

LoRa und TheThingsNetwork

Hubert Högl

Hubert.Hoegl@hs-augsburg.de

<http://hhoegl.informatik.hs-augsburg.de>

Hochschule Augsburg

21. April 2018

Das „Internet der Dinge“

(*engl.: IoT – Internet of Things*)

Beispiele

- Umweltsensoren, z.B. Temperatur, Feuchte, Luftschadstoffe
- Bewässerung
- Glascontainer
- Mausefalle
- Stadtbeleuchtung

Anforderungen

- Geringe Datenraten (Sensorwerte, Schaltbefehle) — n bytes/Tag
- Batteriebetrieb oder „energy harvesting“
- Muss lange ohne Wartung laufen (Jahre)
- Grosse Reichweite (km)

Technische Ansätze

LPWAN – Low Power Wide Area Networks

Die Wichtigsten:

- Mobilfunk (GSM), LTE bzw. „4G“ (LTE Cat-M1, LTE Cat-NB1 = „NB-IoT“, EC-GPRS)
- Sigfox (ISM Band)
- **LoRa, LoRaWAN, TheThingsNetwork** (ISM Band)

Einige weitere:

- Ingenu
- Weightless
- 802.11ah low-power WiFi

LoRa (1)

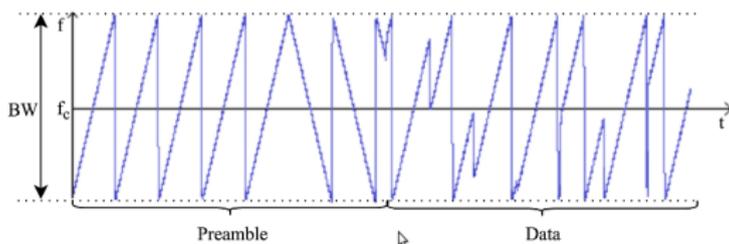


- Cycleo (Frankreich/Grenoble) entwickelt ca. 2009 LoRa Funktechnik und wird von Semtech (USA) 2012 aufgekauft (<https://www.semtech.com>).
- Patentierte 6/2014
Einige Details der LoRa Funktechnik sind **closed source**. Siehe die Arbeit von Matt Knight, „Decoding the LoRa PHY“ (mit SDR-Methoden).
- Chirp Spread Spectrum (CSS)
- ISM Band, EU 868 MHz (863 bis 870 MHz), brutale Funkumgebung
- Kanäle, z.B. bei 125 kHz Bandbreite 868.1, 868.3, 868.5, ... MHz.
Spezifiziert in LoRaWAN Regional Parameters.

LoRa (2)

CHIRP (auch in der Radartechnik und bei Fledermäusen)

- Upchirp
- Downchirp



- Daten werden durch die Sprünge codiert.
- Funktioniert auch bei bewegten Geräten ($> 100\text{km/h}$)

LoRa (3)

Parameter der CSS Funktechnik

- **Sendeleistung** -3 dBm – +15 dBm; EU Norm max. +14 dBm. Das sind 25 mW bzw. ca. 40 mA Stromaufnahme.
- **Spreizfaktor** SF7 – SF12 (7 bis 12 Bits pro Symbol). Entkoppelt Bandbreite von der Datenrate (Bitrate).
klein: hohe Bitrate (bis 50 kbit/sec), schlechteres Signal-Rausch Verhältnis (SNR), kürzere Time-on-Air, geringere Stromaufnahme.
gross: kleine Bitrate (bis 200 bit/sec), besseres SNR, längere Time-on-Air (bis 0.5 sec), mehr Stromaufnahme.
- **Bandbreite** BW 125 bis 500 kHz
klein: hohe Empfindlichkeit
gross: geringe Empfindlichkeit
- **Bitrate** Variabel zwischen 200 bit/sec bis 50 kbit/sec. Wird über den Spreizfaktor eingestellt. Bandbreite bleibt konstant.
- **Codierrate** CR: Fehlerkorrektur Bits CR1 – CR4

LoRa (4)

Der Spreizfaktor bestimmt Bitrate, Reichweite und Time-on-Air.

