Monitor Station

Index

Summary

Raspberry Pi monitoring station

Arbete handlar om ett system som fungerar helt automatisk och är inte uppkopplad till elnätet.

Den skulle som enklast bestå av ett batteri, en energiproducent som tex ett solarpanel, en batteriladdare/reglerare och en energiförbrukare tex en lampa som tänds automatisk när batteriet uppnår en viss laddning och släcks när batteriet har tömts till en viss nivå.

Dem typen av system kan man hitta på båtar, parkeringsautomater och annat som inte är ansluten till elnätet.

Beroende på driften (användningstid per dag och elförbrukning) behöver man dimensionera batteriet och solarpanelen.

Istället för en lampa används en minidator (Raspberry Pi) som sammlar in data.

Då energiproduktionen på vintern inte räcker till att datorn drivs hela dagen, behövs ett modul (WittyPi) som kan starta/stänga av dator vid bestämda tidspunkter.

Denna modulen kan även mäta spänningen av batteriet, spänning av minidatorn och elekriska strömen av minidatorn.

På så sätt kan vi beräkna strömförbrukningen.

Nu vet vi inte hur mycket el som produceras av solarpanelen. I ett första försök skulle den mätas av en strömsensor, men mätningarna var för osäkra och därmed behöver vi hitta en annan metod.

Av batteriets spänning kan man uppskatta laddningsnivån. Sammanhanget mellan spänning och batteri kapacität är inte linijärt utan har en mer S-liknande kurva

Med en kontrollerad urladdning av våran batteri (12V bly batteri, 12Ah) kan vi kalkulera kurvan som beskriver sammanhanget mellan battariets spänning och laddningsnivå.

Förberedelse/Planering:
Komponenter som används i projektet

- Raspberry Pi 3/4
- Micro SD card
- Keyboard, mus.
- Blybatteri 12V 12Ah (Biltema)
- Solarpanel 20W
- Solarladdare/-reglare
- Step-Down DC-DC Converter 12V → 5V (ebay)
- Connectors
- Outdoor housing
- temperature and humidity sensor (DHT11, ebay)
- Rpi camera module

Preparations:
Parameters to measure:

- Battery voltage (BV, wittypi: input voltage)
- 5V Voltage (5Vpi) inside the Pi (wittyPi: output voltage)
- Electric current drawn by Pi. (wittyPi: output current)
- Temperature and humidity inside the box (temp: wittypi sensor, humidity: DHT11)
- Brightness is measured with the camera module

Preparations:
Parameters to log:

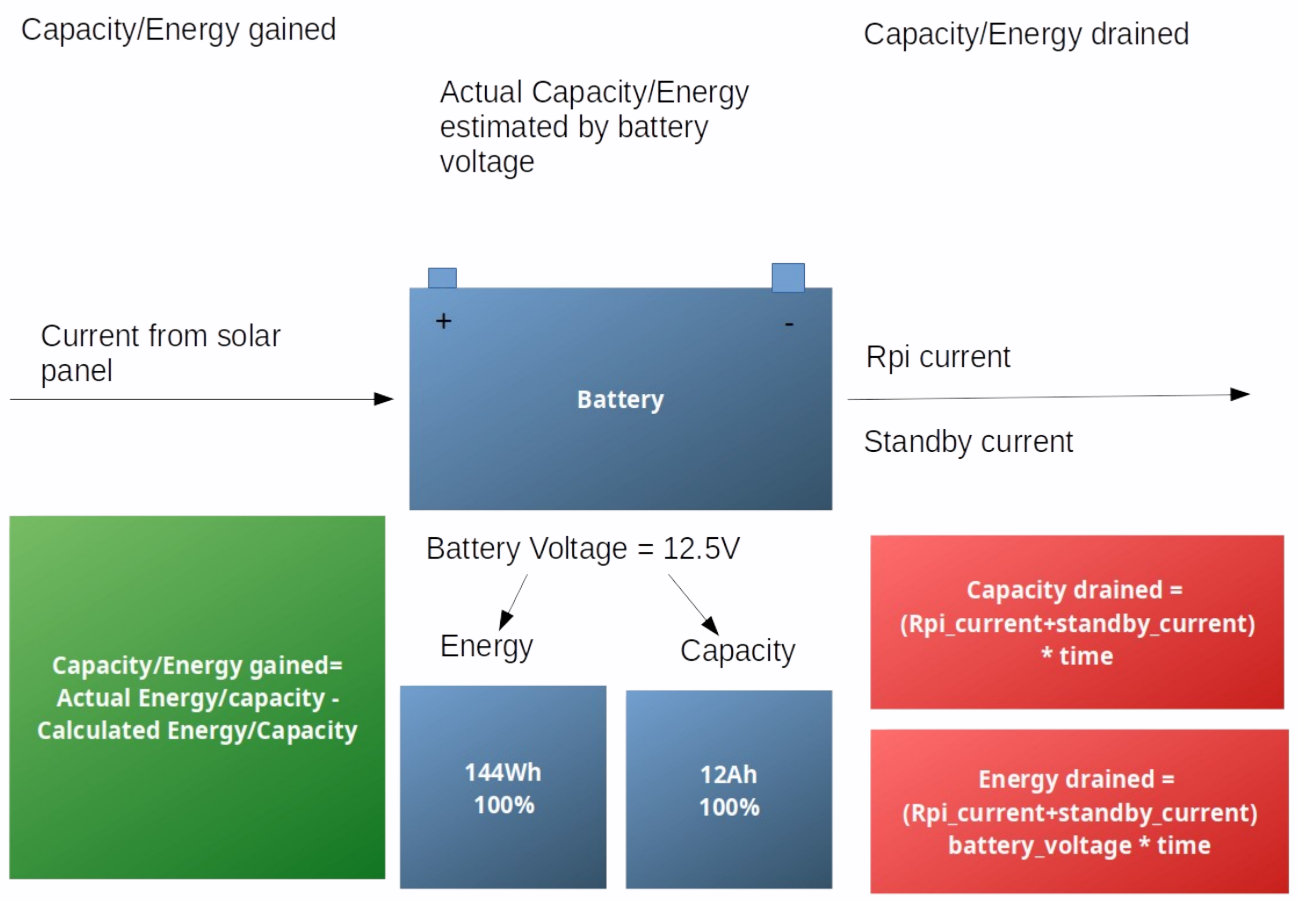
- CPU load
- CPU temperature
- and others

Can be used as surrogate values for power consumption

Preparations:
Values to Calculate

- Power consumption of Rpi: P = Rpi_voltage * RPi_current. (in W)
- Actual Battery capacity in Ah: Battery voltage → lookup battery capacity in table (numpy array)
- Actual Battery energy in Wh: Battery voltage → lookup battery energy in table (numpy array, see **Measurement of battery voltage/capacity curve)**
- Capacity drained in Ah: (Rpi_current + standby_current) * time
- Energy drained in Wh: (Rpi_current + standby_current) * Battery_voltage * time
- Capacity gained is estimated by: actual capacity – calculated capacity
- Energy gained is estimated by: actual energy – calculated energy
- Standby Power Consumption is estimated while Solar panel is covered for some days and energy/capacity loss is estimated by dropping battery voltage. Standby current is about 0.025 A (Solar charger, wittypi module, battery loss)

