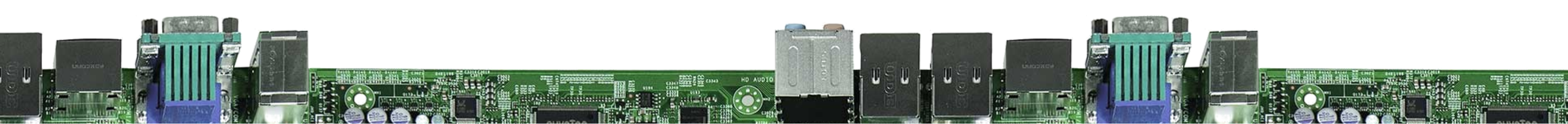
LoRaWAN Day 2019

18.10.2019

@holisticon @iothamburg



Schedule

- LoRaWAN theory
- The Things Network
- Hands-on DIY LoRaWAN Node based on Arduino
- Hands-on DIY LoRaWAN Gateway
- Official launch of the first gateway

LoRaWAN Theory

holisticon

Wireless network classifications

- Short-range wireless communication
  - NFC/RFID
  - ZigBee
  - Bluetooth / BLE
  - WiFi
- Cellular communications
  - 2G
  - 3G
  - 4G
  - 5G
- Low Power Wide Area Network (LPWAN)
  - **LoRa**
  - Sigfox
  - Narrow Band IoT (NB-IoT)

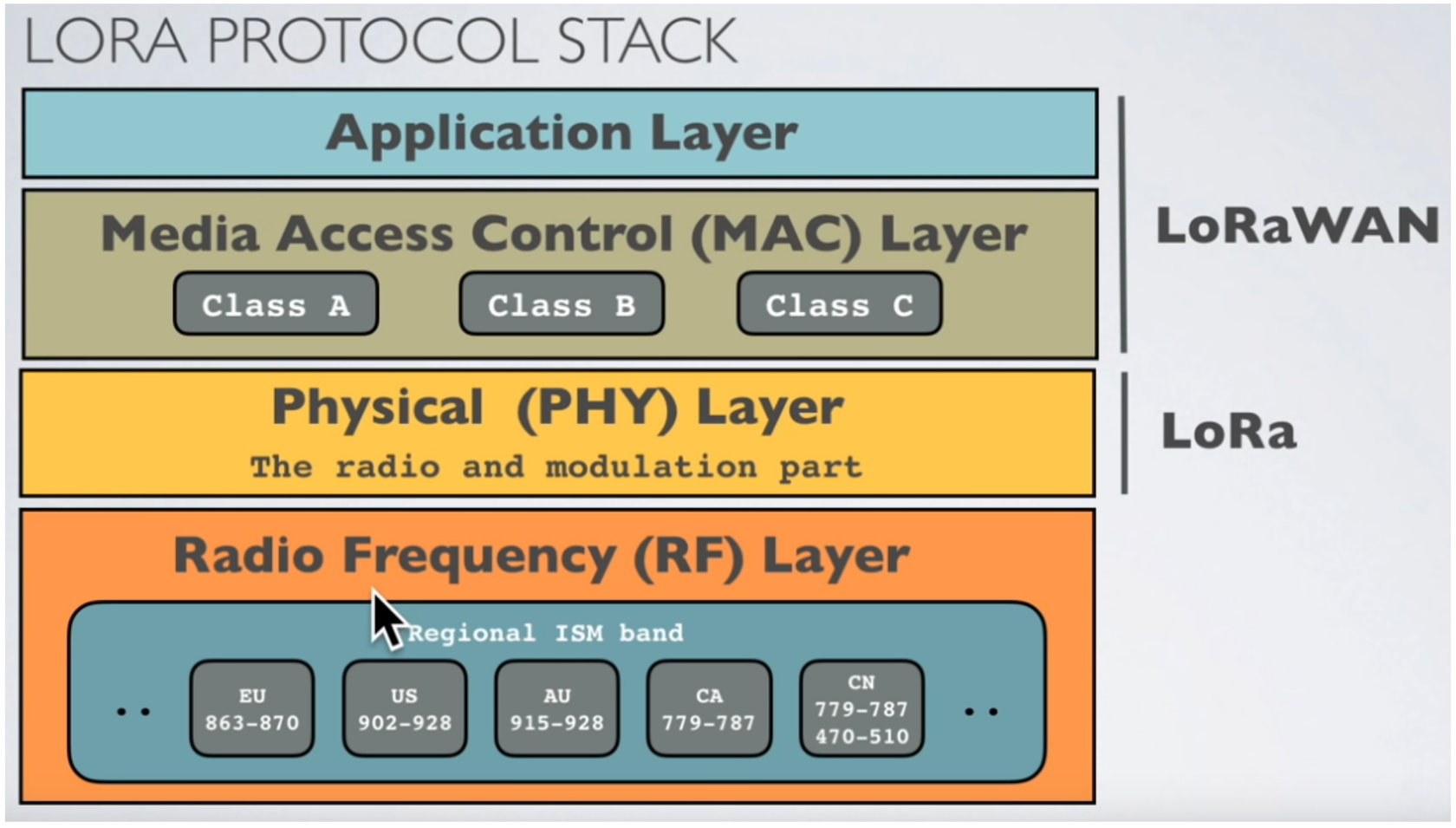
LoRa IP and licensing

- LoRa is developed by Cycleo, France
- in 2012, Cycleo acquired by Semtech Corporation
- Semtech licensed LoRa to chip manufacturers: HopeRF, Microchip, Dorji, etc...
- LoRa is a Trademark of Semtech Corporation filed 2015

