Technische Universität Dresden

Marius Trappe, TU Dresden

OpenSensSystem: Using RIOT for a Universal Sensor System for District Heating

03.09.2026

Disclaimer

- I study Renewable Energy Systems
- I don‘t write our software (other than maybe a driver tomorrow)

→ Meaning: Little computer science background

But somehow this project is my brainchild



Definition District Heating

- We measure up to entire districts
- Mostly neighbourhoods
- Very high location resolution
- Very high time resolution



© Markus Arendt

Our Requirements

Data acquisition

- Measure temperature, humidity, condensation, …
- Self-sufficient operation for more than 2 years while measuring every minute
- Automatic configuration of the sensors

Data transmission

- Wireless with a range of more than 100m through a building
- Collection at a central data collector
- Near real time (about every 5 minutes)

In the future

- also control Actuators

OpenSensSystem





OpenSensGate
Gateway

formatting and forwarding to cloud services (if needed)



LoRaWAN uplink for Data, Maintainance and OTA-Updates



optional configuration



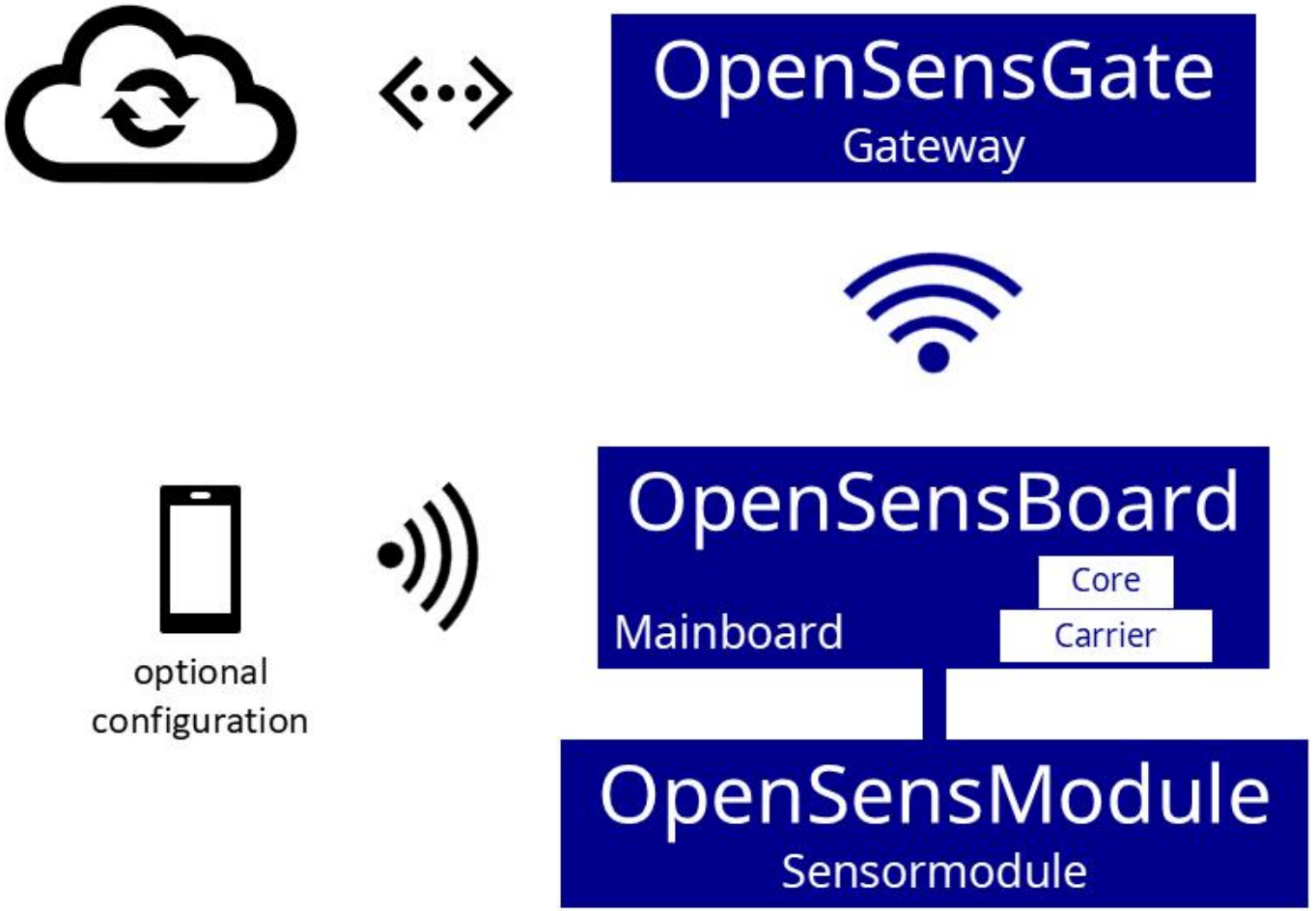
OpenSensBoard
Mainboard
Core
Carrier

Bridge between wireless uplink and wired sensors, special setup via Wifi/Bluetooth

