



Start your IoT project with Gimasi Starter Kit Tuino & TP Developer Portal



In partnership with





Actility
Connecting with intelligence

NOTICE

This document contains proprietary and confidential material of ACTILITY SA. This document is provided under and governed by either a license or confidentiality agreement. Any unauthorized reproduction, use, or disclosure of this material, or any part thereof, is strictly prohibited.

The material provided in this document is believed to be accurate and reliable. However, no responsibility is assumed by Actility SA for the use of this material. Actility SA reserves the right to make changes to the material at any time and without notice. This document is intended for information and operational purposes only. No part of this document shall constitute any contractual commitment by Actility SA.

© 2017 ACTILITY SA. All rights reserved.

Portions of this documentation and of the software herein described are used by permission of their copyright owners.

Actility, ThingPark, are registered trademarks of Actility SA or its subsidiaries may also be registered in other countries.

Other denoted product names of Actility SA or other companies may be trademarks or registered trademarks of Actility SA or its subsidiaries, or their respective owners.

Headquarters

Actility Lannion,
Actility S.A 4 rue Ampère BP 30225
22300 Lannion France
www.actility.com



Document versions

Version	Date	Author	Details
1.0	02/05/2017	A. AMIOT R. GUNSETT	Initial version
2.0	09/05/2017	A. AMIOT R. GUNSETT	Update
3.0	03/07/2017	A. AMIOT R. GUNSETT	Update
4.0	07/09/2017	A. AMIOT R. GUNSETT	Update MultiRegion Firmware



Activity
Connecting with intelligence

Table of Contents

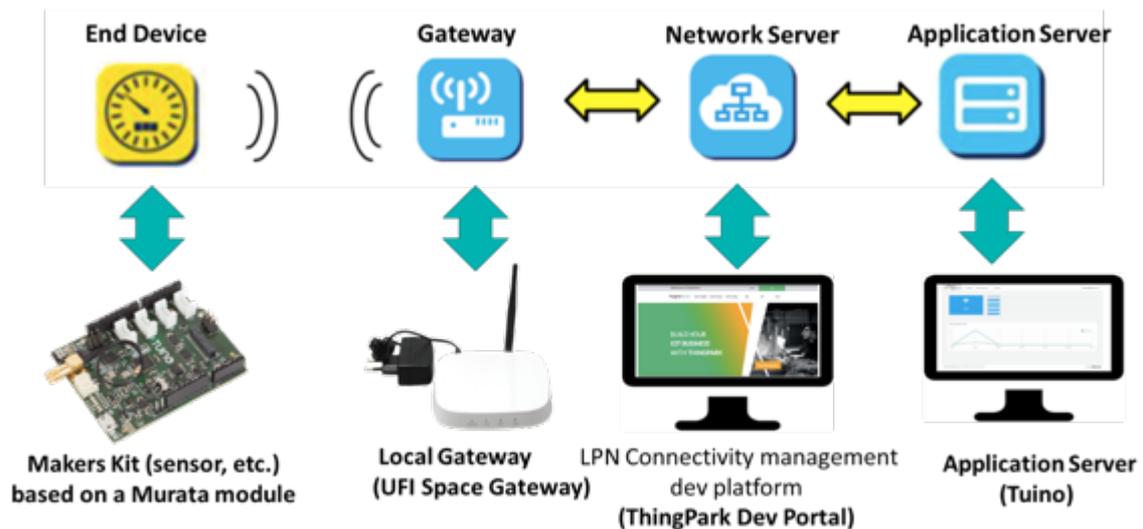
1	GET THE GIMASI DEVKIT	2
2	ASSEMBLE YOUR DEVKIT	2
3	FLASH THE DEVKIT FOR THE FIRST TIME	3
3.1	Set up the device node development Environment.....	3
3.2	Build and flash the Arduino	5
4	CREATE AN ACCOUNT ON THING PARK.....	7
5	PROVISION YOUR GATEWAY (BASE STATION).....	8
6	SETUP APPLICATION ROUTING	9
6.1	Create the Application Server	9
6.2	Create the routing profile.....	10
7	ACTIVATE YOUR DEVICE.....	11
7.1	Create your device	11
7.2	Check that your device is working.....	13
8	USING THE WIRELESS LOGGER	14
9	USING TUINO IOT PLATFORM	16
9.1	Create your device in Tuino Cloud	16
9.2	Send data to node	17
10	BUILD YOUR DEVKIT WEB APPLICATION	18
11	TURN THE THERMOSTAT FOR A BOILER INTO A THERMOSTAT FOR A A/C UNIT ..	20
12	UPDATE YOUR LORAMAC FIRMWARE (ADVANCED).....	21



Overview and Scope

The intention behind is to help developers, project managers and entrepreneurs to experience from a very concrete point of view the LoRaWAN technology by prototyping their first LoRaWAN device.

Using this document, you should be able to go through an end to end experience from IoT Sensor developer all the way to Web Application developer.



This end to end experience is based on the use of the Gimasi dev Kit V2, the ThingPark dev portal and the Tuino application server.



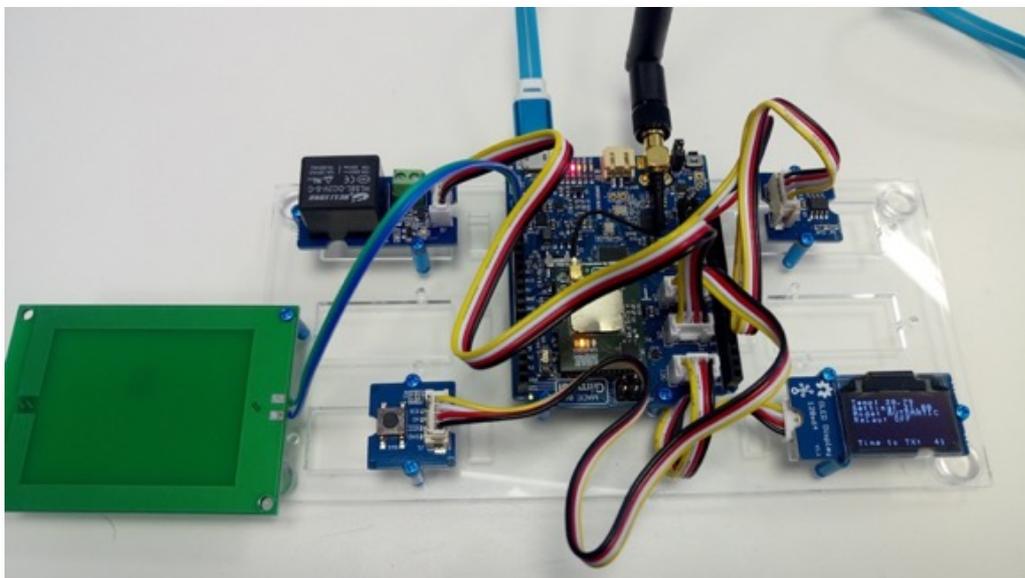
1 GET THE GIMASI DEVKIT

Warning: if you do not have this devkit you can order it NOW through our ThingPark Marketplace:

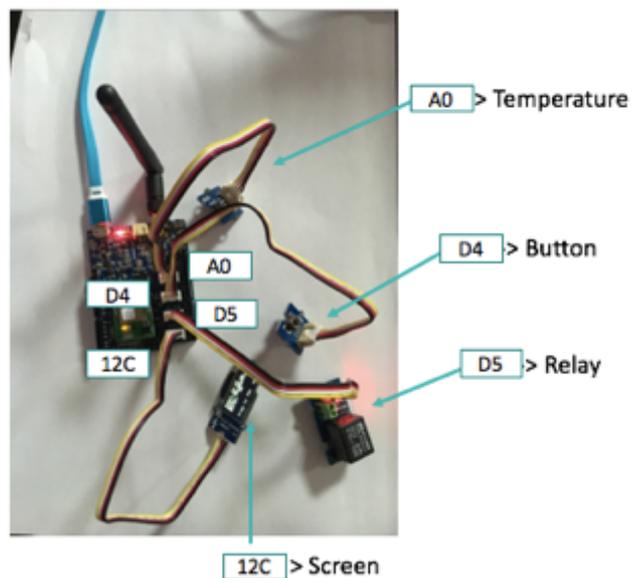
<https://market.thingpark.com/iot-devices/dev-kits-field-tests/gimasi-simple-kit-tunio-1-arduino-compatible-board-lr1-lora-module-tuino1-lr1-kit-eu868>

This user manual is also accessible on the product information at that page under the Technical docs' section.

2 ASSEMBLE YOUR DEVKIT



1. Open your devkit Box
2. Discover your DevKit Components
3. Plug the 4 provided sensors to the right ports as explained on this picture:



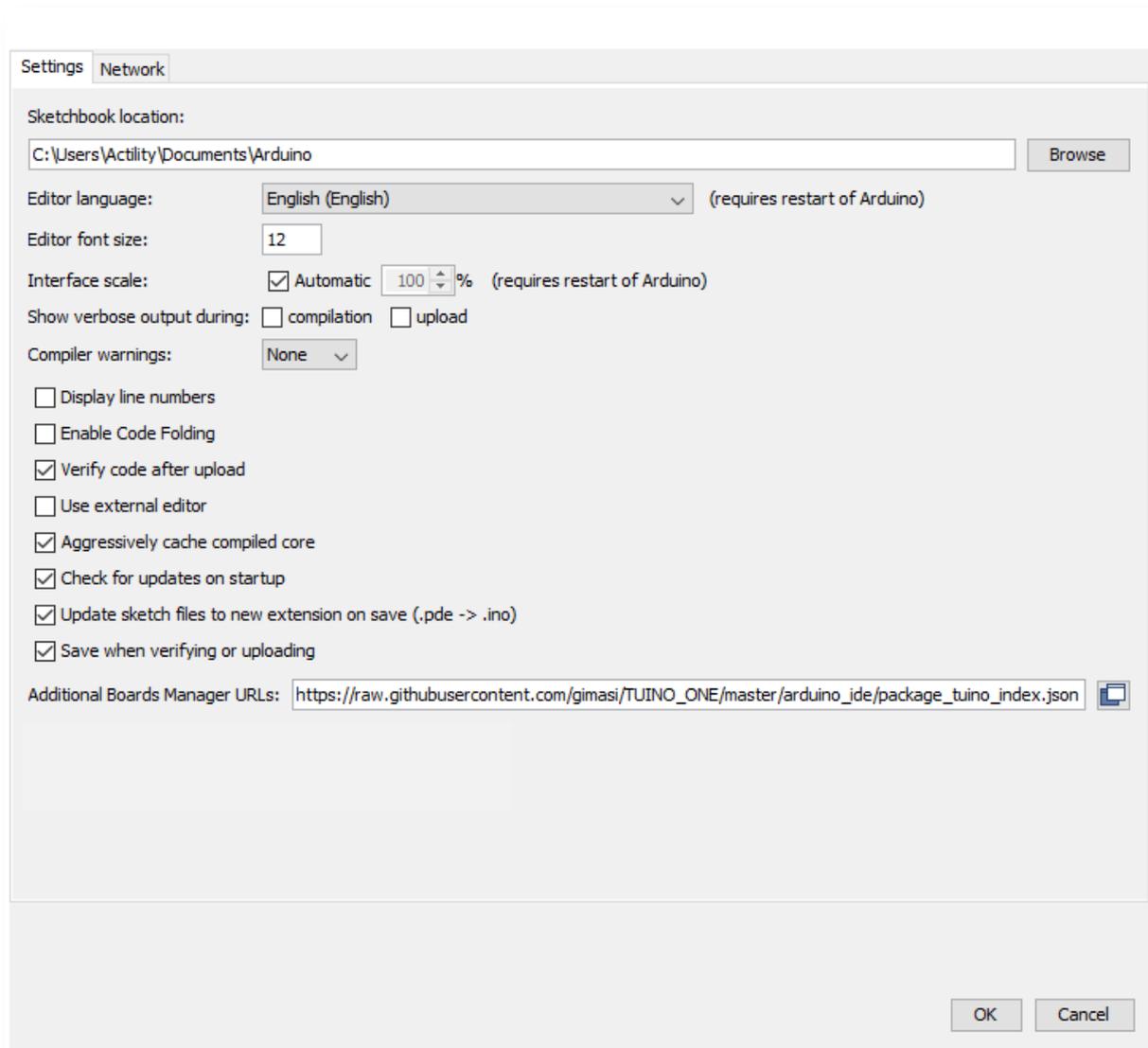
Actility S.A. au capital de 744 810 € - 4 rue Ampère, 22300 Lannion, France
RCS St Brieuc 522 305 473, Siret 522 305 473 00012, TVA FR62522305473



3 FLASH THE DEVKIT FOR THE FIRST TIME

3.1 Set up the device node development Environment

1. Download and install the toolchain from Arduino's website at <https://www.arduino.cc>.
2. Once installed, launch the Arduino IDE
3. Go to **File | Preferences** (on MacOS: **Arduino | Preferences**)