LoRa Specs

- LoRa = Long Range
- Transfer rate: 0.3 - 5.5 kbps
- Tx power ~20 mW
- Range:
  - Urban: ~ 2-5 km
  - Countryside: ~ 5-15 km
  - Direct line of sight: > 15km

LoRa vs. LoRaWAN



https://lora.readthedocs.io/en/latest/

LoRaWAN

- Developed by LoRa Alliance
- Non-Profit, 500+ memebers

LoRaWAN Network

- Star topology
- Gateways, Nodes
- BiDi comm between Gateway and Nodes
  - uplink: from node to gateway
  - downlink: from gateway to node
- no direct Node-Node communication

LoRaWAN how does it work?

- Broadcast from Nodes to every Gateway
- Gateways forward packet to Network Server
- Network Server de-duplicates packets, finds nearest Gateways
- Network Server routes packet to Application Server
- (optional) Application server sends response
- (optional) Network Server routes to nearest Gateway
- (optional) Gateway brodcasts the response

LoRa Regulations

- unlicensed radio band (863-870MHz in EU)
- maximum uplink transmission power limited to 25mW (14 dBm)
- maximum downlink transmission power limited to 500mW (27 dBm)
- duty cycle: 0.1% or 1.0% dependening on channel

The Things Network

Building a global open LoRaWAN™ network.

https://www.thethingsnetwork.org/

holisticon

TTN Details

- Build world-wide network
- Open for everyone
- Free to use (Gateways and Nodes)
- Fair use:
  - uplink: 30 sec / day / node
  - downlink: 10 messages / day / node
- Integrations for own applications