Spreading factor (at 125 kHz)	Bitrate	Range (indicative value, depending on propagation conditions)	Time on Air (ms) For 10 Bytes app payload
SF7	5470 bps	2 km	56 ms
SF8	3125 bps	4 km	100 ms
SF9	1760 bps	6 km	200 ms
SF10	980 bps	8 km	370 ms
SF11	440 bps	11 km	740 ms
SF12	290 bps	14 km	1400 ms

(with coding rate 4/5 ; bandwidth 125KHz ; Packet Error Rate (PER): 1%)

LoRa (5)

- Kein „Listen-before-Talk“, deshalb 1% Regel im ISM Band: 1% duty cycle → max. 30s pro Stunde.
- Man kann mit zwei LoRa Modulen eine Funkstrecke aufbauen und mit allen Parametern experimentieren (z.B. Microchip RN2483). Bastelvorschlag: Arduino + RN2483:
<https://github.com/jpmeijers/RN2483-Arduino-Library>



LoRaWAN (1)

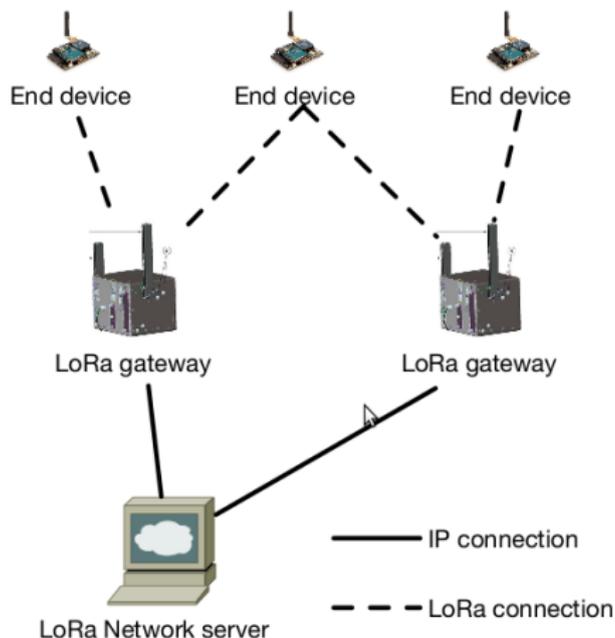


- <https://lora-alliance.org>
- LoRa = **PHY**, LoRaWAN = **MAC**
- Aktuelle Version: 1.1 (ca. 100 Seiten).
Achtung: TTN verwendet zur Zeit noch 1.0 (2015)

LoRaWAN (2)

Architektur

Endgeräte („devices“, „motes“), Gateways („concentrators“, „base stations“), Network-Server, Application-Server.



LoRaWAN (3)

Endgeräte-Aktivierung „join“ und Sicherheit

Zwei verschiedene Aktivierungen

- OTAA – Over The Air Authentication (sicher, erfordert downlink)
- ABP – Authentication By Personalization (einfach, unsicher)

Sicherheit

Drei 128-Bit Schlüssel (AES-128)

- AppKey (in Gerät und Applikation)
- AppSKey (privat)
- NwkSKey (im Netz bekannt)

AppSKey und NwkSKey werden bei der OTAA Aktivierung auf das Gerät übertragen. Bei ABP werden sie im Gerät im Quelltext gespeichert.

<https://www.thethingsnetwork.org/docs/lorawan/security.html>

LoRaWAN (4)

Endgeräte-Klassen

Jedes Endgerät muss Class A können, B und C sind optional. TTN 1.0 kann nur Class A.

- **Class A – All End-Devices**

Uplink-Übertragung mit zwei anschließenden Zeitfenstern (1s, 2s) für den Empfang. Für Batteriebetrieb sehr gut geeignet.

- **Class B – Beacon**

Empfangsfenster zu bestimmten Zeitintervallen, damit Netzwerk-Server einen Downlink initiieren kann. Für Batteriebetrieb gut geeignet.

- **Class C – Continuously Listening**

Für Endgeräte die immer empfangen müssen. Nicht für Batteriebetrieb geeignet.

