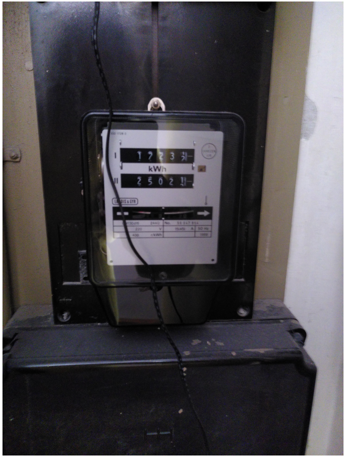
Electricity meter readout with RIOT using DSMR

Koen Zandberg (@bergzand) — Freie Universität Berlin

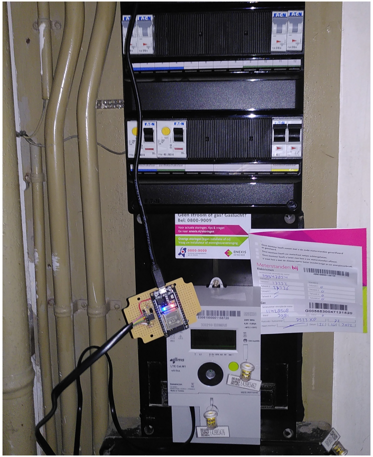
Our electric meter cabinet

Our electric meter cabinet

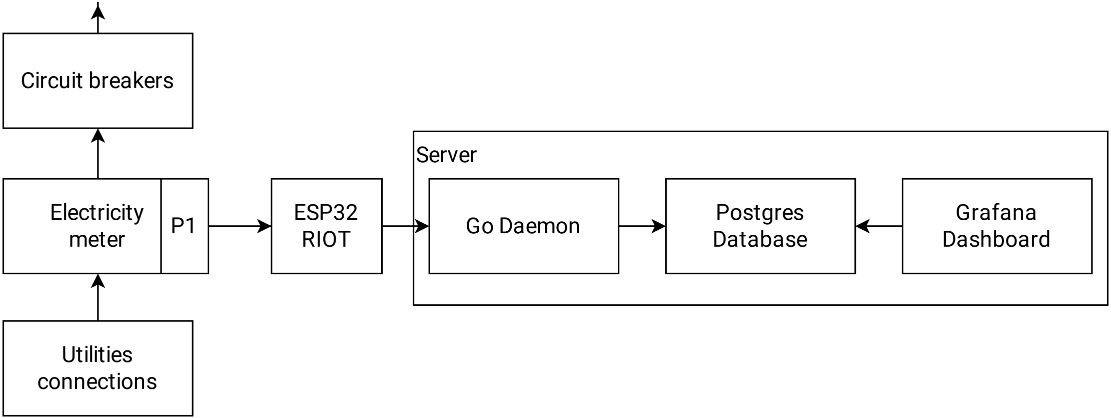


Our electric meter cabinet

Our electric meter cabinet



Overview


Circuit breakers
Electricity meter
P1
Utilities connections
ESP32 RIOT
Server
Go Daemon
Postgres Database
Grafana Dashboard

Why do I want this

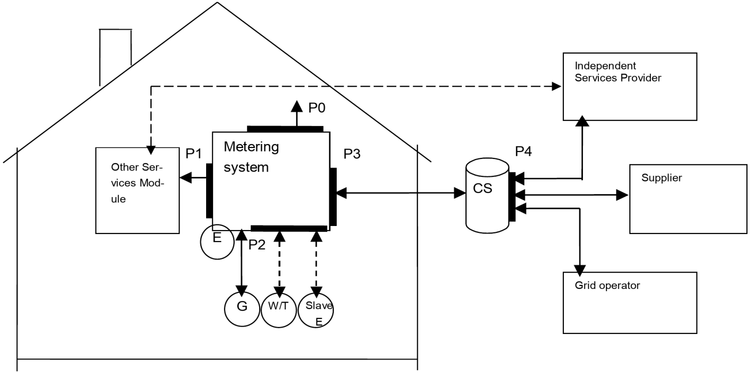
Mostly to get insights:

- Power consumption insights.
- Gas usage insights.