Preparations:
Energy Flow

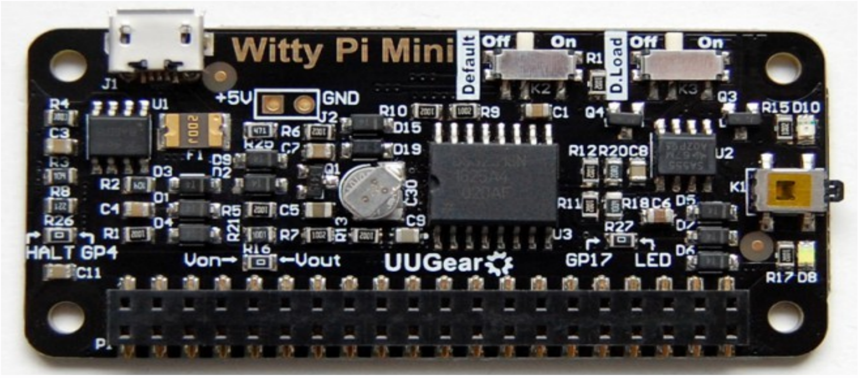


Materials & Methods
1. Preparation of Raspberry Pi

- Just follow the instructions to prepare a Rpi like in on this website: https://projects.raspberrypi.org/en/projects/raspberry-pi-getting-started
- In the first version of the monitor station a ADC and hall sensors are used to measure voltage and current, but the main problem were:

  - Power gain by solar panel is insufficient to support operation 24/7.

  - Current measurement with hall sensors are unreliable. Hall sensors measure the magnetic field of a electric current, but because the electric current is weak and there are many disturbing magnetic fields, no reliable measurement is possible with this design.
- That’s why MonitorStation version 2 makes use of a Witty Pi Mini 3 (WittyPi).



Materials & Methods
1. Preparation of Raspberry Pi

- WittyPi add a RTC (Real Time Clock) and power management and can therefor:

  - power off at low battery levels and scheduled times

  - power on at defined battery voltage or scheduled time

  - measures:

  1. battery voltage (input voltage)

  2. voltage of Rpi (5V, output voltage)

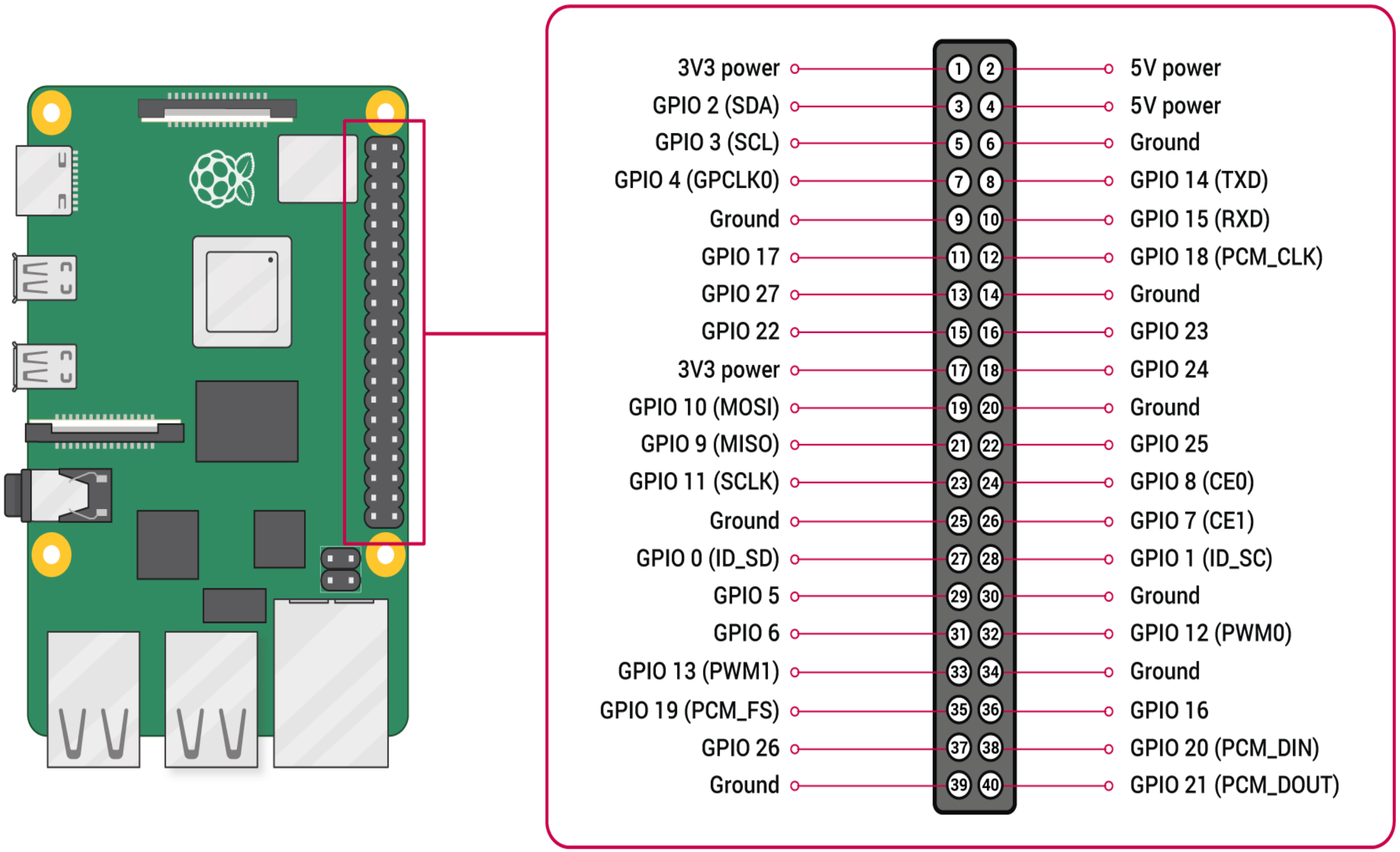
  3. Rpi current (output current)