LoRaWAN (5)

Software für Endgeräte

- Referenz-Implementierung von Semtech (STM32)
<https://github.com/Lora-net/LoRaMac-node>
- LoRaWAN MAC für Embedded Systems in C von IBM Zürich:
<https://github.com/mcci-catena/ibm-lmic> (Eclipse Public License)
- Port für Arduino (AVR)
<https://github.com/matthijskooijman/arduino-lmic>
- LoRaWAN Erweiterung für STM32Cube
<http://www.st.com/en/embedded-software/i-cube-lrwan.html>

LoRaWAN (6)

Gateway (I)

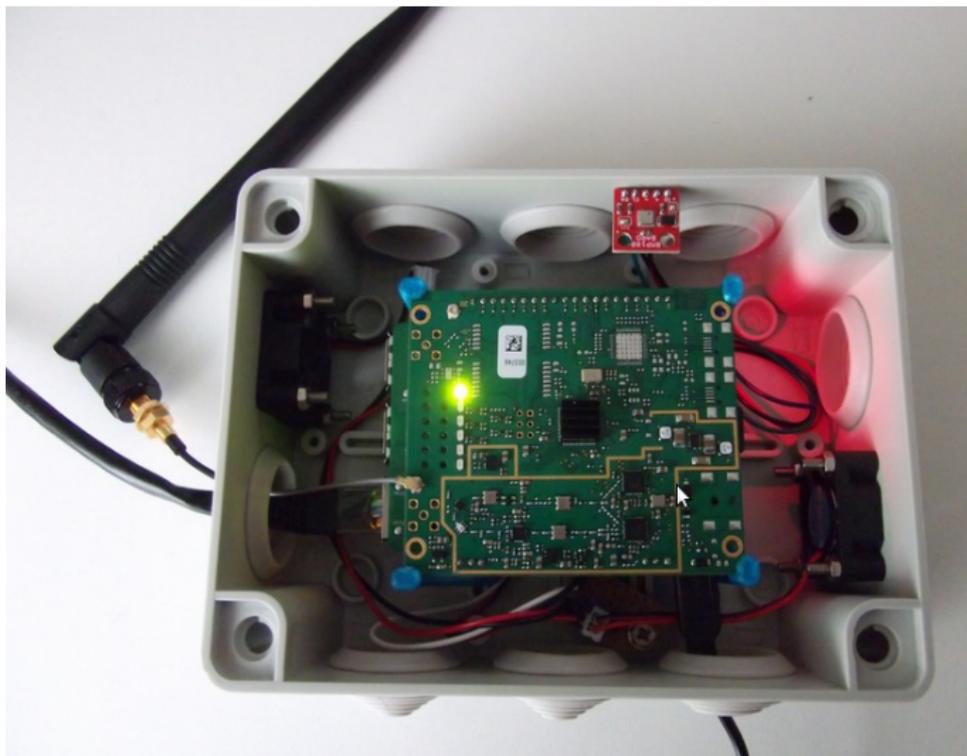
- LoRaWAN Concentrator iC880A von Imst GmbH. Mehrere Kanäle mit unterschiedlichen Spreizfaktoren.
- In Verbindung mit Raspberry Pi oder Beagle Bone
- Software:
<https://github.com/Lora-net>



Fertiges Gateway von imst.de: **WIMOD Lite Gateway** für 199 Euro (auch mit RPi und iC880A). Kaum Preisunterschied zu Selbstbau.
<https://wireless-solutions.de/products/starterkits/lora-lite-gateway.html>