OpenSensModule
Sensormodule

Specifically tailored sensors, automatic configuration

OpenSensSystem





OpenSensModule

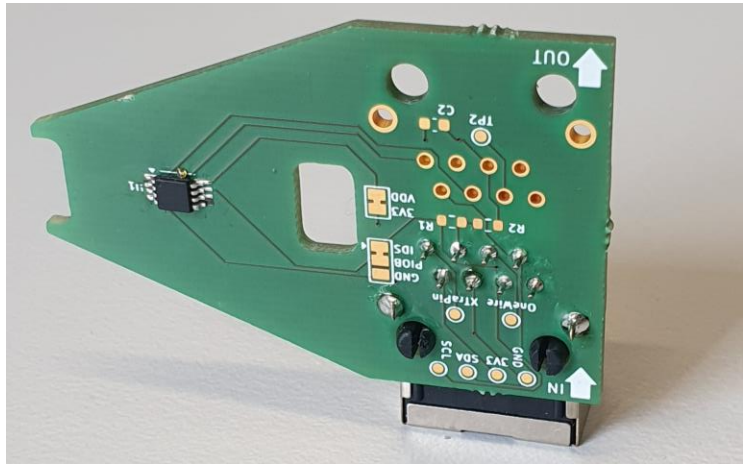
- Carries sensor
- Marker for automatic configuration
- Use of standard connectors (RJ45)

OpenSensBoard

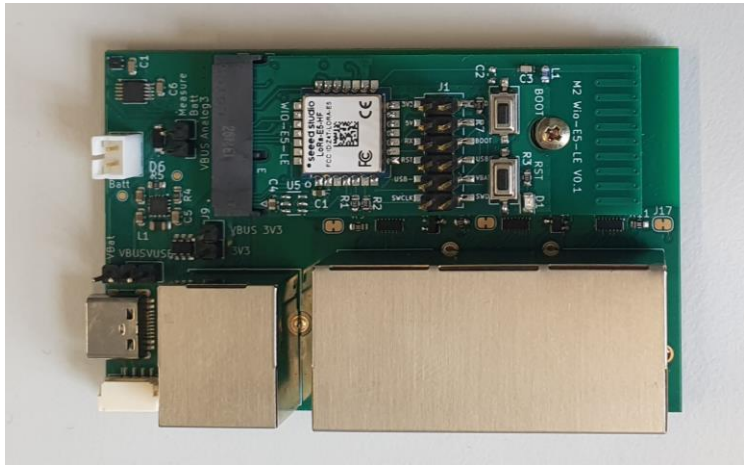
- Collects sensor data
- Handles data processing
- Provides Uplink to gateway