Global Network

86926
MEMBERS

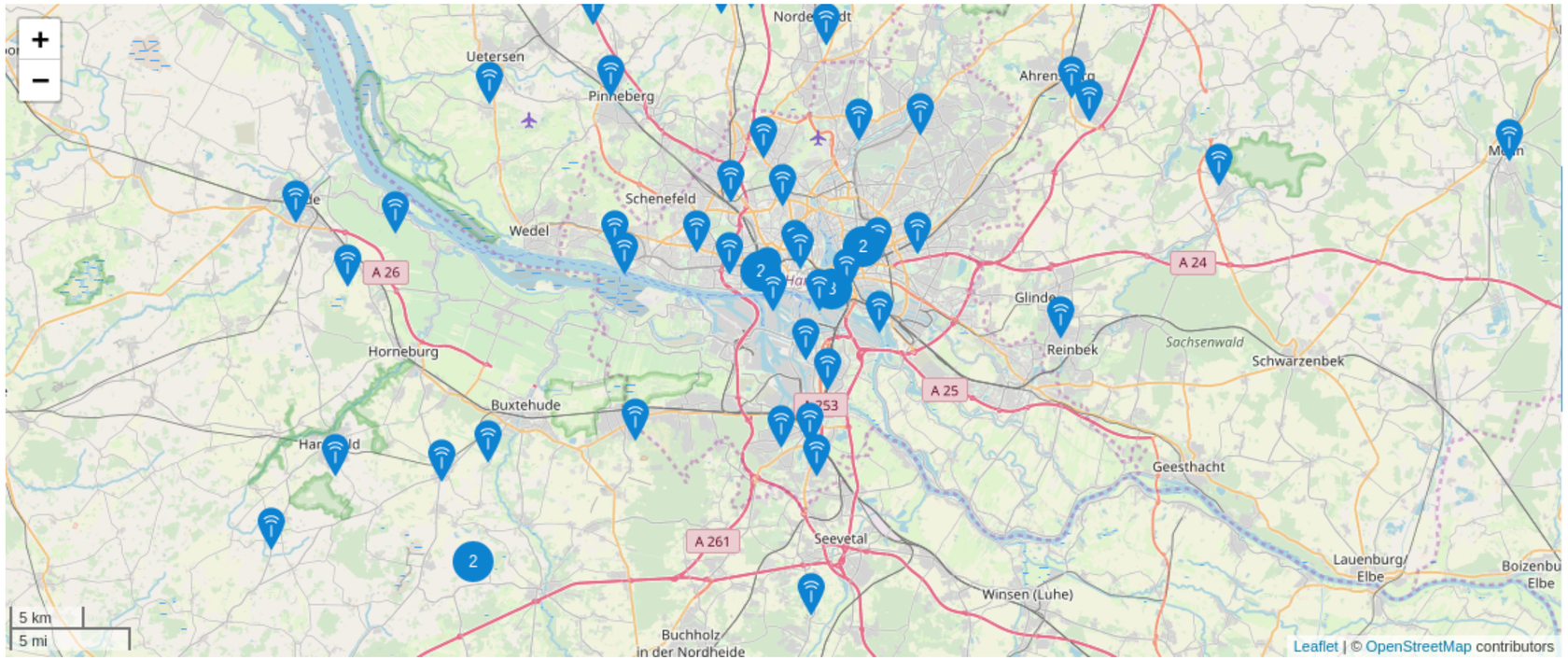
9226
GATEWAYS

141
COUNTRIES

https://www.thethingsnetwork.org/

holisticon

Hamburg+



https://www.thethingsnetwork.org/

holisticon

TTN Overview

- Devices
- Gateways
- Network
- Applications

https://www.thethingsnetwork.org/

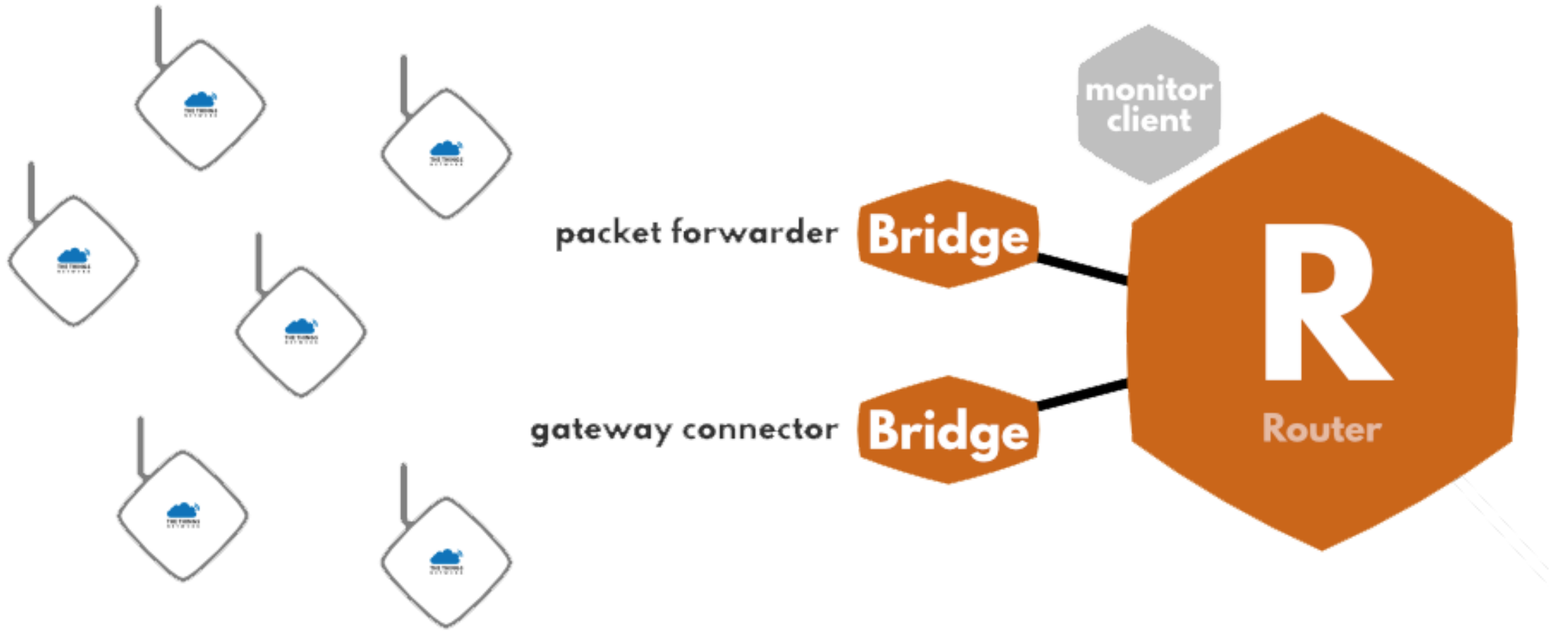
TTN | Devices

- also End Node, LoRa End Node, LoRa Node
- Device
  - ☐ Microcontroller (e.g. Arduino, ATMega, ATTiny)
  - ☐ LoRa radio module (e.g. RFM95B)
  - ☐ (optional) Sensors