Because it sounds fun.

Electricity meter

So called *smart meter* with a Dutch **DSMR P1 port**



DSMR

Dutch Smart Meter Requirement

Open specification from Netbeheer Nederland

- Combined effort of Dutch grid operators
- Based on *IEC 62056*

P1 port for Automatic Meter Reading for consumers

Multiple specifications

All industry specifications with varying levels of openness:

- **DLMS**: Original specification for communication with smart meters
- **IEC 62056**: IEC spec for smart metering derived from DLMS
- **EN 62056**: European Union standard
- **DIN/NEN 62056**: Country-local specifications, between 100 - 400 euro per chapters
- **DSMR**: Dutch Smart Meter Requirements, derived specification (*open*)

IEC 62056

Standards for electricity meter data exchange

24 documents on specifications…

IEC 62056-6-1

OBIS: Object identification system

```
1-0:1.8.1(123456.789*kWh)
```

One OBIS maps to one property. Register values can include the unit

IEC 62056-21

Direct local exchange of data for meter reading (Serial over …)

Multiple protocol modes supported.

Focus on **Protocol mode D**

- Push button -> receive data

DSMR

Combines these specification into DSMR

Currently at version 5.0.2

- Added error checking
- Increased message baud rate
- Decreased interval (10 seconds -> 1 second)

DSMR (Physical)

RJ12 connector (6 pins, 6 connected):

- 5V power supply line
- Data request
- Data ground
- Data
- Power ground

Data request line acts as 'button'

5V provided and not included in the metering registers (free power)

DSMR: Example

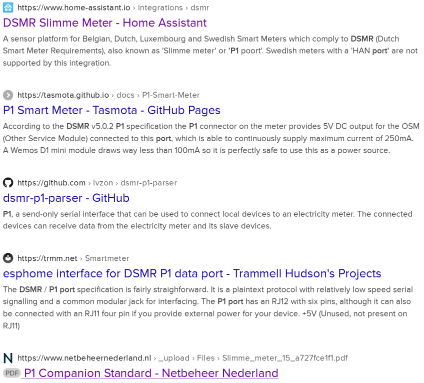
```
/Ene5\\XS210 ESMR 5.0

1-3:0.2.8(50)
0-0:1.0.0(230104215905W)
0-0:96.1.1(4530303631303030303031313638373230)
1-0:1.8.1(000233.793*kWh)
1-0:1.8.2(000300.631*kWh)
1-0:2.8.1(000000.043*kWh)
1-0:2.8.2(000000.000*kWh)
0-0:96.14.0(0002)
1-0:1.7.0(00.416*kW)
1-0:2.7.0(00.000*kW)
0-0:96.7.21(00003)
0-0:96.7.9(00001)
1-0:99.97.0(0)(0-0:96.7.19)
```

```
1-0:32.32.0(00001)
1-0:32.36.0(00000)
0-0:96.13.0()
1-0:32.7.0(237.0*V)
1-0:31.7.0(001*A)
1-0:21.7.0(00.416*kW)
1-0:22.7.0(00.000*kW)
0-1:24.1.0(003)
0-1:96.1.0(4730303538353330303437313231353230)
0-1:24.2.1(230104215500W)(00079.660*m3)
!3216
```

DSMR implementations

Multiple implementations available:
HomeAssistant, Arduino, Libraries, …



https://www.home-assistant.io › integrations › dsmr
DSMR Slimme Meter - Home Assistant

A sensor platform for Belgian, Dutch, Luxembourg and Swedish Smart Meters which comply to **DSMR** (Dutch Smart Meter Requirements), also known as 'Slimme meter' or '**P1** poort'. Swedish meters with a 'HAN **port**' are not supported by this integration.

https://tasmota.github.io › docs › P1-Smart-Meter
P1 Smart Meter - Tasmota - GitHub Pages

According to the **DSMR** v5.0.2 **P1** specification the **P1** connector on the meter provides 5V DC output for the OSM (Other Service Module) connected to this **port**, which is able to continuously supply maximum current of 250mA. A Wemos D1 mini module draws way less than 100mA so it is perfectly safe to use this as a power source.