OpenSensGate

- Provides network
- Formats data



DS18B20 OpenSensModule © Marius Trappe



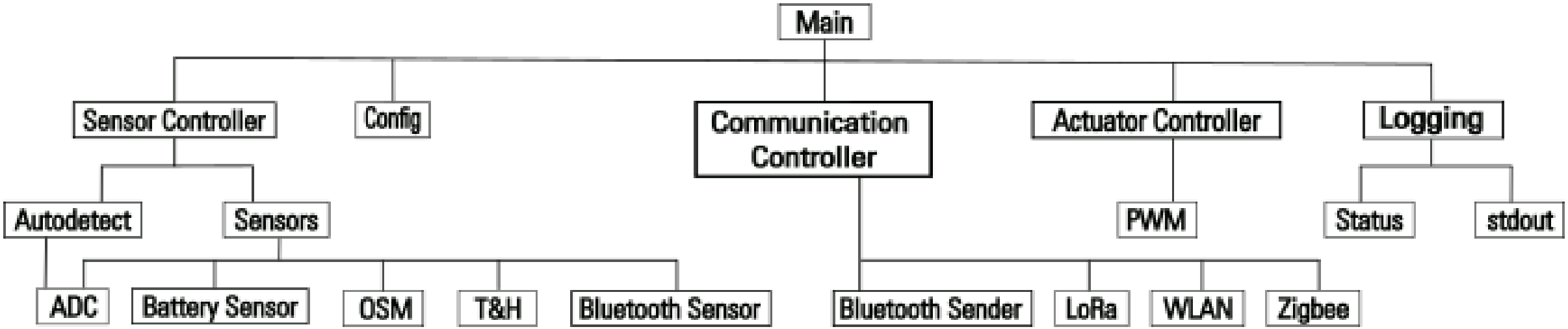
OpenSensBoard STM32WLE5 © Marius Trappe



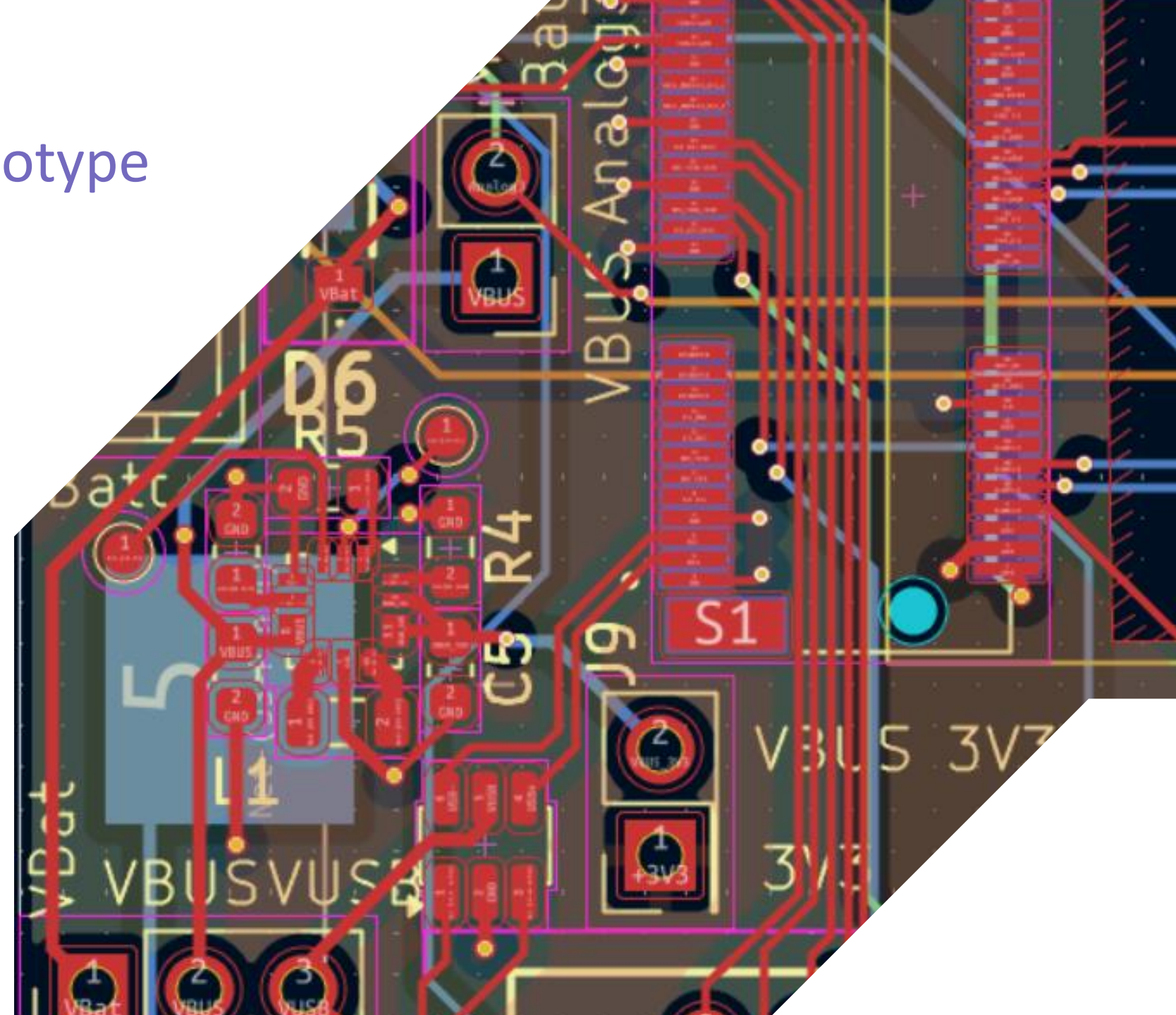
LoRaWAN Gateway © Marius Trappe

OpenSensBoard Firmware

Architecture:



What It‘s Like to Prototype with RIOT



Our Experience

First contact with RIOT

06/2025

- Started looking into RIOT
- Went to Hack‘n‘Ack at TUD

10/2025

- New coworker
- Start of firmware development with RIOT



MCU Triangle

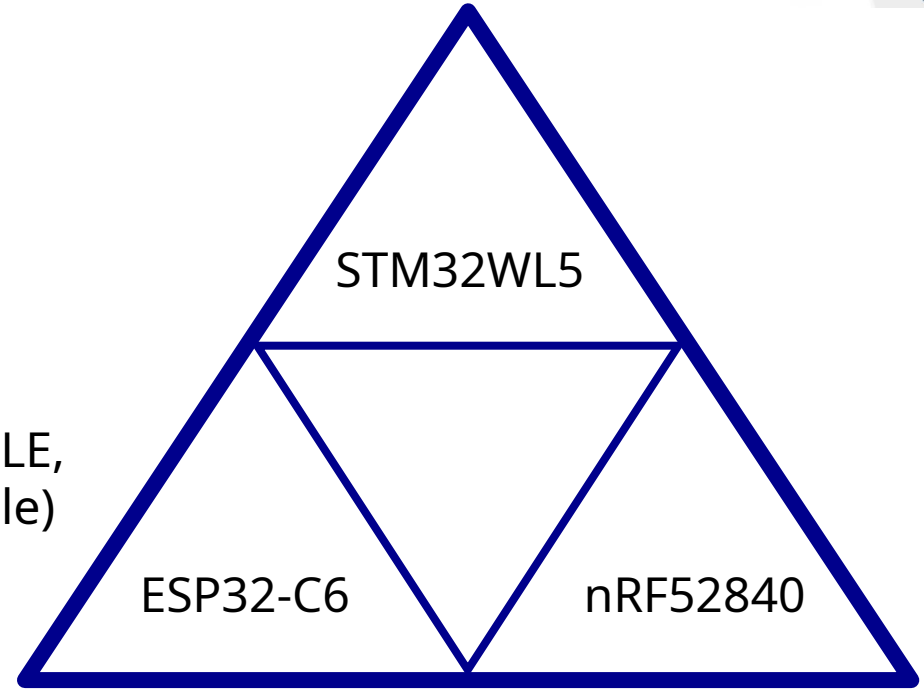
Only LoRaWAN

Energy efficient









Wifi, IEEE 802.15.4, BLE, LoRaWAN (via Module)