TTN Device Activation

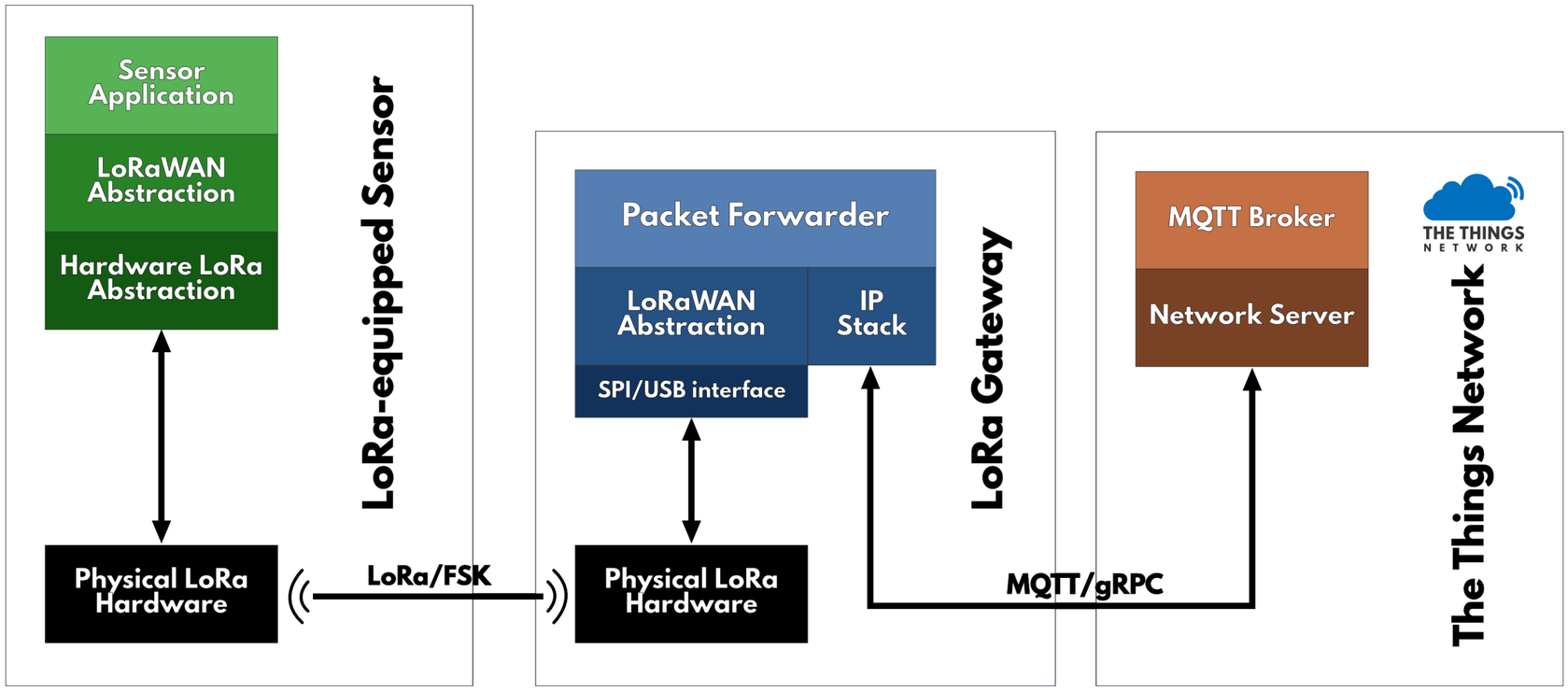
- Over The Air Activation (OTAA)
  - More secure, ready for production
  - Requires a downlink window to confirm activation
- Activation by Personalization (ABP)
  - Pre-configures keys, suitable for workshops and demos
  - Faster

