

LoRaWAN Mapping Node

Quick Start Guide

Contents

Mapper Node – Parts List.....3

Mapper Node – Construction.....4

Payload Decoder.....6

TTN Mapper Integration.....7

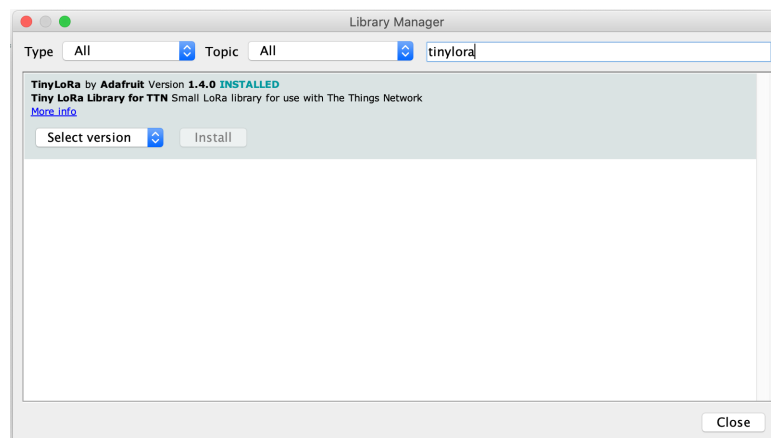
Mapper Node – Parts List

Part	Quantity	Amazon Link	Price
Arduino Uno	1	https://www.amazon.co.uk/Arduino-A000066-ARDUINO-UNO-REV3/dp/B008GRTSV6 (Clone - https://www.amazon.co.uk/ELEGOO-Board-ATmega328P-ATMEGA16U2-Cable/dp/B01EWOE0UU)	£20 - (£9 if using clone)
Draguino LoRa Shield	1	https://www.amazon.co.uk/Stemedu-Dragino-Compatible-Leonardo-consumption/dp/B07HF1S2QG	£24
Ublox Neo-6M GPS Receiver	1	https://www.amazon.co.uk/ICQUANZX-GY-NEO6MV2-Controller-Ceramic-Antenna/dp/B088LR3488	£8
Male to Female Dupont Style Wires	4	https://www.amazon.co.uk/Aukru-Male-Female-Jumper-Arduino-Raspberry/dp/B00MWMEIF2	£0.40 (Full pack £4)
		Total	£52.40 (£41.40 using clone)

* Prices correct at time of writing (14th July 2020), some parts may be swapped for cheaper alternatives, e.g. Arduino Uno could be swapped for an Uno Clone. Parts are probably cheaper from other sources although delivery times may be very long with eBay etc... as often they are shipped from Asia.

Mapper Node – Construction

- Slide the Draguino LoRa Shield onto the Arduino Uno observing the location of the pins and ensuring no pins become bent.
- Screw the LoRa antenna onto the antenna jack of the Draguino LoRa Shield
- Connect the GPS Receiver
 - VCC to 3.3v on the Draguino Shield
 - GND to ground
 - TX to Pin 4
 - RX to Pin 3
- Connect the GPS Antenna to the GPS receiver board.
- Connect the Arduino to a computer running the Arduino IDE using a USB A to USB B cable (some Arduino clones require USB A to USB Mini / Micro)
- Install the required libraries
 - From the menu select “Sketch > Include Libraries > Manage Libraries”
 - Search for and install the latest version of the TinyLoRa and TinyGPS++ libraries.



- Visit the Github repository and copy the mapper sketch into your Arduino IDE - <https://github.com/Green-Minds-Project/TTN-Mapping/blob/master/LoRa-GPS-Mapper/LoRa-GPS-Mapper.ino>
- Fill out the device id and the network session, application keys using values from The Things Network console (<https://console.thethingsnetwork.org>)

```
// Complete the next section with your device settings from https://console.thethingsnetwork.org
// Network Session Key (MSB)
uint8_t NwkKey[16] = { };

// Application Session Key (MSB)
uint8_t AppKey[16] = { };

// Device Address (MSB)
uint8_t DevAddr[4] = { };
```

- Click the Verify & Upload button, if all is well the compiler should complete without error and upload your code to the Arduino.
- The mapper node can now be tested, it works best outside, once the GPS has received data from a decent number of satellites the blue LED on the GPS module should flash intermittently. LoRa packets should be observed in The Things Network console every 15 seconds or so providing you are in range.

Payload Decoder

The raw data sent from the mapper node is an ASCII string encoded as bytes, these must be decoded via The Things Network before configuring the TTN Mapper integration. Login to The Things Network console (<https://console.thethingsnetwork.org>).

- Open your mapping application in TTN console.
- Navigate to the “Payload Formats” tab.
- Copy the decoder from Github (<https://github.com/Green-Minds-Project/TTN-Mapping/blob/master/TTN-Decoder/decoder.js>) into the decoder editor.
- Save the decoder
- Packets within the application’s data tab should now provide both the bytes encoded payload and the decoded payload including GPS location and accuracy data.

TTN Mapper Integration

Once the Payload Decoder has been configured the TTN Mapper integration can be enabled.

- Visit your mapping applications page in The Things Network console (<https://console.thethingsnetwork.org>)
- Navigate to the Integrations tab
- Click add integration
- Select TTN Mapper from the integration selector
- Complete the form, the only required field is the “Process ID” this should be something unique. The rest of the fields can be left blank
- Any valid GPS data received from your mapper node will now be used to populate the TTN Mapper database, it may take several hours for new data to show up on the map, for this reason it is suggested to also use the TTN Mapper app on your phone so you know when packets are being received to avoid spending time mapping dead areas as this provides more instant feedback.