  4. temperature

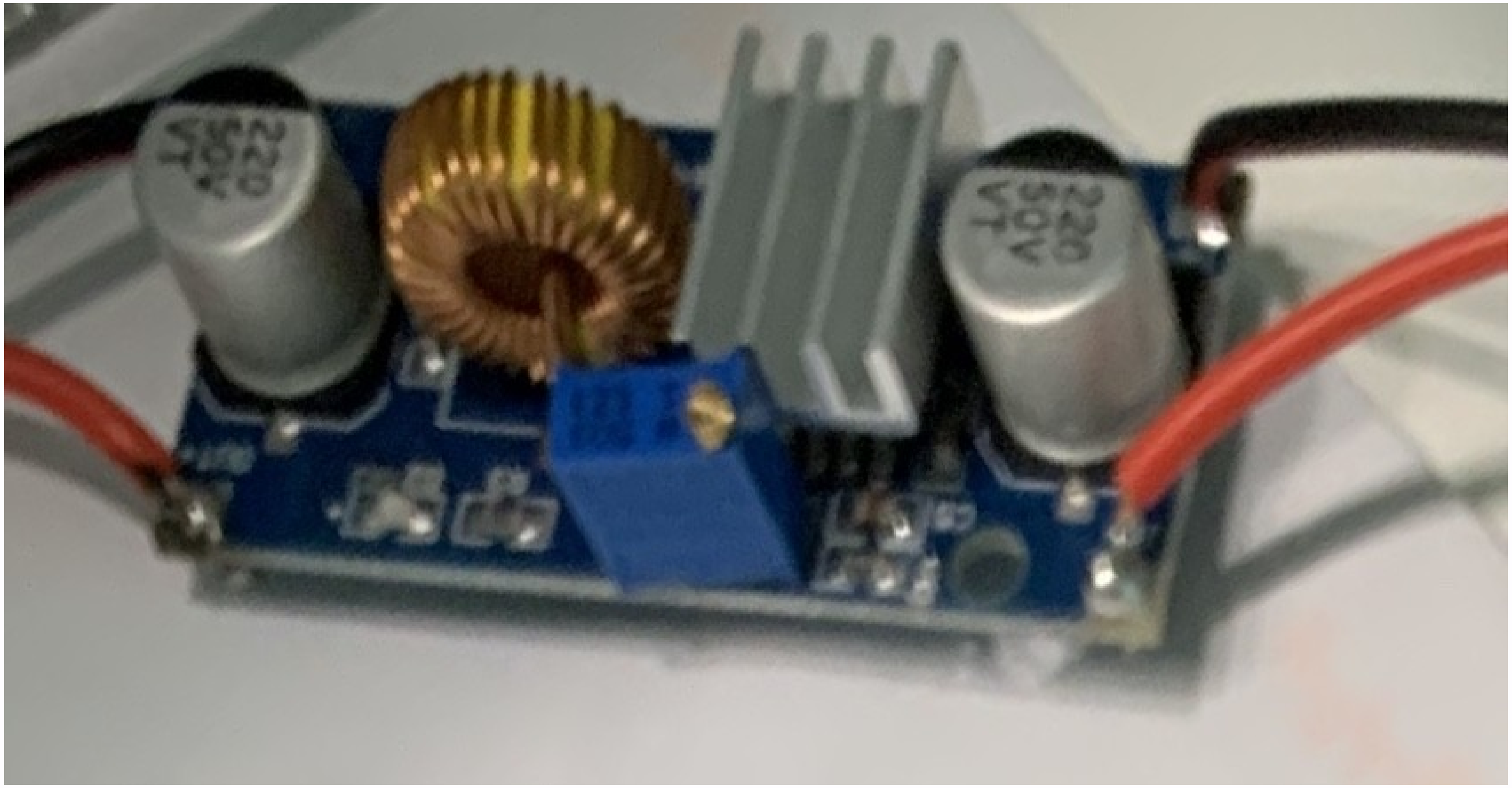
- Rpi power consumption in Watt can be calculated with P = U*I (RPi_Voltage * RPi_current)

Materials & Methods

GPIO General-Purpose Input/Output

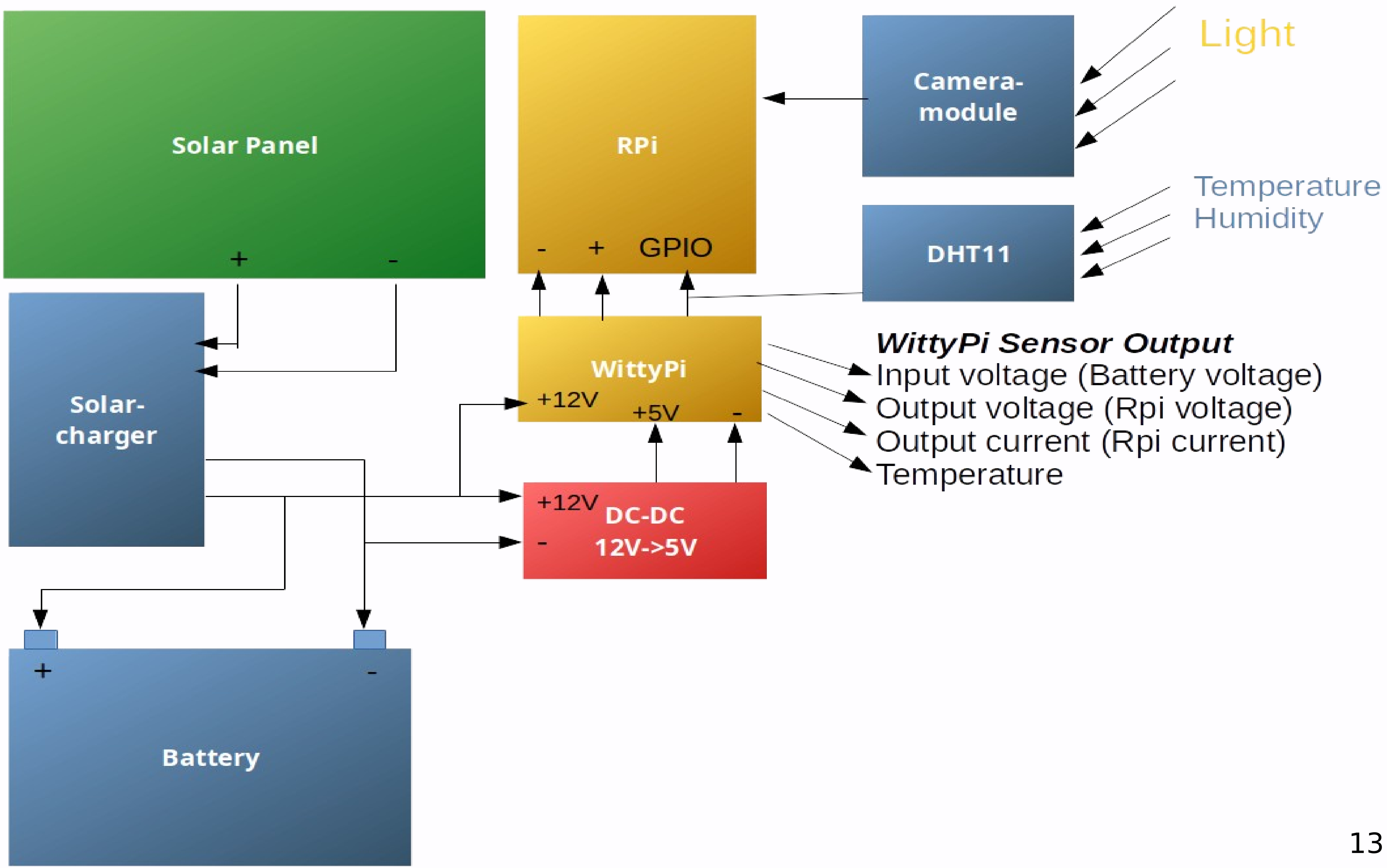


Materials & Methods
Step Down DC-DC Converter (12V->5V)