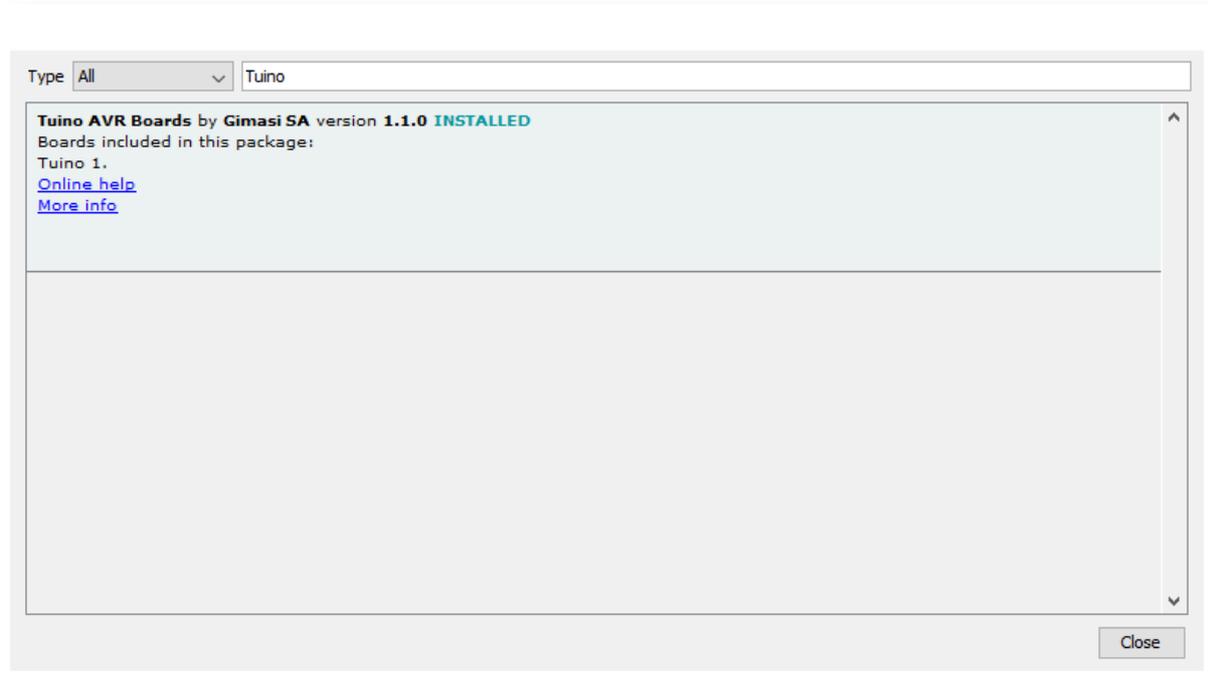


4. In the field "Additional Boards Manager URLs" enter the following:
https://raw.githubusercontent.com/gimasi/TUINO_ONE/master/arduino_ide/package_tuino_index.json
5. Click on the "OK" button
6. Go to **Tools | Boards | Boards Manager ...**

Activity S.A. au capital de 744 810 € - 4 rue Ampère, 22300 Lannion, France
RCS St Brieuc 522 305 473, Siret 522 305 473 00012, TVA FR62522305473



7. Enter “Tuino” in the search bar
8. Click on “Tuino AVR Board by Gimasi SA” then “Intall”
9. Once all the packages have been downloaded and installed, the version number of the package should be displayed next to “Gimasi SA”



10. Click on “Close”

THE ARDUINO IDE IS NOW SET UP TO RECEIVE THE CODE FOR THE TUINO 1 BOARD.

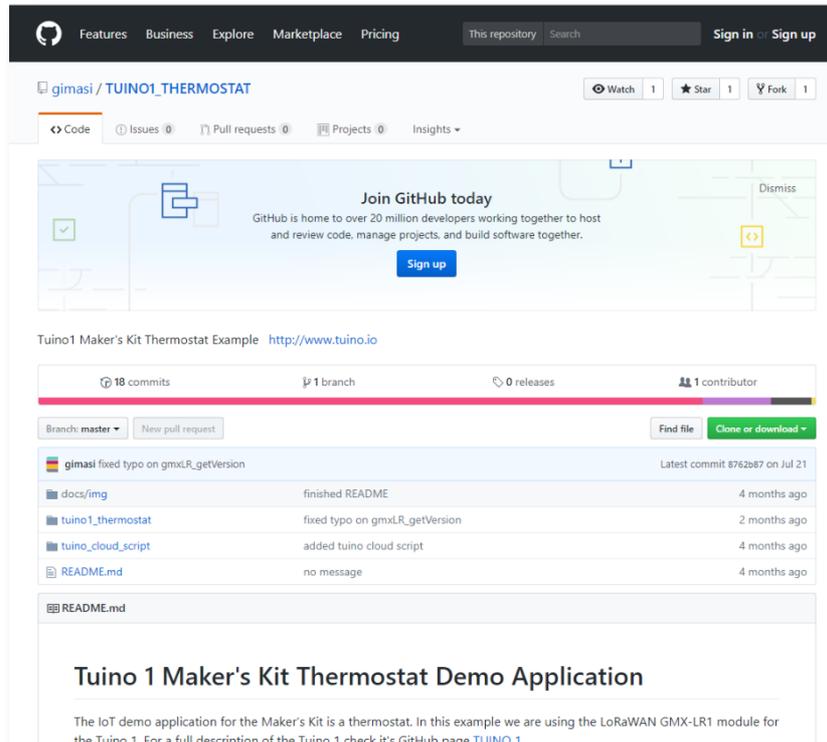
Note: All this can be found at GitHub Tuino One repository:

https://github.com/gimasi/TUINO_ONE.



3.2 Build and flash the Arduino

1. Download the Tuino 1 code from https://github.com/gimasi/TUINO1_THERMOSTAT by clicking on the 'Click or download' button then on 'Download ZIP'.



2. Decompress the downloaded archive into a dedicated folder on your computer.

Note: This file must be decompressed for the Arduino toolchain to be able to build the firmware. Opening the file from the compressed archive will not work.

3. In that folder, under `thuino1_thermostat` folder, open the file `thuino1_thermostat.ino`. Most of the time double clicking on that file will suffice, otherwise open the Arduino IDE and go to **File | Open...** then browse to that file.
4. Go to **Tools | Board** and select "Tuino 1" at the end of the list
5. Click on the "Verify" button (the checkmark button), or go to the menu **Sketch | Verify/Compile**.
6. The program should check out (build) successfully. A message like the one below should be displayed below the source code:

Sketch uses 28306 bytes (21%) of program storage space. Maximum is 130048 bytes. Global variables use 2372 bytes (14%) of dynamic memory, leaving 14012 bytes for local variables. Maximum is 16384 bytes.