Tasmota-capable

Many peripherals

IEEE 802.15.4, BLE, LoRaWAN (via Module)

Energy efficient

Even more peripheral capabilities

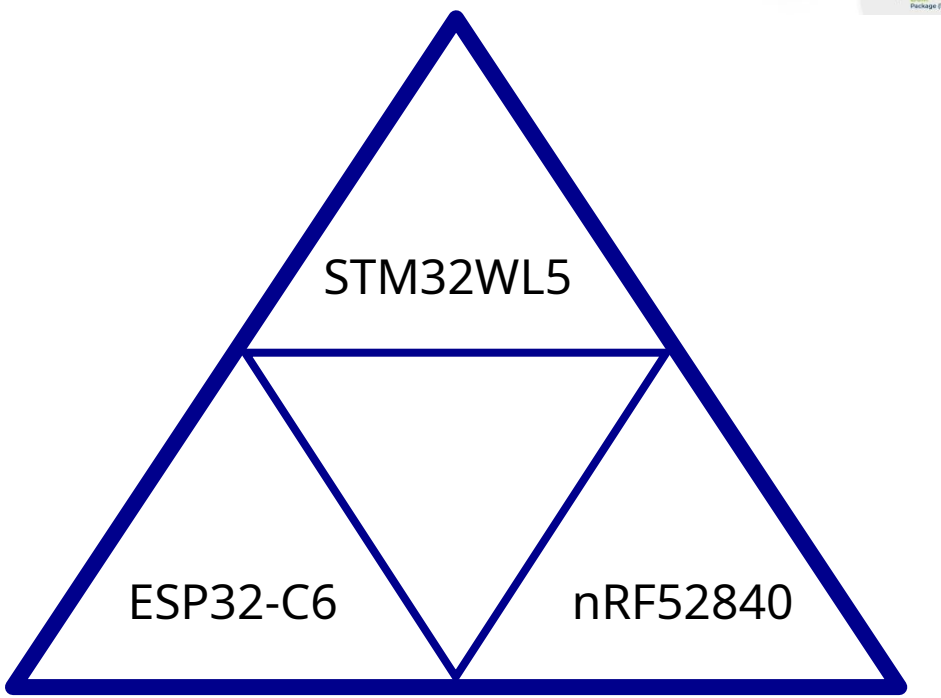
MCU Triangle

Not enough pins
(on Module)