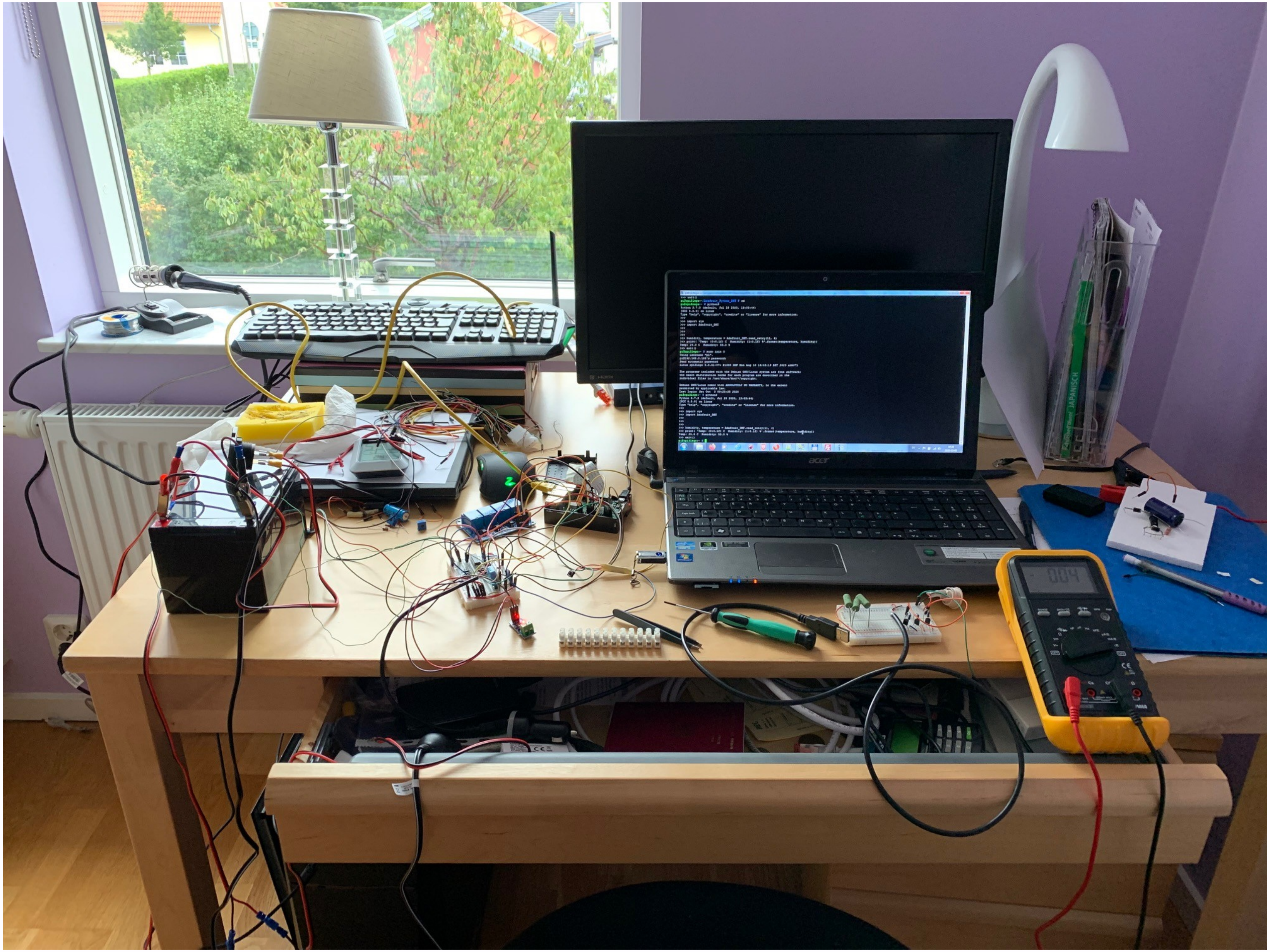


Materials & Methods

Circuit Diagram



Materials & Methods
My Work Station



Materials & Methods (Version 1)

Python Installation of MCP3xxx Library

Install program library. Type in terminal:

```
sudo pip3 install adafruit-circuitpython-mcp3xxx
```

start python with: `python3`

import libraries with:

```
import busio
import digitalio
import board
import adafruit_mcp3xxx.mcp3008 as MCP
from adafruit_mcp3xxx.analog_in import AnalogIn
```

configure serial connection

```
spi = busio.SPI(clock=board.SCK, MISO=board.MISO, MOSI=board.MOSI)
cs = digitalio.DigitalInOut(board.D5)
mcp = MCP.MCP3008(spi, cs)
```

Assign channel to pin 0

```
channel = AnalogIn(mcp, MCP.P0)
```

```
print('Raw ADC Value: ', channel.value)
print('ADC Voltage: ' + str(channel.voltage) + 'V')
```

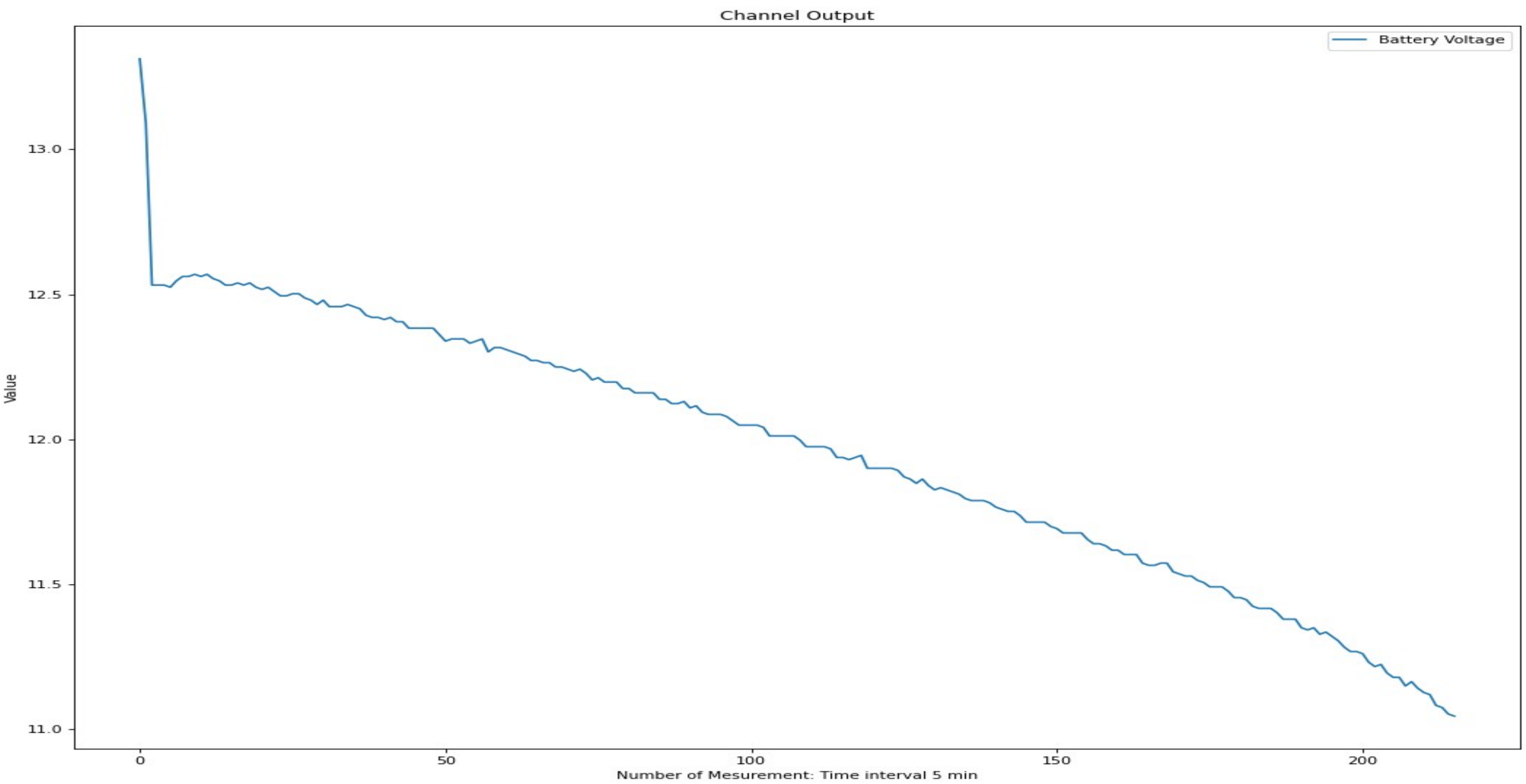
Materials & Methods

2. Measurement of battery voltage/capacity curve

To estimate actual capacity and energy of the battery instead of measuring current from and to the battery we discharge the battery over a long time limiting the current by resistors and measure dropping battery voltage

.