7. Make sure that the Tuino 1 Development Kit is connected to the computer
8. In the menu, go to **Tools | Port** and select the port stating "Arduino" in its name

Activity S.A. au capital de 744 810 € - 4 rue Ampère, 22300 Lannion, France
RCS St Brieuc 522 305 473, Siret 522 305 473 00012, TVA FR62522305473



9. Click on the “Upload” button (the arrow button) to start the transfer process.
10. During the upload, the TX and RX LEDs next to the USB connector should be flashing intensely
11. Once the transfer is complete, the software on the Tuino 1 will reset. The display will then display the welcome message (“TUINO1 LoRa Thermostat”) and wait for the button to be pressed.

THE TUINO 1 BOARD IS OPEN FOR BUSINESS.

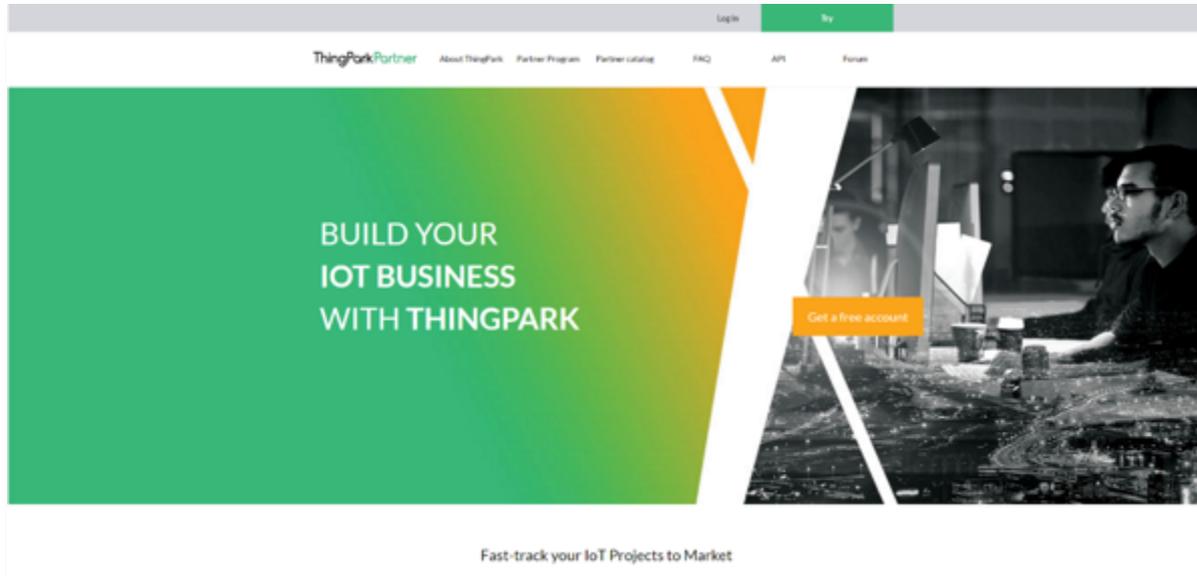
***Note:** It is possible that the LoRaMAC module does not contain the most up to date. To ensure that your module is using the most recent firmware, please go to section 12 and follow the procedure.*

***Note:** If you are in a region outside of EU868 band, please make sure you update your module as most modules got shipped with an older revision of the firmware which only supported EU868 bands.*



4 CREATE AN ACCOUNT ON THING PARK

1. Go to <https://partners.thingpark.com>



2. Click on the “Try” button on top
3. Fill out the form with all the required information

Note: *The information given here need to be unique in our system otherwise the registration will fail.*

4. Click on “Create new account”
5. An email should be sent at the given address.

YOU ARE NOW ALL SET TO START USING THE THINGPARK PARTNER PLATFORM.

Actility S.A. au capital de 744 810 € - 4 rue Ampère, 22300 Lannion, France
RCS St Brieuc 522 305 473, Siret 522 305 473 00012, TVA FR62522305473



5 PROVISION YOUR GATEWAY (BASE STATION)

Note: if you are in an area (such as an incubator, a school, a university, a lab...) covered by a local LoRa development Network, you can skip directly to step 2 of the user guide.

The base station provisioning allows the Network Partner to add a new base station to its network. The information required to create a new base station are:

- ID
- Serial Number (SMN)
- Model

To create a new base station:

1. From the Partner Portal dashboard, click on the **Network Manager** entry from the left side menu.
2. A new window/tab will open into your Network Manager.
3. Click on the 'Base stations' on the left then on the 'Create' button from the 'Add base stations' section.
4. A new pop-up appears with 4 fields: ID, SMN, Model, Name

New base station

Use this dialog box to add a new base station. New base station becomes active only after review and approval from the connectivity supplier.

ID: 12345678

SMN: 1234-AC-5678-9ABC

Model: Actility outdoor bs (kerlink 868mhz)

Name: My base station

5. Fill out the fields. Please note that all fields are mandatory
 - ID: Identification of the base station (8 hexadecimal digits)
 - SMN: Serial Number (format: 0000-XX-0000-0000)

The fields ID and SMN are written on the label found on the base station, if you do not have it, enter 0000-XX-0000-0000 as SMN.

- Select the right model
 - Enter a name
6. Click on **Create** to add the base station.
 7. To view the base station in the list, click on Search in the search section.



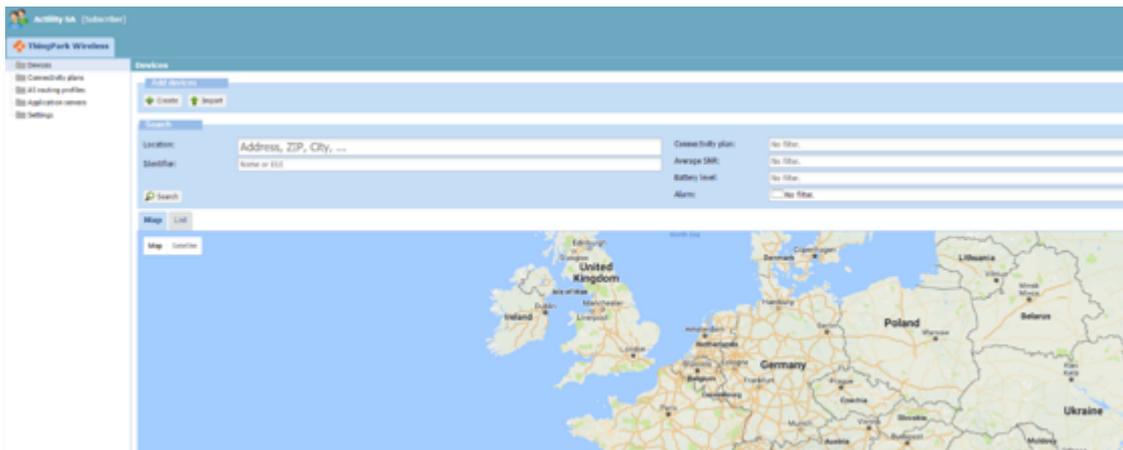
Note: Once a new base station has been created, it stays in status **VALIDATING** until your **Operator Manager** has validated the base station on its network.