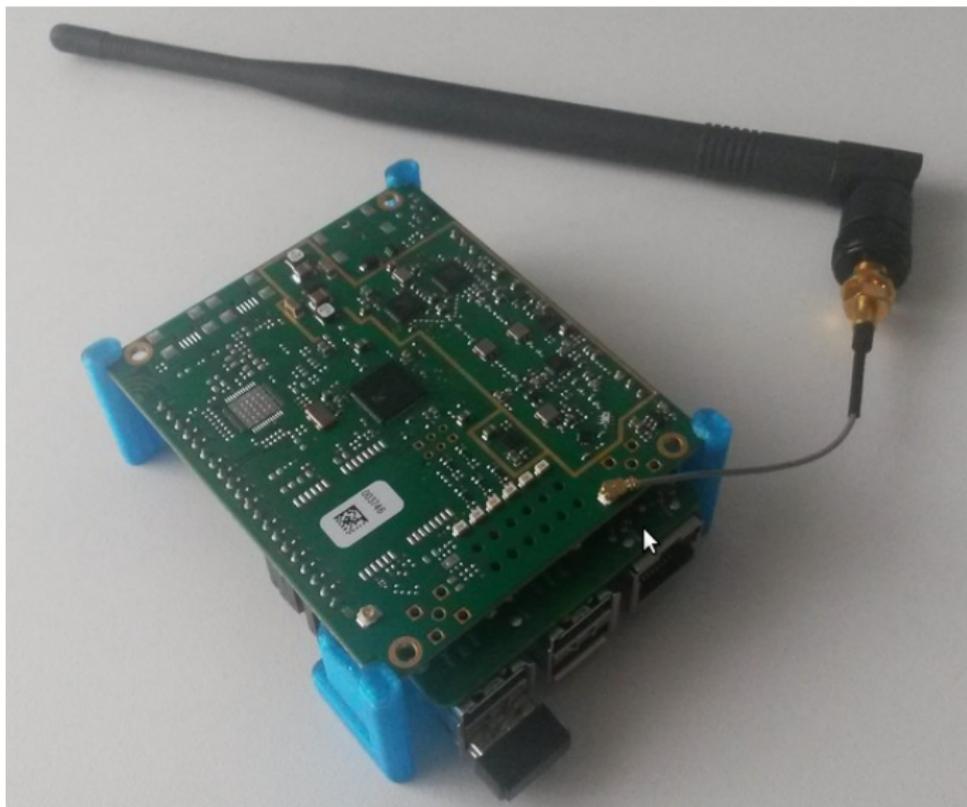
LoRaWAN (7)

Gateway (II): RPi2, iC880A, Halterungen (3D Druck), Temperatursensor, 2 Lüfter, Lüfterregelung, Gehäuse, LoRa Antenne (ca. 200 Euro).



LoRaWAN (8)

Gateway (III)



LoRaWAN (9)

Gateway (IV): Groundplane Antenne, 868 MHz



LoRaWAN (10)

Netzwerk-Server („network stack“)

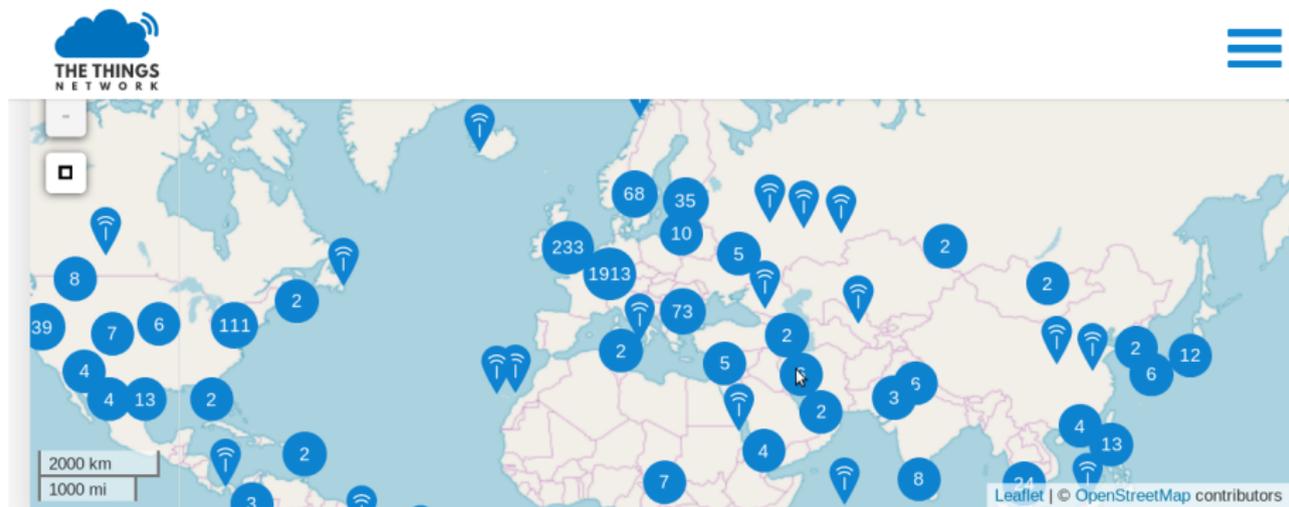
Wertet RSSI, SNR, SF der empfangenen Pakete aus. Kann SF der Endgeräte ändern, je nach Qualität der Funkverbindung.