https://github.com › lvzon › dsmr-p1-parser
dsmr-p1-parser - GitHub

P1, a send-only serial interface that can be used to connect local devices to an electricity meter. The connected devices can receive data from the electricity meter and its slave devices.

https://trmm.net › Smartmeter
esphome interface for DSMR P1 data port - Trammell Hudson's Projects

The **DSMR** / **P1 port** specification is fairly straighforward. It is a plaintext protocol with relatively low speed serial signalling and a common modular jack for interfacing. The **P1 port** has an RJ12 with six pins, although it can also be connected with an RJ11 four pin if you provide external power for your device. +5V (Unused, not present on RJ11)

https://www.netbeheernederland.nl › _upload › Files › Slimme_meter_15_a727fce1f1.pdf
PDF P1 Companion Standard - Netbeheer Nederland

RIOT integration

Goal: Convert the data from DSMR into a periodic CoAP POST request.

- DSMR: UART wrapper to IEC 62056 parsing
- IEC 62056: One module handling the OBIS register parsing (#19168)

Glue:

- Translate IEC 62056 into CBOR-encoded SenML metrics.
- Small GCoAP wrapper to periodically send the metrics

Experience

Parsing DSMR is rather tricky

- COSEM IDs are very flexible in their representation
- Registers *might* contain a unit
- Some registers have multiple data units associated

```
1-2:3.4.5.6
2:3.4.5
3.4
```

All together increases the number of edge cases.

RIOT integration

Result:

- DSMR: Handles and buffers a full telegram
- IEC 62056: Parses the telegrams with OBIS values and register values

Glue:

- Buffer all received values of interest
- Periodically send the latest metrics to the server of CoAP

List of metrics end up at the server!

Server side

Receive the measurements and store them into a database

Go application

- CoAP registry
- SenML parsing
    - Translate SenML fields into metric labels, including unit
- Database interface
    - Postgres with TimescaleDB for storing time series data
    - Alternatives possible

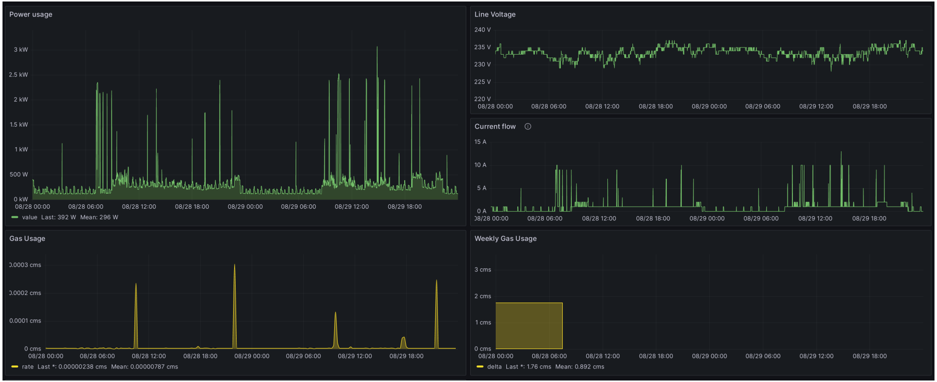
Database

PostgreSQL relational database

- **Realms**: A region of nodes
  - **Nodes**: A single device
    - **Series**: A single series of measurements
      - **Labels**: Labels associated
      - **Measurements**: The values

Display

Grafana with PostgreSQL plugin



Results

230 V is not 230 V: Swings between 226 V and 240 V

Power consumption consist of a background draw with scattered peaks:

- Background: Computers, refridgerators and lamps
- Peaks: Water boiler, Coffee machine, dishwasher.

Gas only used for hot water and cooking:

- Could be used to track shower and cooking durations.

Future work

DSMR data rate greatly exceeds database sampling rate:

- Extend metrics with extra info: High, low, average

Add 802.15.4 / 6LoWPAN capabilities:

- Fixed powered device as intermediate router

Conclusion

- Could be done within a weekend but …
- Overengineered applications
  - Most components generic enough for any sensor network