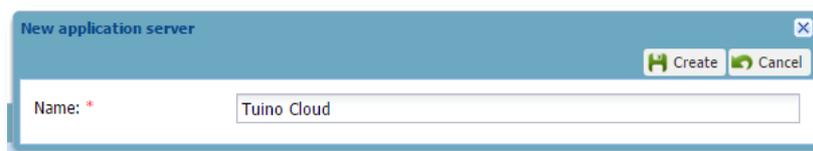
6 SETUP APPLICATION ROUTING

6.1 Create the Application Server

1. From the partner portal, go to the left side menu and click on “Device Manager” in the “ThingPark Applications” section
2. Click on Device Manager (Application Section Dashboard menu). A new tab opens with your Device Manager application.



3. Click on “Application servers” on the left side section
4. Click on “Create” button in the “Add application servers” section, in the new pop up dialog, enter the name of the Application Server: “Tuino Cloud” then click on “Create”.



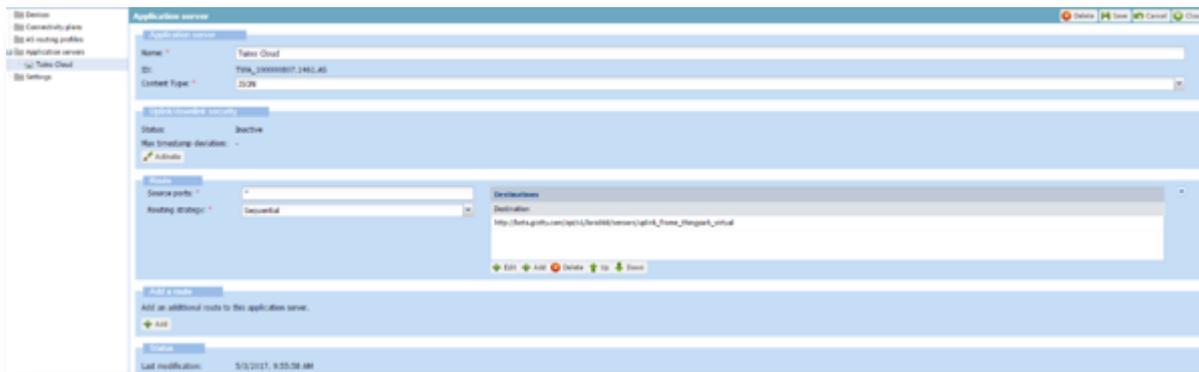
5. In the new application server screen, under the “Application server” section, change the content type to JSON



6. Under the “Add a route” section, click on “Add” button and enter the following information:
 - a. Source ports: *
 - b. Routing strategy: Sequential
 - c. Under the “Destinations” box, click on “Add” button and enter the following URL in the pop up menu:

http://beta.giotty.com/api/v1/lora868/sensors/uplink_frame_thingpark_virtual

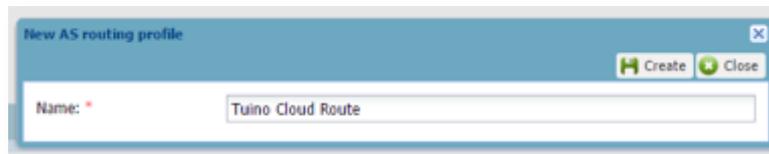
- d. Click on the “Add” button



7. Click on “Save” button, a small pop up should appear in the bottom left corner stating “The application server has been successfully updated”

6.2 Create the routing profile

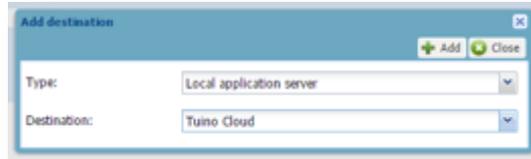
1. In the Device Manager left hand side, click on “AS routing profiles”
2. Under the “Add AS routing profiles” section, click on “Create”
3. Indicate a name (for your own usage): “Tuino Cloud Route”



4. Click on “Create” button. A small pop up message appears on the bottom left corner: “AS routing profile successfully updated”
5. In the new dialog, under the “Destinations” section, click on the “Add” button



6. In the pop up dialog, select as “Type”: “Local Application Server” and as “Destination” the Application Server created above: “Tuino Cloud”

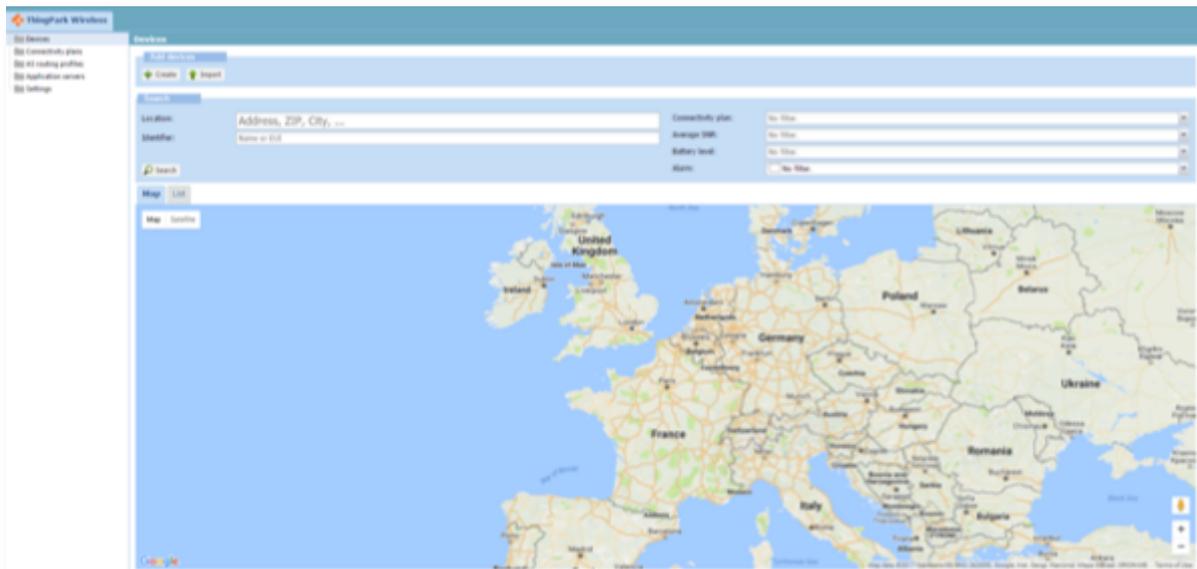


7. Click on the “Add” button
8. Click on the “Save” button
9. A pop up message appears at the bottom left corner: “The AS routing profile has been successfully updated”