- LoRaWAN network server (von Semtech - gegen NDA)
- TheThingsNetwork <https://github.com/TheThingsNetwork/ttn>
- <https://github.com/gotthardp/lorawan-server>
- <https://www.loraserver.io>
- <https://www.resiot.io>
- Loriot.io

The Things Network (1)

<https://www.thethingsnetwork.org>

„You are the network“



The Things Network (2)

- Vier **Gemeingüter**: (a) Offener Standard (b) Freie Software (c) Offene Hardware (Kickstarter) (d) offenes, netzneutrales, von der Gemeinschaft betriebenes Netzwerk.
- **Lizenzen**: GPL, CERN Open-Hardware License, CC
- **Open-Source** <https://github.com/thethingsnetwork> (Go!)
- Versionen des Network Stack
 - ▶ V2: 2017, LoRaWAN 1.0, wird zur Zeit verwendet, keine Class-B/Class-C
 - ▶ V3: 2/2018, LoRaWAN 1.1 und 1.0, Class-B und Class-C möglich. Auch Peering und Roaming mit anderen Netzen.
- TTN bietet auch **Geräte** zum Kauf an
 - ▶ The Things Uno
 - ▶ The Things Node
 - ▶ The Things Gateway
- Kommerzieller Ableger: <https://www.thethingsindustries.com>

TheThingsNetwork (2)

TTN fair use

- Golden rule: 30 seconds air-time per device per day
- For 10 bytes of payload, this translates in (approx.):

20 messages per day at SF12

500 messages per day at SF7

more for SF7BW250 and FSK (local-area)

- If your application requires more bandwidth, think of another solution

This allows for >1000 nodes per gateway

Downlink bandwidth is even more restricted

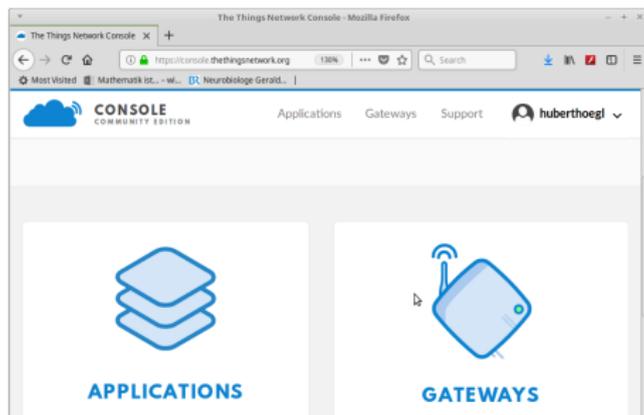
you can't send all messages as "confirmed uplink"

The Things Network (3)

Administration

Zunächst Account auf TTN anlegen.

Web-Anwendung <https://console.thethingsnetwork.org>, oder ...



... Kommandozeilenprogramm `ttnc1`

<https://www.thethingsnetwork.org/docs/network/cli>

TheThingsNetwork (4)

Konfigurierbar sind Anwender, Kollaborateure, Gateways, Applikationen und registrierte Geräten.

GATEWAY Nr. 1 (ID=eui-b827ebfffe1be004)

APPLICATIONS N ----- 1 USER 1 ----- N GATEWAYS
(+ COLLABORATORS)

APPLICATIONS 1 --- N DEVICES

AppEUI DevEUI (= MAC Adresse), keine Verwendung
OTAA: AppEUI, AppKey
ABP: DevAddr, NwSKey, AppSKey

The Things Network (6)

The screenshot shows a web browser window titled "The Things Network Console - Mozilla Firefox". The address bar shows the URL "https://console.thethingsnetwork.org/ap...". The page header includes the "CONSOLE COMMUNITY EDITION" logo, navigation links for "Applications", "Gateways", and "Support", and a user profile for "huberthoegl". The breadcrumb navigation path is "Applications > labor_ws > Devices > arduino_dht". The main content area is titled "DEVICE OVERVIEW" and displays the following information:

- Application ID: labor_ws
- Device ID: arduino_dht
- Activation Method: ABP
- Device EUI: 34 34 34 34 56 56 56 56

TheThingsNetwork (7)

Integration

Wie komme ich von aussen an die Daten?