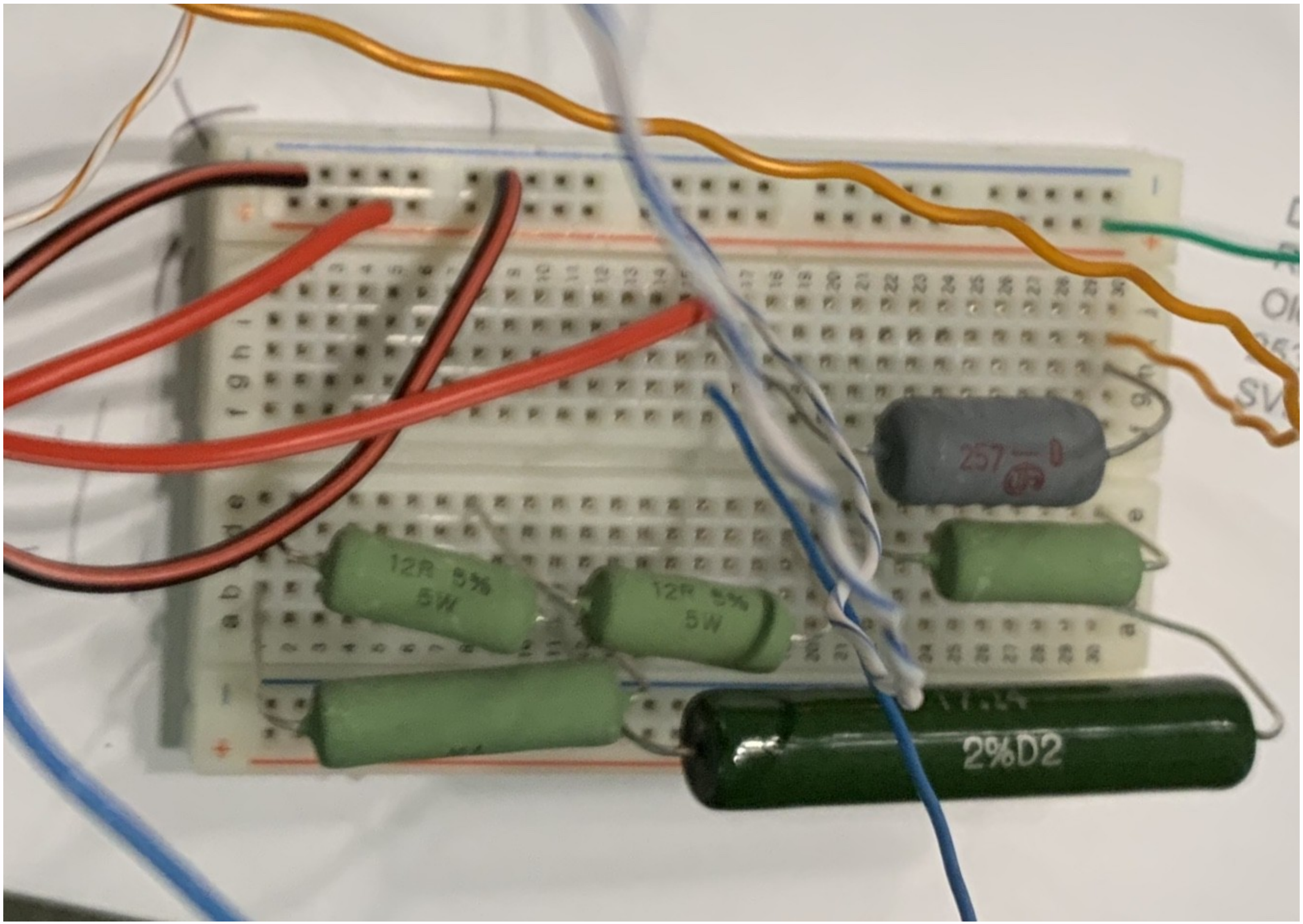
In this way we can estimate battery capacity and energy by simply measure battery voltage.