TTN | Gateways | Overview



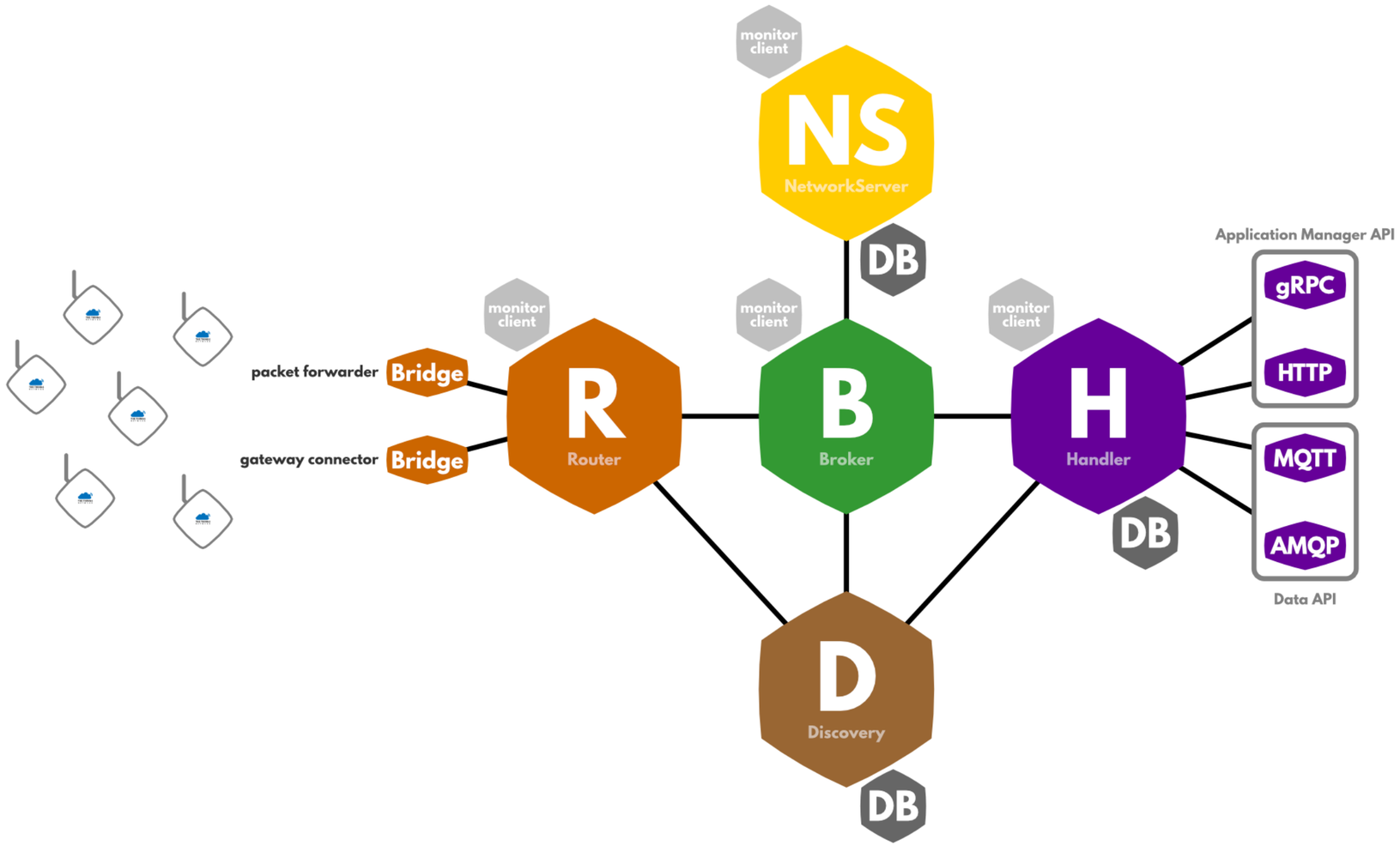
https://www.thethingsnetwork.org/

TTN | Gateways | Packet Forwarding



https://www.thethingsnetwork.org/

TTN | Network | Overview

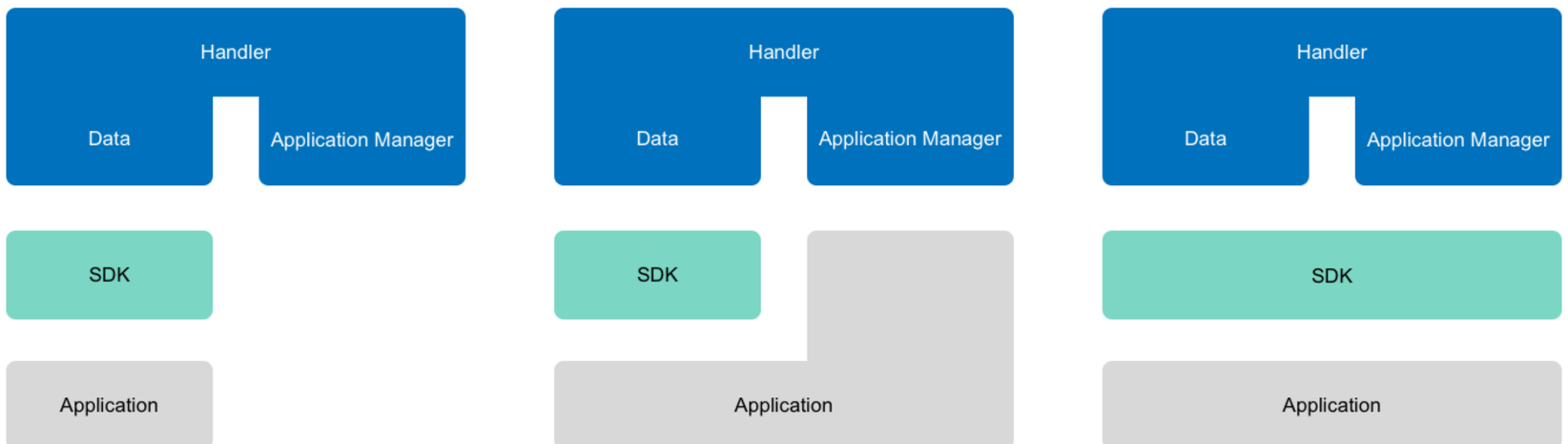


https://www.thethingsnetwork.org/

TTN | Applications

- Application Manager API
  - manage applications
  - manage devices
- Data API
  - receive data (downlink)
  - send data (uplink)