Power Consumption

LoRaWAN ✗

sth. sth. LoRaWAN

Arduino ✓
Riot ✗

LoRaWAN and RIOT

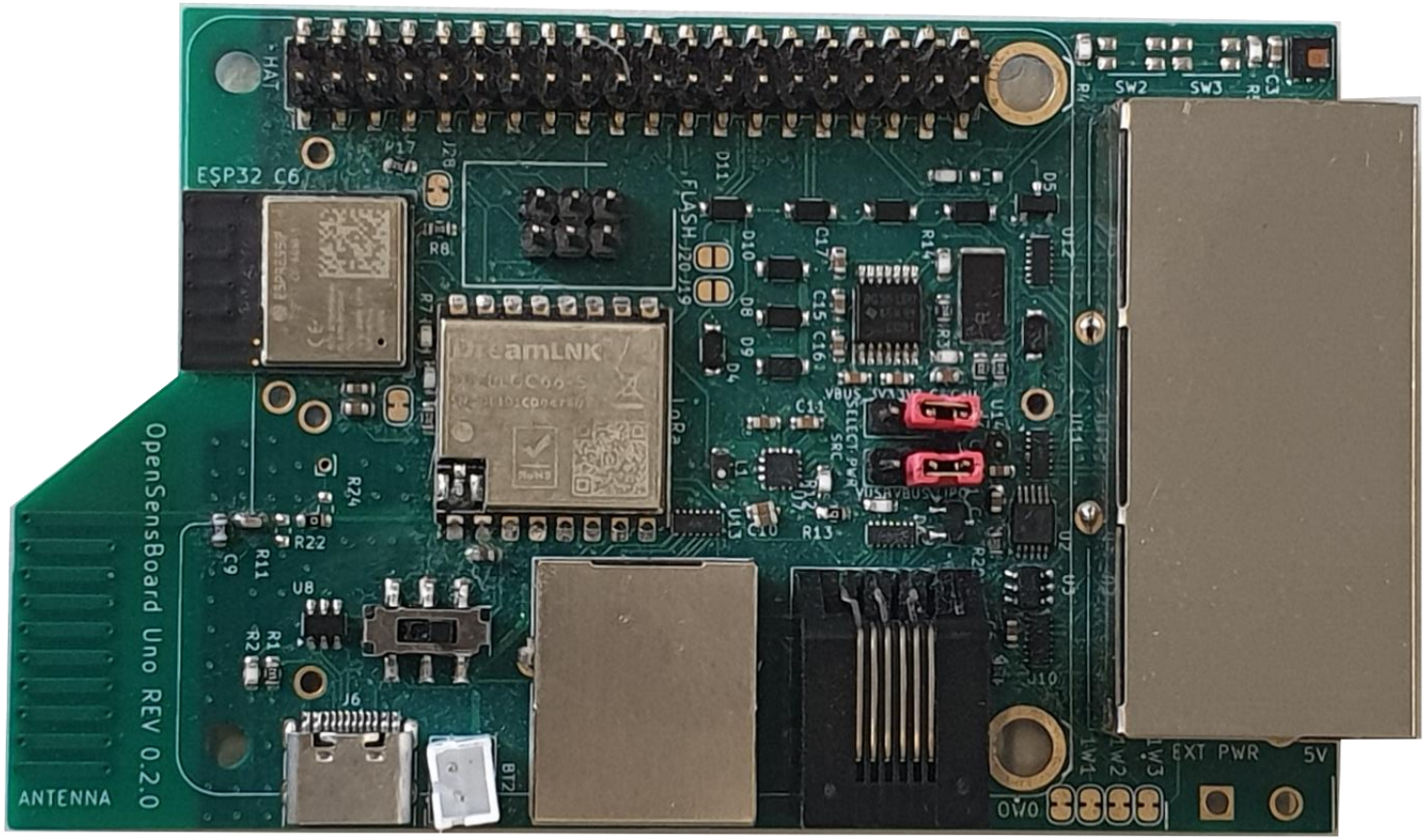
loramac

ESP32-C6 with LLCC68

→ didn‘t work (at all)