- TTN bietet **Payload Functions** (JavaScript), **HTTP Integration**, **Data Storage** (7 Tage Persistenz + API)

`https://www.thethingsnetwork.org/docs/applications/integrations.html`

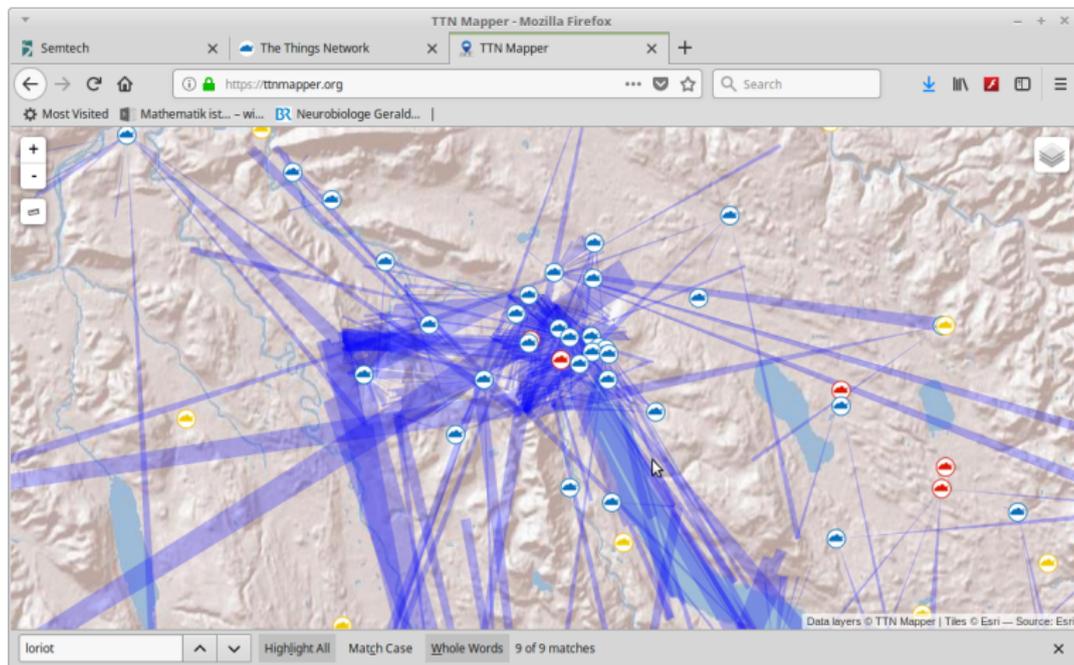
`https:`

`//www.thethingsnetwork.org/docs/applications/storage`

- Cayenne
`https://mydevices.com`
- IFTTT Maker
`https://platform.ifttt.com/maker`
- AllThingsTalk
`https://www.allthingstalk.com`
- Evrythng, OpenSensors, Azure IoT Hub, AWS IoT, ...

The Things Network (6)

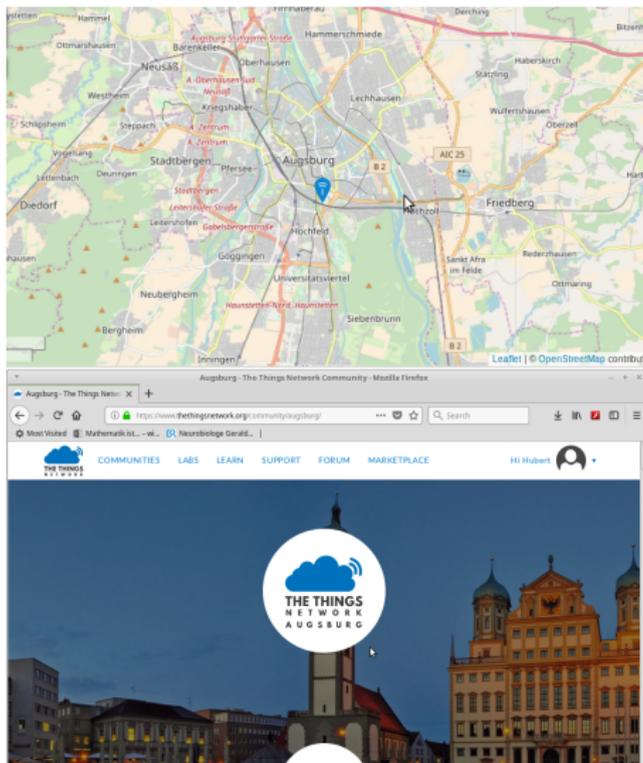
<http://ttnmapper.org> (man braucht LoRa Gerät und Smartphone zur Ortsbestimmung)