7 ACTIVATE YOUR DEVICE

7.1 Create your device

1. In the Device Manager, click on “Devices” in the left-hand side section



2. Under the “Add devices” section, click on the “Create” button



3. In the pop up dialog, enter the following information:



- a. Administrative data:
 - Device Name: “Tuino 1 DevKit”
- b. Device identification:
 - Device activation: “Over The Air Activation (OTAA)”
 - DevEUI: *check the device label*
 - AppEUI: *check the device label*
 - AppKey: *check the device label*
 - Device profile: “LoRaWAN 1.0 – class C – ETSI – Rx2_SF12”
- c. Network parameters:
 - Connectivity plan: select the default plan to allow for device activation
- d. Application layer handling:
 - Application server routing profile: select the routing profile that was previously created. In this case, “Tuino Cloud Route”

Note: This information need to be accurate or the device will not be recognized by the system. Please make sure they are accurate.

4. Click on “Create” button to add the device
5. Click on the “List” tab then on the “Search” button
6. The device should be in the list now



Note: Alternative ways of activating a device can be used, such as the DX API.

Note: If multiple devices need to be provisioned, a list can be imported from csv file



7.2 Check that your device is working

1. Connect the Tuino 1 to a power supply, such as the computer USB port
2. After a few seconds, the device screen should display a small message describing the board: "TUINO1 LoRa Thermostat"

Note: If the screen does not turn on, please check the cabling and that the firmware has been properly flashed. For these two items, please check sections 2 and 3 respectively

3. Press the device button, the device screen should change and display the provisioning information exactly as written on the label and as it was provisioned into the Device Manager
4. Press the device button again, the device attempts to join
5. Once the device has successfully joined the network, the screen will change again and will display the temperature and start sending messages
6. Wait for the countdown to go to 0 then the device screen should display "LoRa TX" when attempting to send an uplink message



IoT Dev Event_User_Guide

V4

7. On the device, click on the button, the device should send a new uplink.
8. On the Wireless Logger, click on the “Refresh” button
9. In the updated list, check that a new packet has been received.

The screenshot shows the '50 last packets' table in the Wireless Logger application. The table has the following columns: UTC Timestamp, Local Timestamp, DevAddr, DevEUI, RPort, Pkts #, Rssi, SNR, ESP, SF, Subband, Channel, LQI, LRR, LRR Lat, LRR Long, LRR SNR, Device Lat, Device Lon, and LQI Delta. The table contains several rows of data, with some cells highlighted in green or orange. A 'Refresh' button is visible at the top of the interface.

UTC Timestamp	Local Timestamp	DevAddr	DevEUI	RPort	Pkts #	Rssi	SNR	ESP	SF	Subband	Channel	LQI	LRR	LRR Lat	LRR Long	LRR SNR	Device Lat	Device Lon	LQI Delta
2017-05-03 15:01:21.579	2017-05-03 17:01:21.579	04CE8074	001222340536177	0	4				10	00	UC10	0000127	2000128						
2017-05-03 15:01:28.579	2017-05-03 17:01:28.579	04CE8074	001222340536177	2	4	-87	3	-45.2%	10	00	UC5	0000127	2000128	48.87490	2.132179	1			
2017-05-03 15:00:51.130	2017-05-03 17:00:51.130	04CE8074	001222340536177	0	4				10	00	UC10	0000127	2000128						
2017-05-03 15:00:58.130	2017-05-03 17:00:58.130	04CE8074	001222340536177	2	4	-84	11.2%	94.2%	10	00	UC6	0000127	2000128	48.87490	2.132179	1			
2017-05-03 15:00:48.829	2017-05-03 17:00:48.829	04CE8074	001222340536177	2	3				10	00	UC10	0000127	2000128						
2017-05-03 15:00:39.829	2017-05-03 17:00:39.829	04CE8074	001222340536177	2	3	-81	6.2%	-41.8%	10	00	UC7	0000127	2000128	48.87490	2.132179	1			
2017-05-03 15:00:38.522	2017-05-03 17:00:38.522	04CE8074	001222340536177	0	2				10	00	R1C	0000127	2000128						
2017-05-03 15:00:29.522	2017-05-03 17:00:29.522	04CE8074	001222340536177	2	2	-86	1%	-46.3%	10	00	UC1	0000127	2000128	48.87490	2.132179	1			



9 USING TUINO IOT PLATFORM

9.1 Create your device in Tuino Cloud

1. Go to <http://thingpark.tuino.io> and click on “New User? Sign up!” to register by entering your email address and a password and click on “Sign up”
2. Check your email box and look for a confirmation email “Confirmation instructions for your Account” and click on the link to confirm your account
3. Log in the website with the credentials given during the registration
4. In the main page, under the “Logging” section, set the fields “Enable user’s nodes logging” and “Enable company’s nodes logging” to “Yes”

Manage profile (ID: 00112233445566778899-abc123def456 Company ID: 00112233445566778899-abc123def456)

Station: 0, Node: 1

First name:
Last name:
Password:
If you want to keep your current Password leave this field blank
Password confirmation:
If you want to change your Password, you must re-type it inside this field

Authentication Tokens

User Auth token: abcDefghIjKlmnopqrsTuvwx

Company Auth token: abcDefghIjKlmnopqrsTuvwx

Logging

Enable user's nodes logging: Yes
Enable company's nodes logging: Yes

5. Click on “Update User” to save the change
6. In the top menu, click on “Nodes”
7. Click on “New Node”
8. In the new dialog, enter the information related to the device:

New Node

Basic | Advanced

Name: Tuino Uno DevKit
DevEUI: 0011223344556677
DevEUI is byte value unique.