ESP32-C6 with SX1262

→ also didn‘t work



LoRaWAN and RIOT

GNRC LoRaWAN

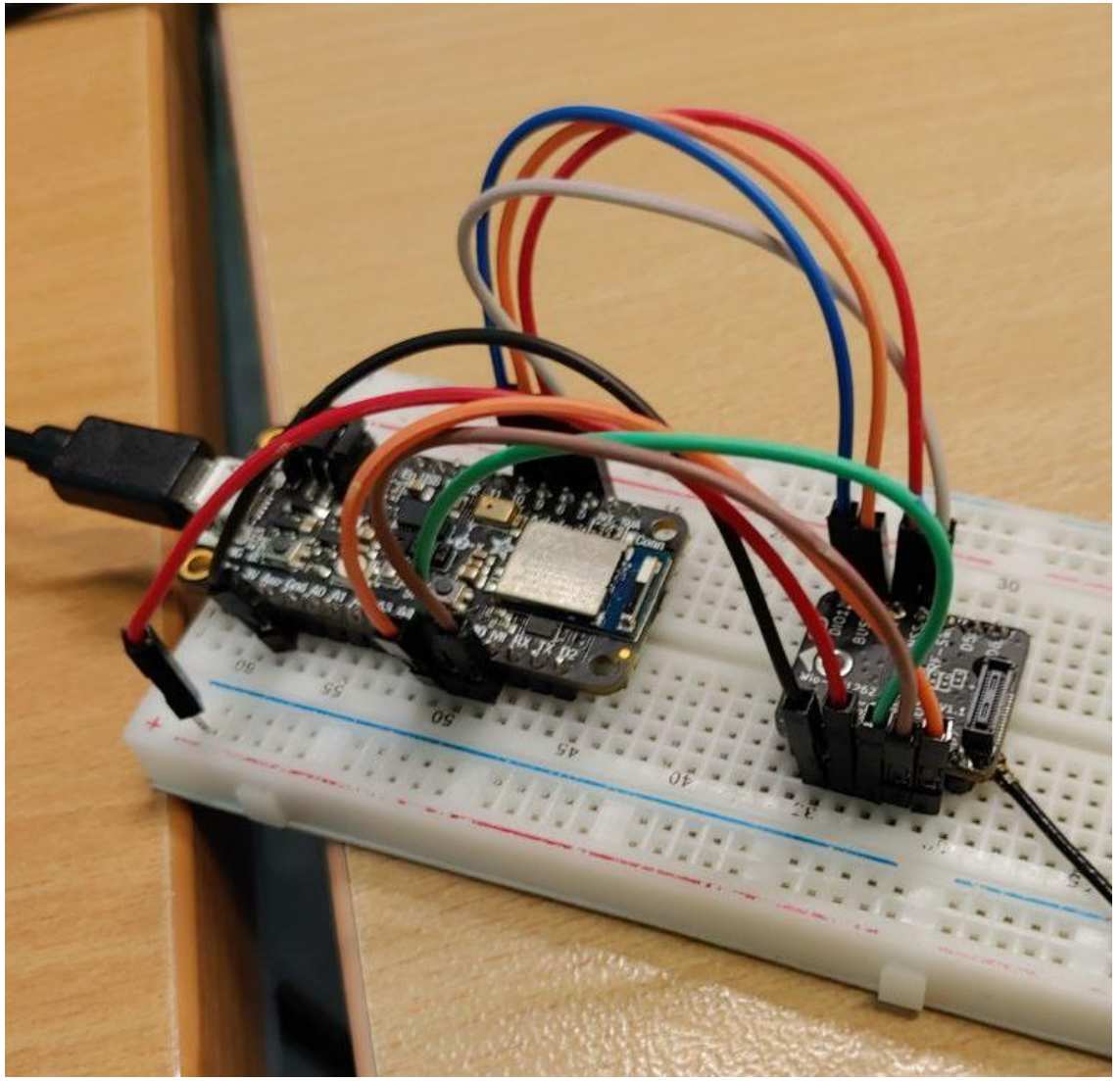
ESP32-C6 with SX1262 didn’t work

nRF52840 with SX1262

- worked with Arduino
- Never worked with RIOT

STM32WL5 worked out of the box 🥳

➔ Full Circle Innovation



Adafruit Feather nRF52840 Sense with SX1262 © Maya Hofmann

RIOT Drivers

Many standard components are supported

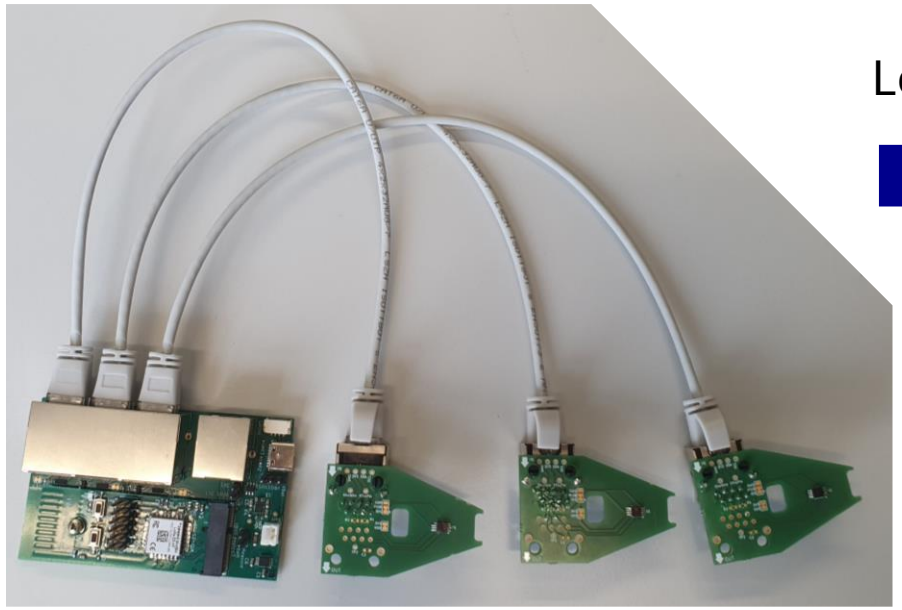
- Not those we had → switched components
- Especially low-power devices aren’t supported that well