Zürich hat eine vollständige TTN-Abdeckung

The Things Network (7)

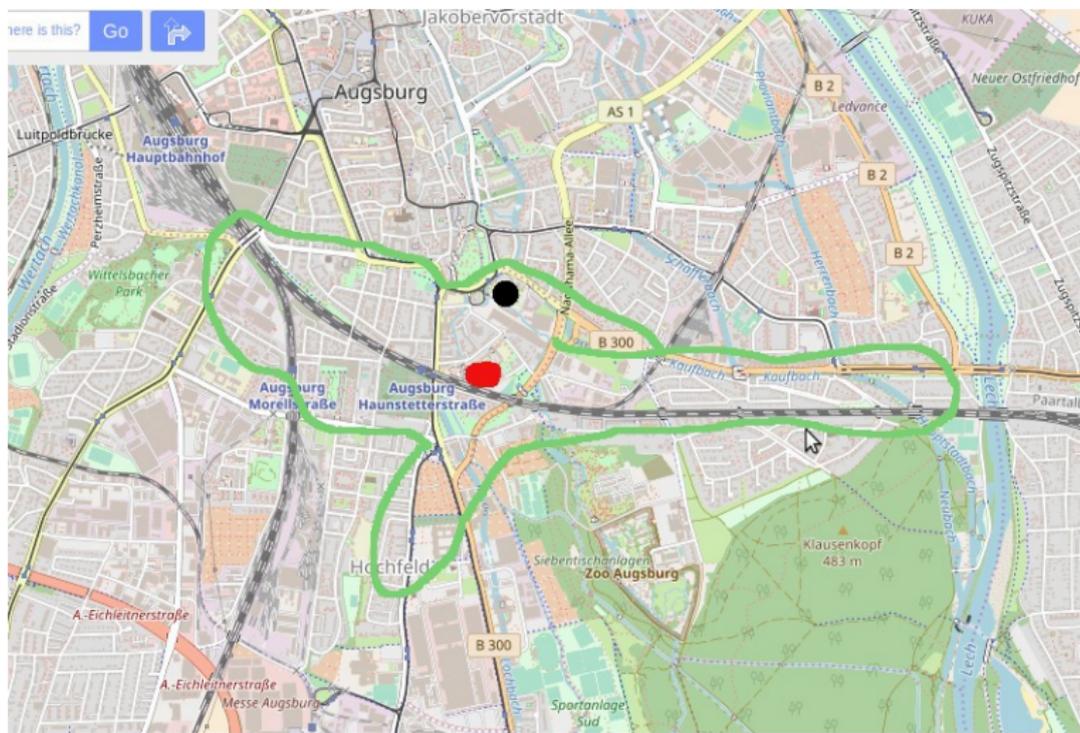
Augsburger Community



The Things Network (8)

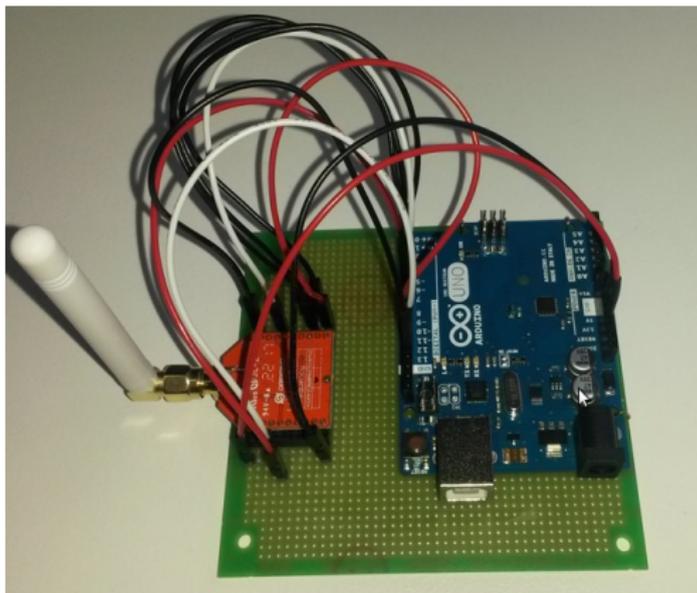
Reichweite

rot: Gateway auf E-Technik Hochhaus, schwarz: M-Bau, LIT2018



Unsere Testgeräte (1)