TTN | Applications | SDK Options



- Go
- Java
- Node-RED
- Node.js
- Python

https://www.thethingsnetwork.org/

TTN | Applications | Integration Options

holisticon

TTN | Applications | HTTP Options

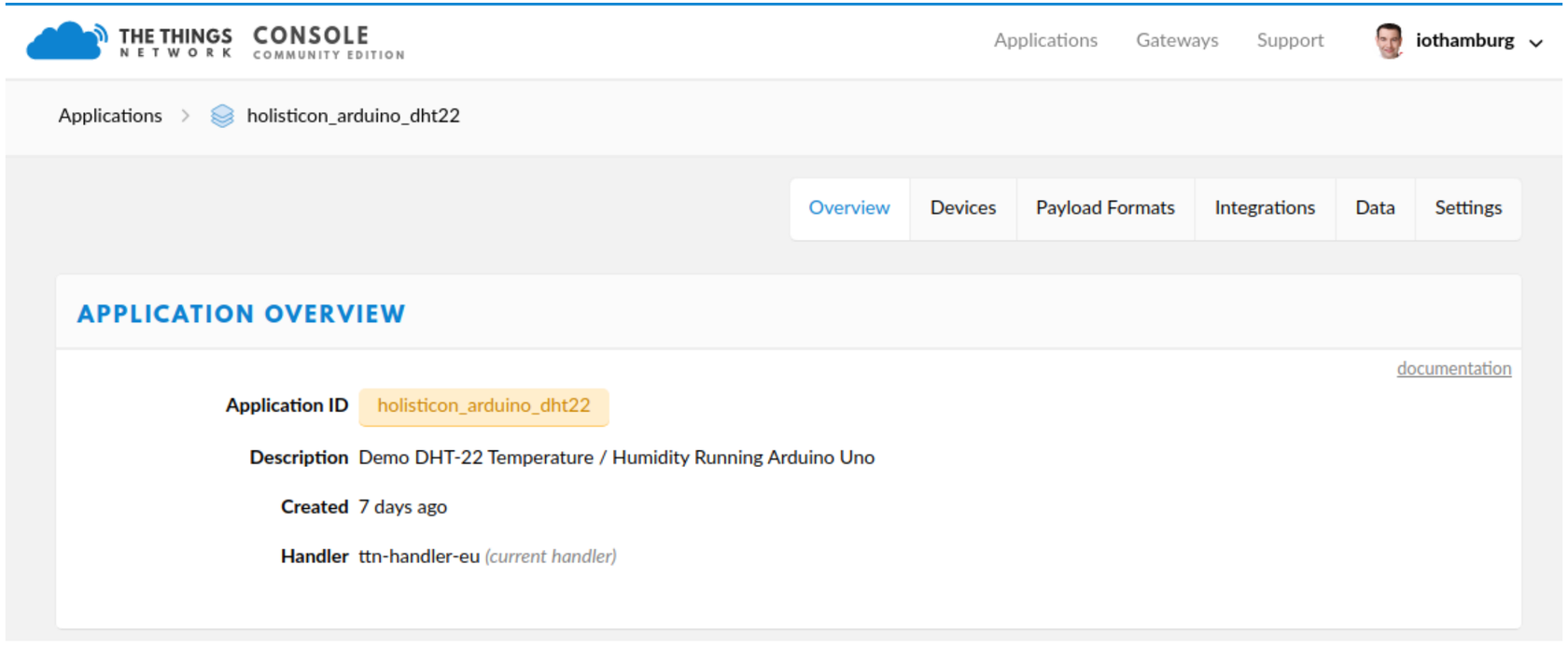
holisticon

TTN Application Integration

holisticon

- Data Integration API
  - read data via REST (native)
  - use TTN library
- HTTP Integration API
  - get informed via REST (POST)

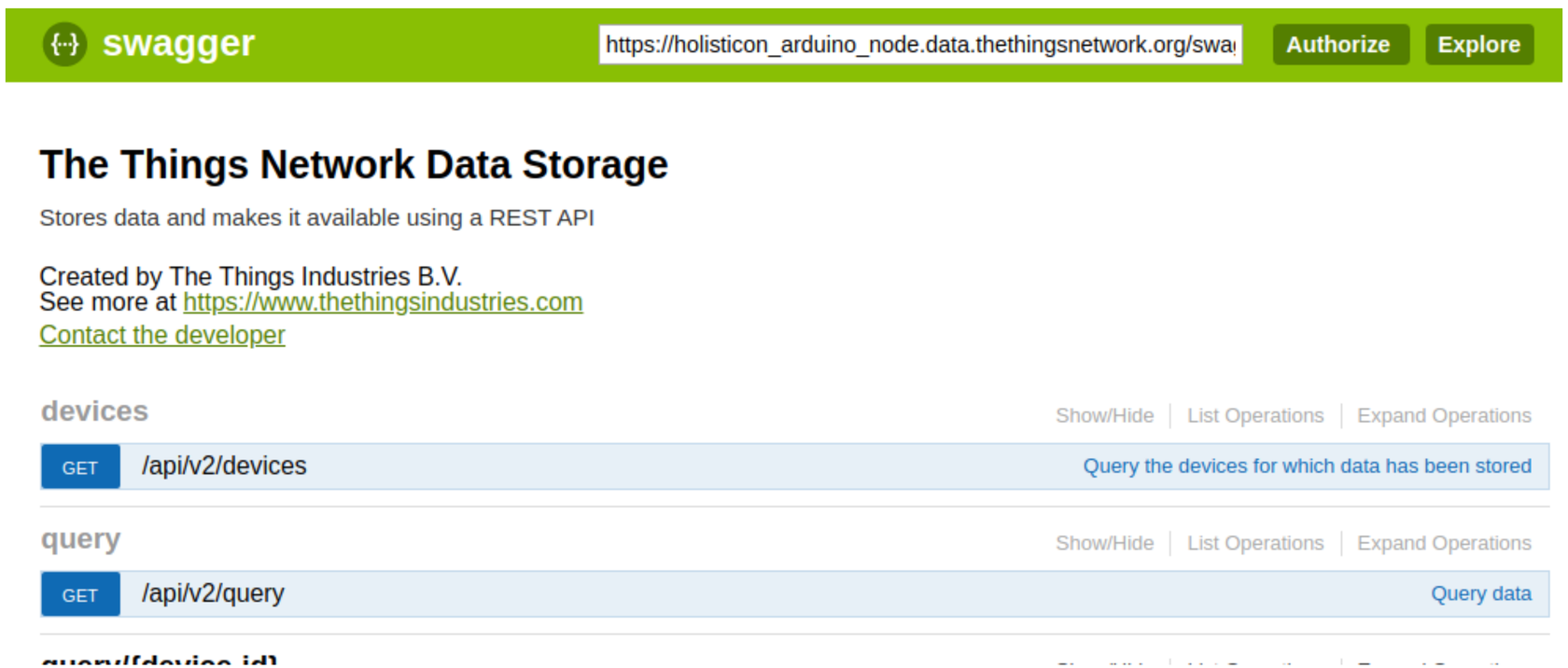
TTN DHT22 Demo Application



link

holisticon

TTN DHT22 Demo Console



link

holisticon

TTN Data Integration Example (ttn)

```
var ttn = require("ttn")
var appID = "" // app id
var accessKey = "" // app access key

ttn.data(appID, accessKey)
  .then(function (client) {
    client.on("uplink", function (devID, payload) {
      console.log("Received uplink from ", devID)
      console.log(payload)
    })
  })
  .catch(function (error) {
    console.error("Error", error)
    process.exit(1)
  });
```