Driver Quality fluctuates

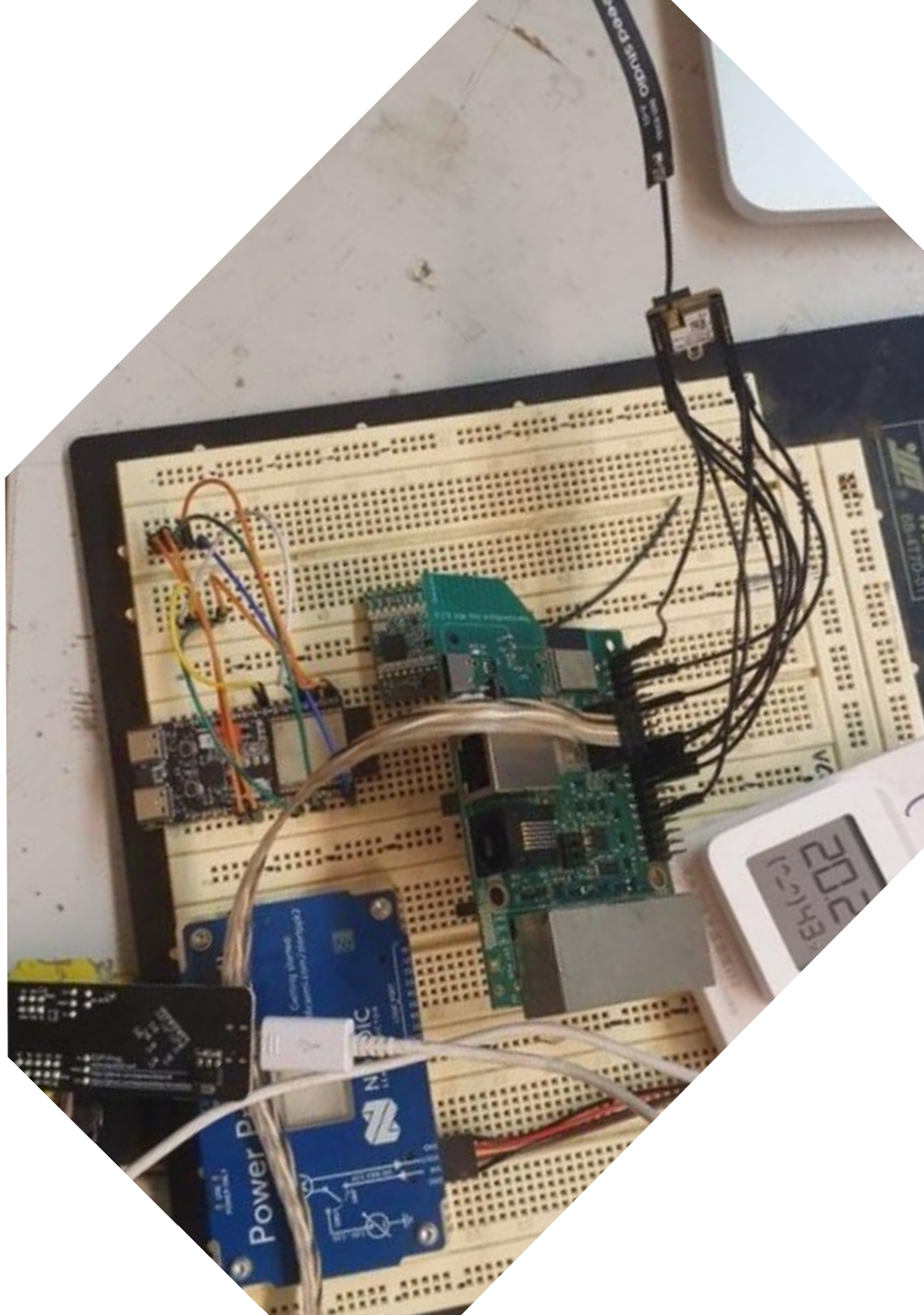
- At least documented when hacky

Our own board

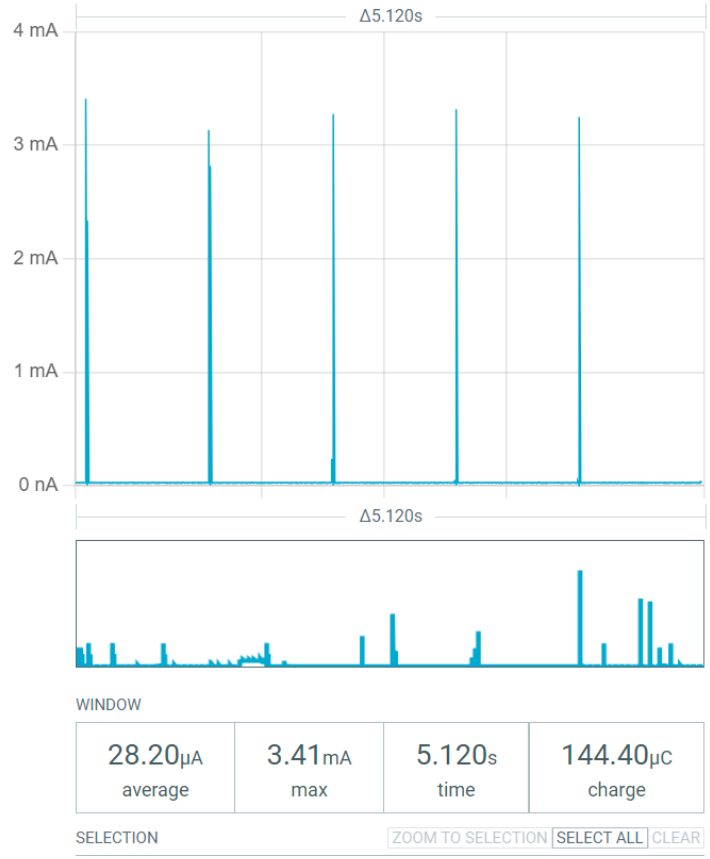
- Pin definitions not that clear
  - Especially for SPI



Lots of cross checking



Sleepmodes and Power consumption



Batterielebensdauer

$$
\begin{aligned}
I_{\text{sleep}} &= 26\,\mu\text{A} = 0.026\,\text{mA} \\
I_{\text{send}} &= 80\,\text{mA} \\
t_{\text{send}} &= 0.5\,\text{s} \\
T &= 300\,\text{s} \quad (= 5\ \text{Minuten}) \\
C &= 9\,\text{A h} = 9000\,\text{mA h}
\end{aligned}
$$

1. **Energieverbrauch pro Sendeintervall (300 s):**

$$
\begin{aligned}
Q_{\text{send}} &= I_{\text{send}} \cdot t_{\text{send}} = 80\,\text{mA} \cdot 0.5\,\text{s} = 40\,\text{mA s} \\
Q_{\text{sleep}} &= I_{\text{sleep}} \cdot (T - t_{\text{send}}) = 0.026\,\text{mA} \cdot 299.5\,\text{s} \approx 7.787\,\text{mA s} \\
Q_{\text{total}} &= Q_{\text{send}} + Q_{\text{sleep}} = 40\,\text{mA s} + 7.787\,\text{mA s} = 47.787\,\text{mA s}
\end{aligned}
$$