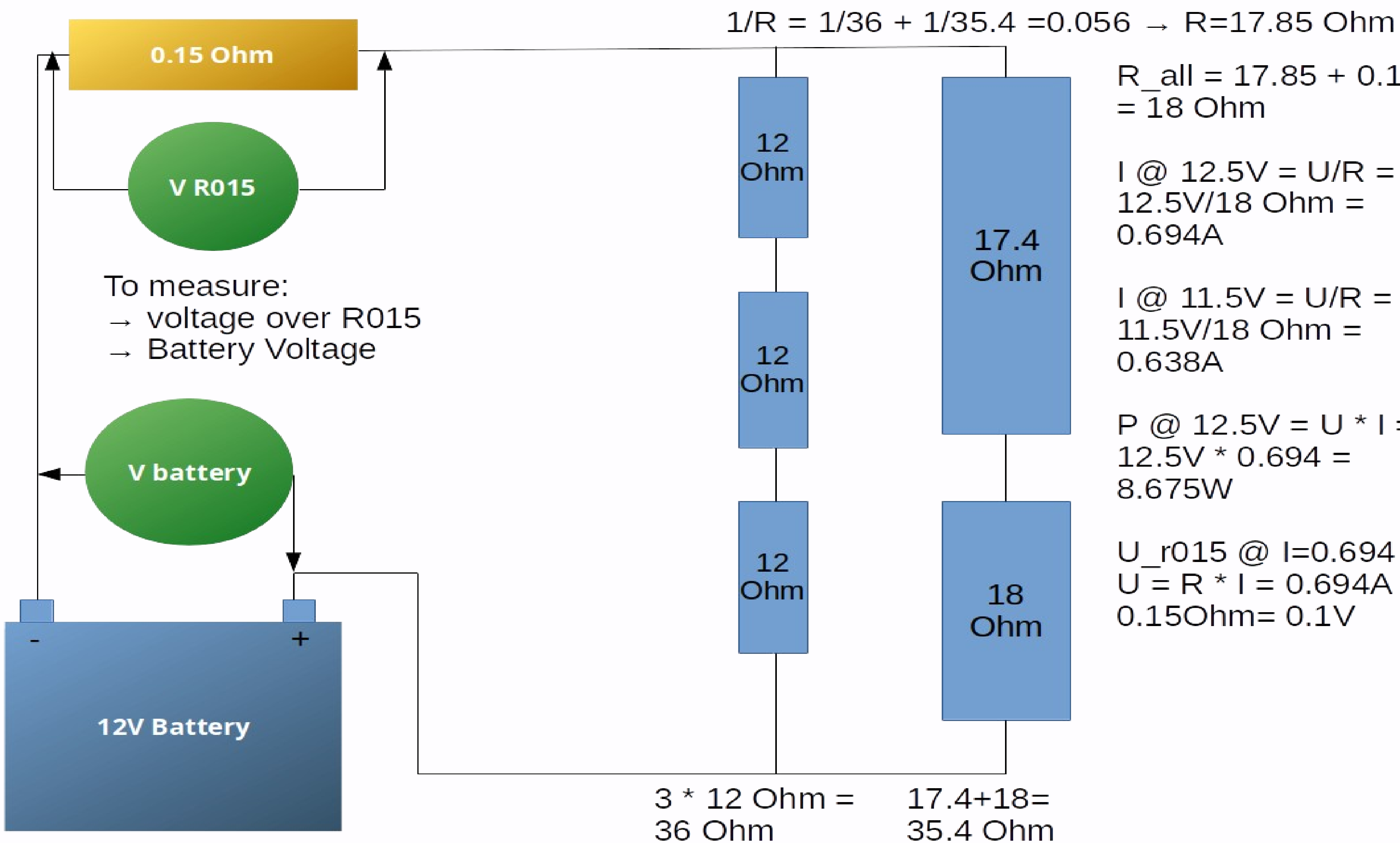
Materials & Methods

2. Measurement of battery voltage over time while discharging the battery

Resistor Array used to discharge the battery to relate battery voltage to battery energy and capacity.



Materials & Methods
2. Diagram of the Resistor Array



R_all = 17.85 + 0.15 = 18 Ohm

I @ 12.5V = U/R = 12.5V/18 Ohm = 0.694A

I @ 11.5V = U/R = 11.5V/18 Ohm = 0.638A

P @ 12.5V = U * I = 12.5V * 0.694 = 8.675W

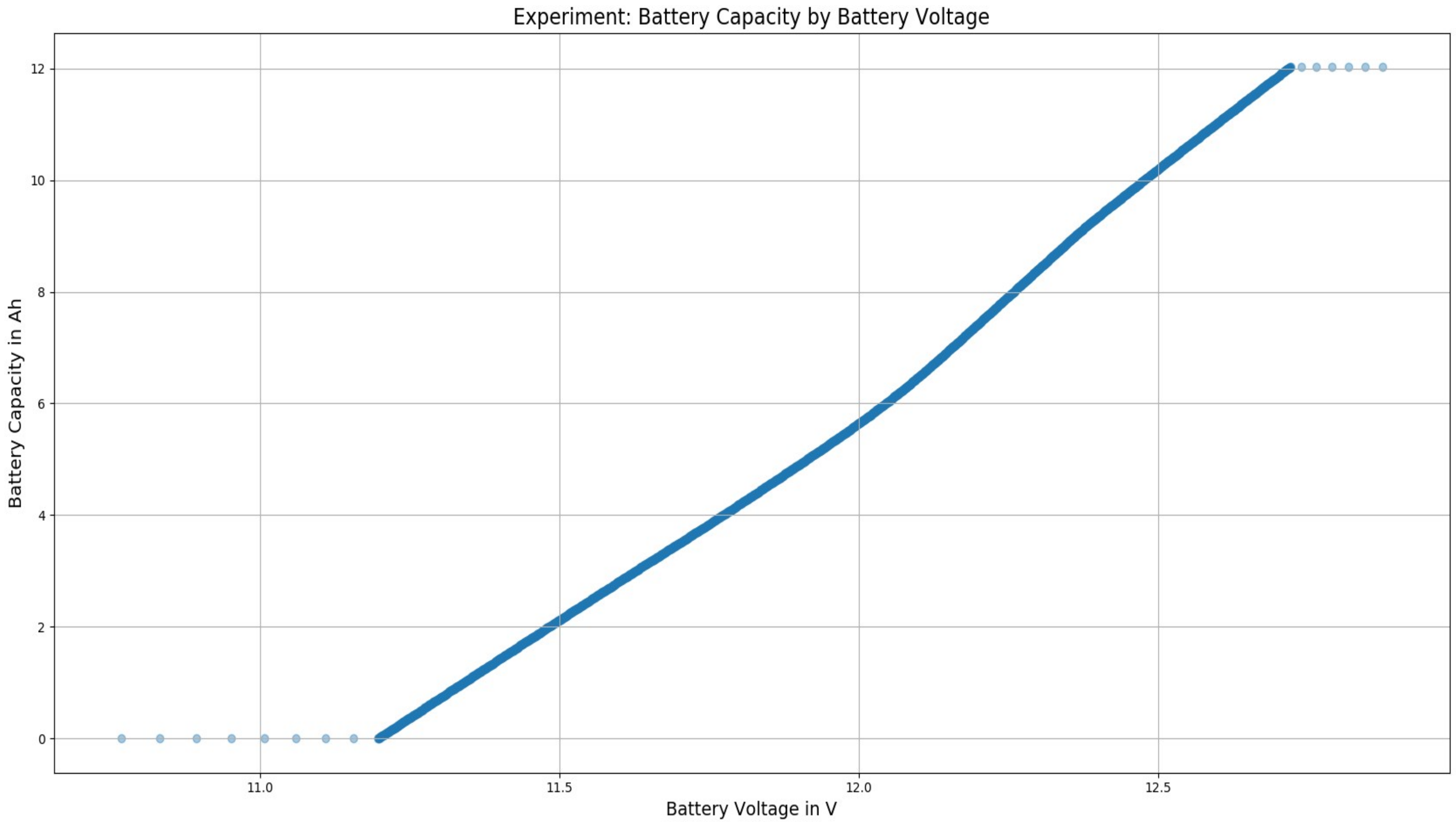
U_r015 @ I=0.694
U = R * I = 0.694A * 0.15Ohm= 0.1V

Materials & Methods
2. Calculate Battery Capacity and Energy

- Discharge the battery over an array (serial and parallel connected) resistors to divide current so that resistors don’t get too hot.
- Voltage over a 0.15Ohm resistor is measured.
- Discharge current is calculated by measuring the voltage over a 0.15 Ohm resistor according to: I = U/R = U/0.15Ohm.
- Discharge current can also be calculated by: Actual Battery Voltage / Resistance of the array (18Ohm)
- Battery voltage is measured.
- Battery capacity (dischared) is calculated by: I * time (in hours)
- Energy is calculated by: E = U * I * time (Volt, Ampere, hours) in Wh
- The results show that battery capacity is nearly 12 Ah as promised by the manufactor and energy is according 144 Wh (12V * 12Ah).
- Results are saved in a numpy array and used to estimate remaining battery energy and capacity by battery voltage.
- While charging the battery, voltage is higher and estimated energy/capacity shows false higher values.