TTN Data Integrtion Example (REST)

```
const express = require('express')
const https = require('https');
const cors = require('cors');
const app = express()
app.use(cors());
app.get('/data', function (req, res) {
  https.get({
      hostname: 'holisticon_arduino_dht22.data.thethingsnetwork.org',
      port: 443,
      path: '/api/v2/query',
      headers: {
        Accept: 'application/json', Authorization: 'key ....' // auth key
      }
  }, (resp) => {
    let data = '';
    resp.on('data', (chunk) => { data += chunk; });
    resp.on('end', () => { res.send(JSON.parse(data)); });

    }).on("error", (err) => { res.send(err); }
  );
});
```

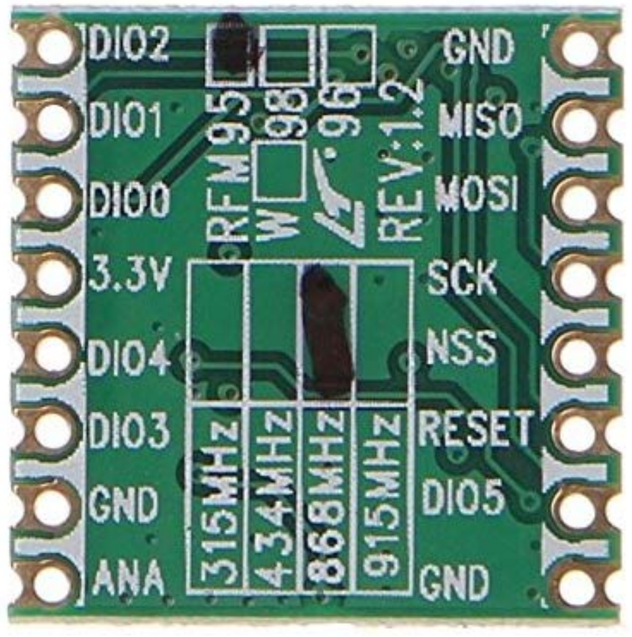
holisticon

LoRaWAN Hello World Node

holisticon

Hardware

- Arduino Uno
- RFM-95B (SX1276)



https://www.antratek.de/rfm95-lora-module

Wiring plan

holisticon

Radio connection plan

HopeRF RFM95	Arduino Uno	HopeRF RFM95	Arduino Uno
ANT	-	GND	-
GND	GND	DIO5	-
DIO3	-	RESET	5
DIO4	-	NSS	10
3.3V	3.3V	SCK	13
DIO0	2	MOSI	11
DIO1	3	MISO	12
DIO2	-	GND	-

Development

- Install Arduino IDE
- Install IBM LMIC Framework (https://github.com/matthijskooijman/arduino-lmic)
- Configure LMIC
- Customize `ttn-abp`
- Flush, Run..

Configure LMIC

~/Arduino/libraries/arduino-lmic-1.5.0-arduino-2/src/lmic/config.h

```
// If you live in Europe, set frequency.
#define CFG_eu868 1

// If you use a HopeRF RFM95 module, set chip model.
#define CFG_sx1276_radio 1

// If you want logging in the serial monitor, set log level.
#define LMIC_DEBUG_LEVEL 2

#define LMIC_FAILURE_TO Serial

// Disable this feature. It is not needed and space is freed up.
#define DISABLE_PING

// Disable this feature. It is not needed and space is freed up.
#define DISABLE_BEACONS
```

Register TTN application

- Open Console
- Create application
- Create device (activation = ABP)
- Configure decoder function (in Payload Formats)

```
function Decoder(bytes, port) {
  var result = "";
  for (var i = 0; i < bytes.length; i++) {
    result += (String.fromCharCode(bytes[i]));
  }

  return {text: result};
}
```

Configure TTN parameters and pins

- Open `ttn-abp`
- Change code below
- Select Tools > Board > Arduino Uno
- Select Tools > Port
- Compile, Flash, Restart Arduino
- Open Serial Monitor

```
// Network Session key (MSB)
static const PROGMEM u1_t NWKSKEY[16] = { 0x.. };

// Application Session Key (MSB)
static const u1_t PROGMEM APPSKEY[16] = { 0x.. };

// Device Address
static const u4_t DEVADDR = 0x.... ;

// Pin mapping
const lmic_pinmap lmic_pins = {
    .nss = 10,
    .rxtx = LMIC_UNUSED_PIN,
    .rst = 5,
    .dio = {2, 3, LMIC_UNUSED_PIN},
};
```

How does it work?

