Pressure with I2C connection
  - SI7021 : T°C and Humidity with I2C connection
  - TEMT6000 : Ambient light with analog output
  - SRC4+ : Distance with digital connection
  - Or else ...
- You have to wire the sensor on the breadboard on the UCA board



# Example: Weather monitoring





Good luck for you projects!