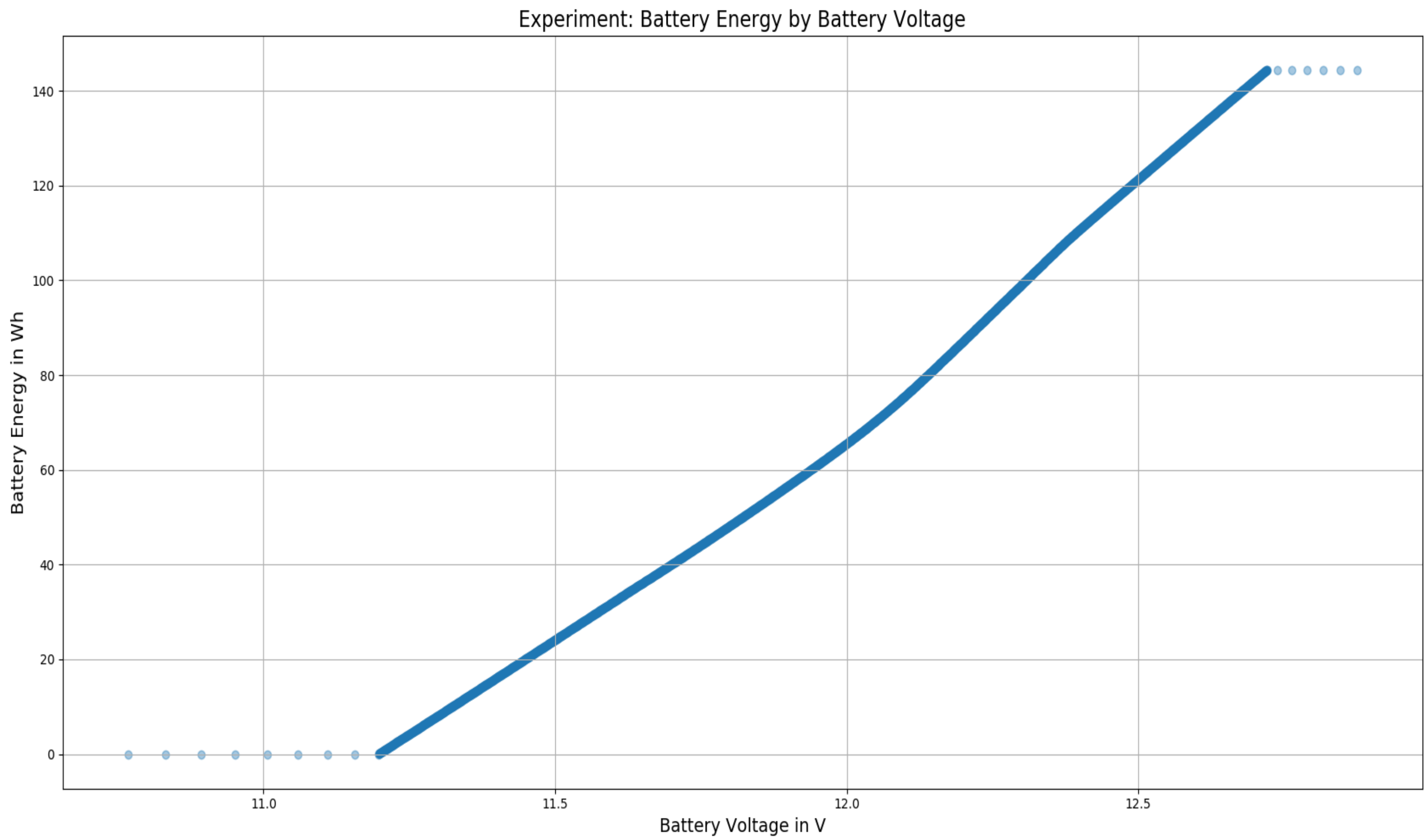
Materials & Methods

2. Battery Capacity over Battery Voltage Plot for this lead 12Ah battery (Biltema)



Materials & Methods

2. Battery Energy over Battery Voltage Plot for the same battery



Materials & Methods
3. Software

- Now we’ve got a table relating battery voltage to capacity and energy. We only need to define a function to lookup values in the table. This is done by the voltage2energy:
- This function calculates the difference of actual battery voltage to all values in the table and find the index of the best match (least difference)
- This index is used to lookup what energy and capacity the battery should have at this voltage (absolute values and procent)

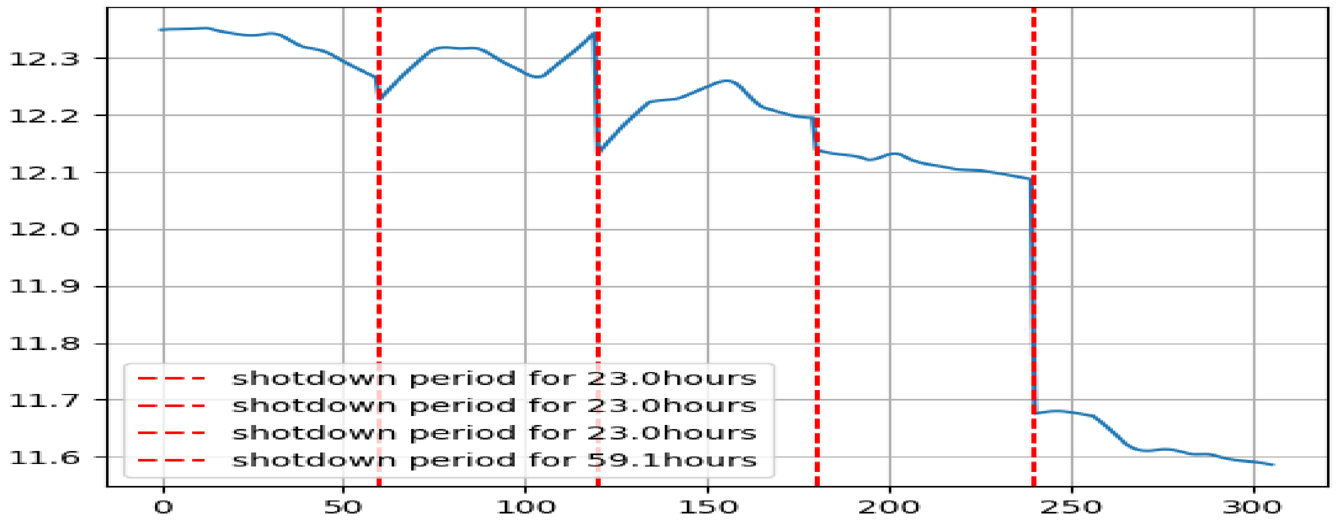
```
def voltage2energy(act_volt = 11.9):
    """map voltage to energy left in battery, when battery is known"""
    diffvolt = np.absolute(battVolt - act_volt)
    mindiff = diffvolt.min()
    act_idx = diffvolt == mindiff
    act_energy = energy[act_idx][0]
    act_capacity = capacity[act_idx][0]
    act_energy100 = energy100[act_idx][0]
    act_capacity100 = capacity100[act_idx][0]
    return act_energy,act_capacity,act_energy100,act_capacity100
```

Materials & Methods
3. Software

- The main software is located in monitor3.py
- It may look complicated, but everything is straight forward:
- Read config file.
- Set up log file.
- Load battery files (voltage, capacity, energy) as numpy arrays.
- Start a monitor thread (runs in background) to monitor every minute (or other interval) all values that should be measured or logged.
- Save values in a csv-file (comma seperated file) for further analysis.
- *AnalyzeData* and *AnalyzeLastDays* are functions to calculate and estimate power consumption and production.
- The code is published at Github: https://github.com/marl2en/monitorstation/blob/main/README.md

Results

- Here is a plot of battery voltage over 5 periods. The x-axis are number of measurements (every minute over one hour = 60 values per period)
- Y-axis is actual battery voltage.
- The red dotted lines are shutdown periods of 23 or 59 hours.
- Voltage readings are fluctuating.
- Voltage drops a lot under standby for 59 hours: No solar power production while still consumtion. Standby current = 0.025A. Capacity drained by 0.025A * 59h=1.475Ah.
- From period 1 to 2 battery voltage is relative stabil: Power consumption is nearly equal to solar power production.