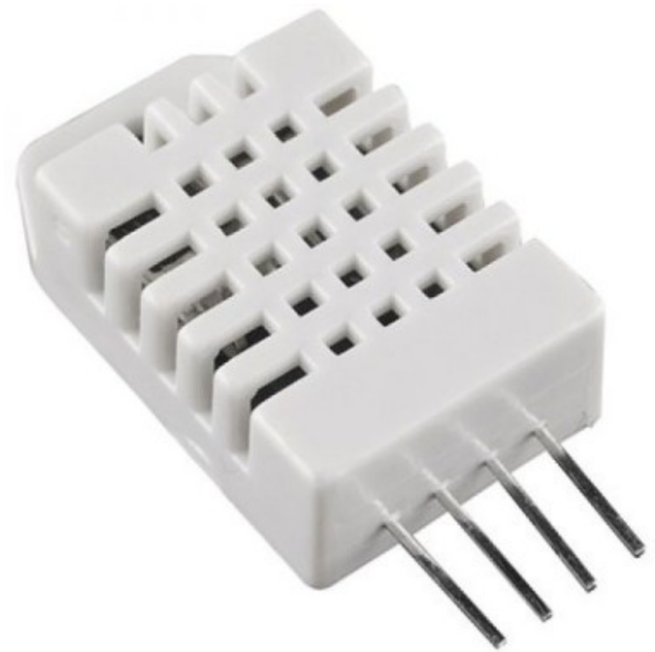
```
/**
 * Schedules send callback to every 60 seconds
 */
void onEvent (ev_t ev)
/**
 * Send callback, sends "hello world" string
 */
void do_send(osjob_t* j)
/**
 * Init LMIC, setting connection parameters
 */
void setup()
/**
 * Starts the loop
 */
void loop()
```

LoRaWAN DHT22 Node (incremental)

holisticon

Hardware

- Arduino Uno
- RFM-95B (SX1276)
- DHT-22 (AM2302)



https://tronixlabs.com.au/sensors/humidity/dht22-temperature-and-humidity-sensor-australia/

Wiring plan

holisticon

DHT22 connection plan

DHT22	Arduino Uno
VCC	3.3V
DATA	7
-	-
GND	GND

Additional installation

- Install Arduino DHT Sensor library

Configure TTN

- Create application
- Create device (activation = ABP)
- Create decoder function (in Payload Formats)

```
function Decoder(bytes, port) {
    if(bytes.length == 4) {
        var humidity = (bytes[0]<<8) | bytes[1];
        var temperature = (bytes[2]<<8) | bytes[3];
        return {
            'humidity': humidity/ 100,
            'temperature': temperature/100
        }
    } else {
        return {
            'error': 'payload unknown'
        }
    }
}
```

Configure TTN parameters and pins

- Open `ttn-abp-dht22`
- Change code below
- Compile, Flash, Restart Arduino
- Open Serial Monitor

```
/*
 * Define data pin of DHT sensor
 */
#define DHTPIN 7

/*
 * DHT11 or DHT22
 */
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// Network Session key (MSB)
static const PROGMEM u1_t NWKSKEY[16] = { 0x.. };

// Application Session Key (MSB)
static const u1_t PROGMEM APPSKEY[16] = { 0x.. };

// Device Address
static const u4_t DEVADDR = 0x.... ;
```

How does it work?

```
/**
 * Schedules send callback
 **/
void onEvent (ev_t ev)
/**
 * Send callback -> send temp and humidity
 */
void do_send(osjob_t* j)
/**
 * Init LMIC,
 * init DHT
 **/
void setup()
void loop()
```

LoRaWAN Gateway

holisticon

Hardware

- iC880A-SPI concentrator board
- Pigtail for antenna
- Raspberry Pi 3B
- Antenna
- Power Supply 2.5A with micro USB
- MicroSD Card
- RPi to iC880a interface (backplane from tindie)

ic880A-SPI

holisticon

ic880A-SPI to PI wiring

iC880a pin	Description	PI pin
21	Supply 5V	2
22	GND	6
13	Reset	22
14	SPI CLK	23
15	MISO	21
16	MOSI	19
17	NSS	24

Preparation of PI

- Install Raspbian
- Enable SPI
  - `sudo raspi-config`
  - `Interfacing options -> P4 SPI`
  - `Advanced options -> A1 Expand filesystem`
- Configure Timezone and Locale
  - `sudo dpkg-reconfigure locales`
  - `sudo dpkg-reconfigure tzdata`
- Get updates and get `git`
  - `sudo apt-get update`
  - `sudo apt-get upgrade`
  - `sudo apt-get install git`

Installation on PI

- Create `ttn` user
  - `sudo adduser ttn`
  - `sudo adduser ttn sudo`
- Get installer
  - `git clone -b spi https://github.com/ttn-zh/ic880a-gateway.git ~/ic880a-gateway`
  - `cd ~/ic880a-gateway`
  - `sudo ./install.sh spi`
- Write down the Gateway EUI

Gateway configuration on TTN

- Login to Console
- Register new gateway
- Enable `legacy packet forwarder`
- Enter Gateway EUI (printed by installer)
- Select EU (868MHz)
- Confirm

Links

- https://github.com/ttn-zh/ic880a-gateway/wiki

References

MobileFish Tutorials

- https://www.mobilefish.com/download/lora/lora_partX.pdf
  (where X =[1..36])
- Quick guide

holisticon

Heise Check Letter on ATTiny84

ATTiny84

- Quick Guide ATTiny
- attiny-base@github
- pin layouts

Check Letter

- mailbox@github

holisticon