Dragino und Arduino



IBM LMIC auf Arduino:

<https://github.com/matthijskooijman/arduino-lmic>

Unsere Testgeräte (2)

LoPy auf Expansion Board



Unsere Testgeräte (3)

LoPy

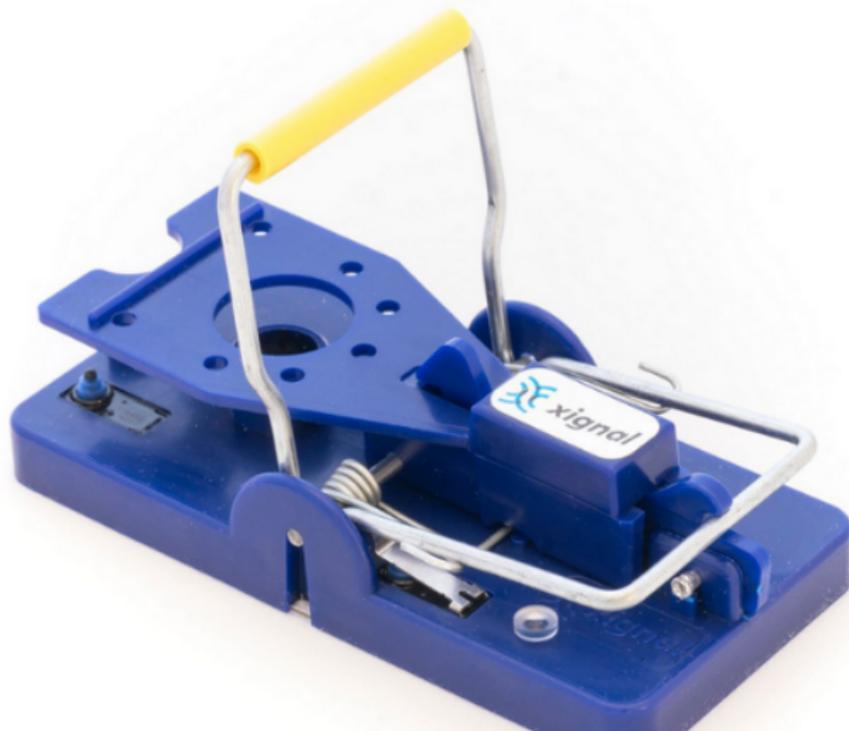
- <https://pycom.io/hardware>
- ESP32 + Semtech SX1272
- 160MHz CPU, 512K RAM, 4 MByte Flash
- Programmierbar mit Python 3 (<http://micropython.org>)
- Sehr wenig Code

```
lora = LoRa(mode=LoRa.LORAWAN, region=LoRa.EU868)
dev_addr = struct.unpack(">l", binascii.unhexlify('26011BEE'))[0]
nwk_swkey = binascii.unhexlify('0C5ACBB5F46C48CE110EF6C20C430A09')
app_swkey = binascii.unhexlify('80846A7ED4E027D02464F6FC7A84AB48')
lora.join(activation=LoRa.ABP, auth=(dev_addr, nwk_swkey, app_swkey))
s = socket.socket(socket.AF_LORA, socket.SOCK_RAW)
s.setsockopt(socket.SOL_LORA, socket.SO_DR, 5)
s.setblocking(True)
s.send(bytes([1, 2, 3]))
data = s.recv(64)
```

- Anleitungen
<https://www.thethingsnetwork.org/docs/devices/lopy/usage.html>
<https://docs.pycom.io>

Mausefalle

<https://www.signal.com/products/xignal-mousetrap>



Ausblick

- Spannend: TheThingsNetwork wird in diesem Jahr auf LoRaWAN V1.1. umgestellt.
- Wann gibt es ein zweites Gateway im Augsburger Raum?
- Vortrag von Nicholas Sornin (Lora Erfinder), „The future of LoRa“
<https://www.youtube.com/watch?v=jNnPTxWRNxs>

Danke fürs Zuhören!

Fragen?

(Gerne auch später per E-mail an Hubert.Hoegl@hs-augsburg.de)

Siehe auch

<http://hhoegl.informatik.hs-augsburg.de/hhwiki/LoRa>