Class: CLASS_C

- a. Name: “Tuino 1 DevKit”
- b. DevEUI: Enter the DevEUI for the device
- c. Class: “CLASS_C”

Activity S.A. au capital de 744 810 € - 4 rue Ampère, 22300 Lannion, France
RCS St Brieuc 522 305 473, Siret 522 305 473 00012, TVA FR62522305473



9. Click on “Create Node”



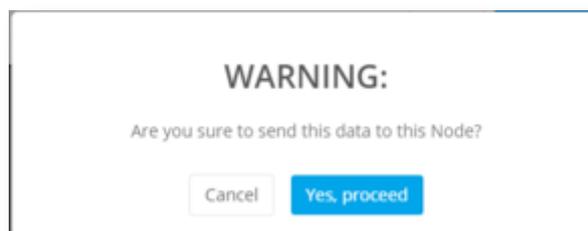
9.2 Send data to node

1. On the website note page (<http://thingpark.tuino.io> then Nodes from the top menu)
2. Find the device you want to send data too and press on the “Send data to Node” button
3. In the pop up dialog, enter the data in hexadecimal that want to be sent to the node. In this case, enter “0101” to turn on the relay on the devkit, or enter “0100” to turn it off



Data	Result
0100	Turns off the relay
0101	Turns on the relay
02xxxx	Set the thermostat temperature where xxxx is the temperature in Celsius centigrade (e.g.: 25.00 degC is 09C4)

4. Click on the “Send data to Node” and confirm in the pop up dialog by clicking on the “Yes, proceed” button

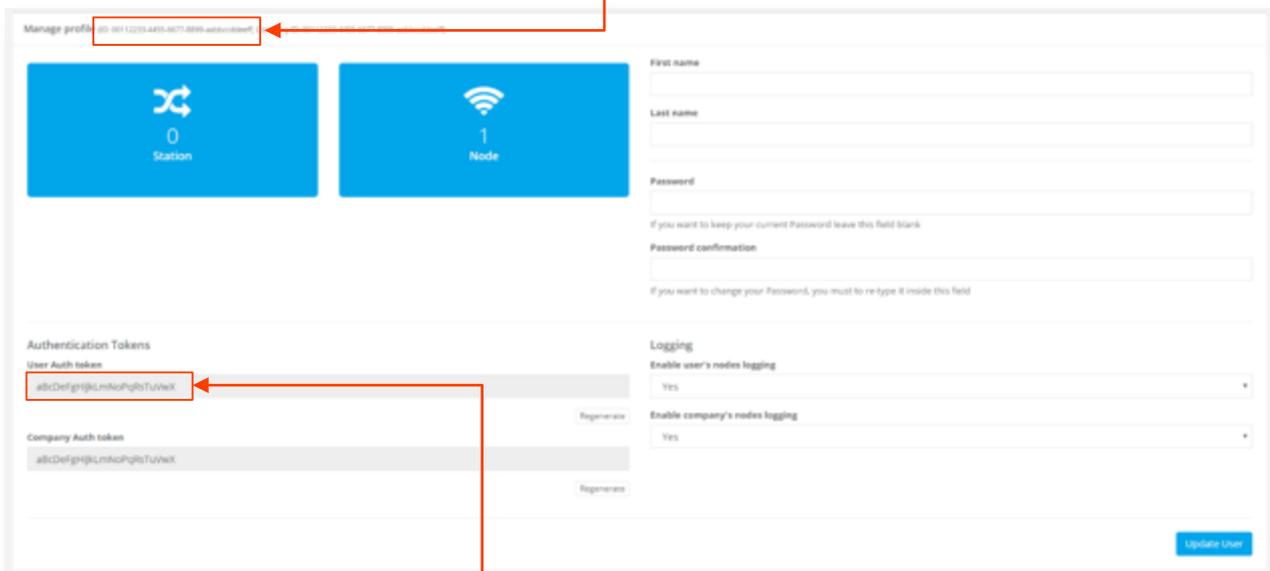


Note: This is usually the way to test that the connectivity works well between the various components that are part of the system. This is not how a final product will act.

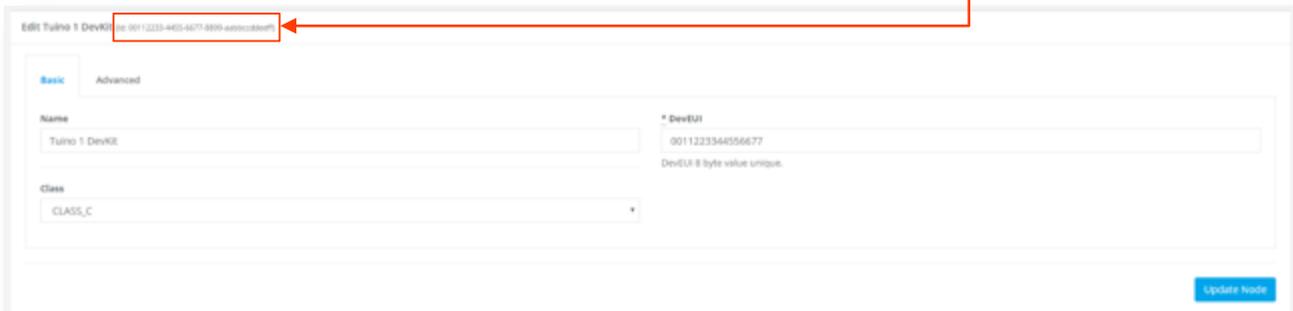


10 BUILD YOUR DEVKIT WEB APPLICATION

1. Download the code on your computer from:
<https://github.com/gimasi/TUINO-LPN-KIT-HTML-FRONTEND/>
2. Enter the **thermostat** directory and open the file for edition **configuration.js** in a text editor. This file contains the necessary information for the Web Application to connect properly to the Tuino Cloud.
3. On Tuino cloud, on the top right, click on the email address then click on “Manage profile”
4. In the new page, locate the ID next to the ‘Manage profile’ title and copy it into the configuration.js file as user_id.