2. **Durchschnittlicher Stromverbrauch:**

$$
I_{\text{avg}} = \frac{Q_{\text{total}}}{T} = \frac{47.787\,\text{mA s}}{300\,\text{s}} \approx 0.1593\,\text{mA}
$$

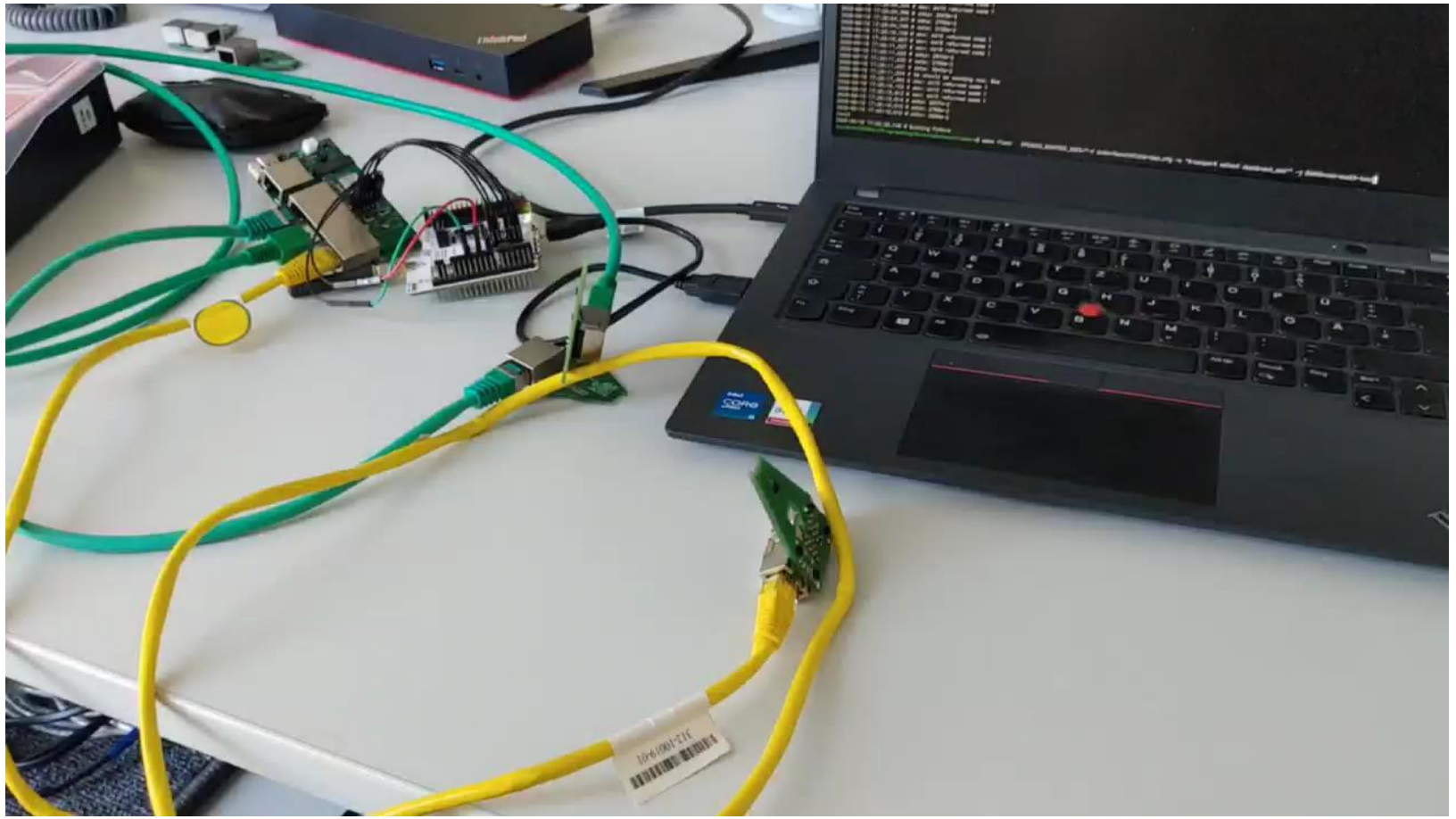
3. **Batterielebensdauer:**

$$
\begin{aligned}
t_{\text{life, h}} &= \frac{C}{I_{\text{avg}}} = \frac{9000\,\text{mA h}}{0.1593\,\text{mA}} \approx 56\,485.5\,\text{h} \\
t_{\text{life, d}} &= \frac{t_{\text{life, h}}}{24\,\text{h d}^{-1}} = \frac{56\,485.5\,\text{h}}{24\,\text{h d}^{-1}} \approx 2353.6\,\text{d}
\end{aligned}
$$

Ergebnis: erwartete Batterielebensdauer etwa 2353 d bzw. **6,4 Jahre**.

Debugging

- Relied purely on terminal
- Messages were very helpful



Summary

- Overwhelming for non-computer science people

- Driver testing and reworked/new drivers → less struggle
- New guide → very helpful for newcomers

- Very happy, that we tried RIOT

Special Thanks

- Mikolai Gütschkow
- Martine Lenders
- People at TUD Hack‘n‘Ack

- Prof. Matthias Wählisch
- Elena Frank