5. Below, find the User Auth Token and copy it into the configuration.js file as auth_token
6. Ensure that Logging is enabled for both User nodes and company nodes
7. On the top menu, go to ‘Nodes’, find the device and click on the edit button ()
8. In the new dialog, next to the name of the device, the sensor id can be found. Copy it into the file configuration.js as sensor_id





9. The **configuration.js** file should look like this:

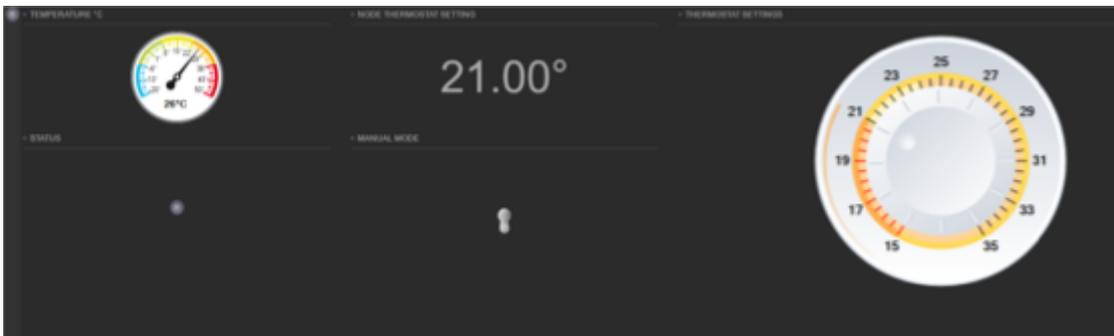
```
// Set This based on your setup
var auth_token = "aBcDeFgHiJkLmNoPqRsTuVwX";
var user_id = "00112233-4455-6677-8899-aabbccddeeff";

//
var sensor_id = "00112233-4455-6677-8899-aabbccddeeff";

// Sensor UPLINK interval in seconds
var beacon_timer = 60;
```

Note: Please make sure to have double quotes surrounding the IDs or it will not work.

10. Once the file **configuration.js** saved, the file **index.html** can be opened in a modern browser.



Note: For this example to work, the URL “beta.giotty.com” and the port 15674 have to be open for outbound connections.

Note: To find the latest for this example, go to <https://github.com/gimasi/TUINO-LPN-KIT-HTML-FRONTEND>. From there, it can be downloaded as zip or cloned using a git client.



11 TURN THE THERMOSTAT FOR A BOILER INTO A THERMOSTAT FOR A A/C UNIT

1. Go back to the Arduino IDE.
2. Find the function “void loop()”, then find the section that start with the following comment:

```
// do thermostat logic
```

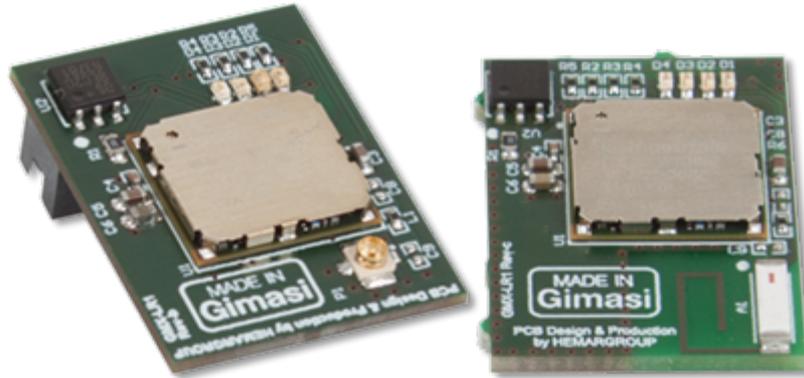
3. Underneath should be the code that handles the relay logic. Adapt it to change the behavior from turning on when the temperature is too low, to turning on when the temperature is too high.
4. Once the modifications have been made, click on the “Verify” button (the checkmark button), or go to the menu **Sketch | Verify/Compile**.
5. The program should check out (build) successfully. A message similar to this one should be displayed below the source code.

```
Sketch uses 28306 bytes (21%) of program storage space. Maximum is 130048 bytes.  
Global variables use 2372 bytes (14%) of dynamic memory, leaving 14012 bytes for  
local variables. Maximum is 16384 bytes.
```

6. Make sure that the Tuino 1 Development Kit is connected to the computer
7. In the menu, go to **Tools | Port** and select the port stating “Arduino” in its name
8. Click on the “Upload” button (the arrow button) to start the transfer process.
9. During the upload, the TX and RX LEDs next to the USB connector should be flashing intensely
10. Once the transfer is complete, the software on the Tuino 1 will reset. The display will then display the welcome message (“TUINO1 LoRa Thermostat”) and wait for the button to be pressed.



12 UPDATE YOUR LORAMAC FIRMWARE (ADVANCED)



The LoRaMAC module contains a firmware that can be updated to support new features (Regions, LoRaWAN specifications update ...) and bug fixes.

Firmwares for this Gimasi LR1 AT Modem can be found at:

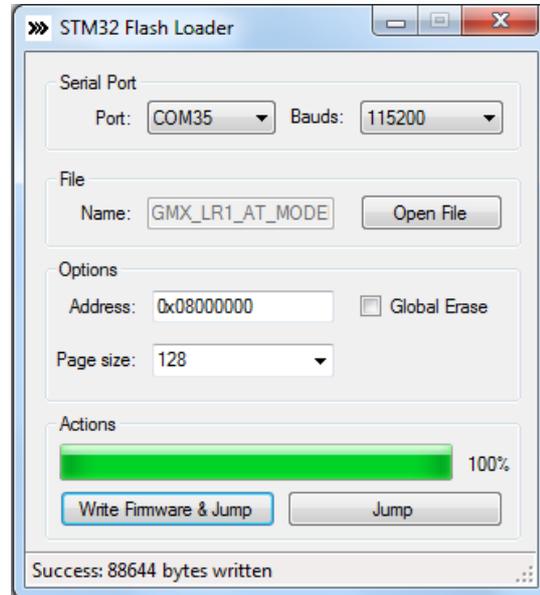
https://github.com/gimasi/GMX_LR1_AT_MODEM/releases

To update this module, make sure it is properly plugged into your Gimasi Development Kit and follow the following instructions:

1. Download the Arduino code to allow flashing of the LoRaMAC module:
https://github.com/gimasi/GMX_LR1_AT_MODEM
2. Download latest binary firmware at:
https://github.com/gimasi/GMX_LR1_AT_MODEM/releases
3. From the downloaded code in setp #1, open the file **firmware_update/tuino1-gmx-lr1-bootloader/tuino1-gmx-lr1-bootloader.ino** in your Arduino IDE and follow the procedure to flash this code into your Tuino 1 Development Kit.



4. Open the FlashLoader application located in the downloaded code at **firmware_update/STM32FlashLoader/STM32.Flash.Loader.exe**



5. In the Serial Port section, select the right port. This will depend on your computer. Make sure that the baud rate is set to 115200 bps.
6. In the Options section, make sure the Page size is set to 128 bytes.
7. In the File section, click on 'Open File' to browse where the latest firmware is located on your computer
8. Click on the 'Write Firmware & Jump' button.
9. Once the programming is completed, the module will reset and the LEDs on it will blink sequentially.
10. The Arduino code can now be modified/changed to fit your purpose.

Warning: *To use your Development Kit in another band than EU868, you need to set multiregion firmware instead of the default one. If this is the case, please follow the instructions above.*