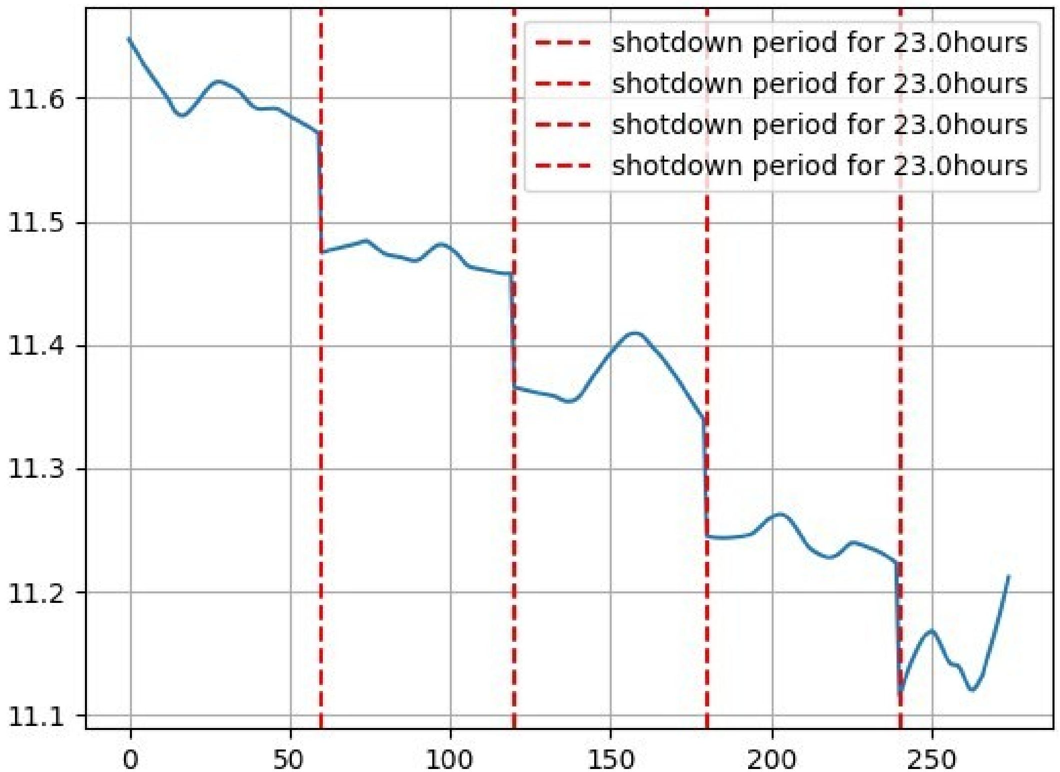


Discussion

- All the time a current of 0.025A is drawn from the battery.
- For one day this means the capacity is going down by: 0.025A * 24h = 0.6Ah (5% of battery capacity).
- When Rpi is running it draws additionally 0.25 – 0.5A (@12V) from the battery.
- Rpi is mostly idle. CPU load about 1-2%. Power consumption about 3-6W.
- To keep the battery from dying, the solar panel should produce at least 0.6Ah or 12V * 0.6Ah = 7.2Wh per day.
- Because of dark and short days at winter and position of the panel behind a window, power production per day is enough for operating Rpi one hour per day (about 4Wh) and compansate for standby power loss → 7.2Wh + 4Wh = 11.2Wh
- Maximum power production of the panel is 20W. If we get 10W for 8 hours is would be 80Wh. (sunny sommer day)

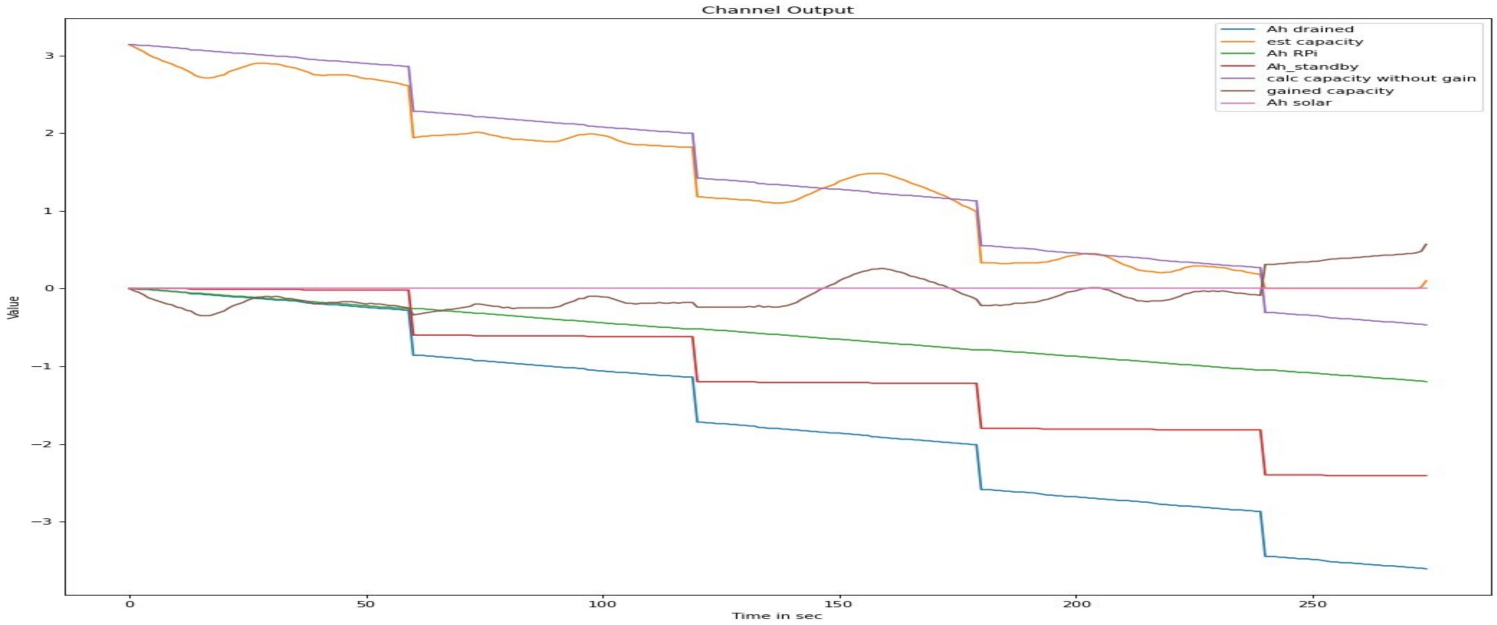
Discussion

Battery voltage drops from day to day. Not really any solar power production.



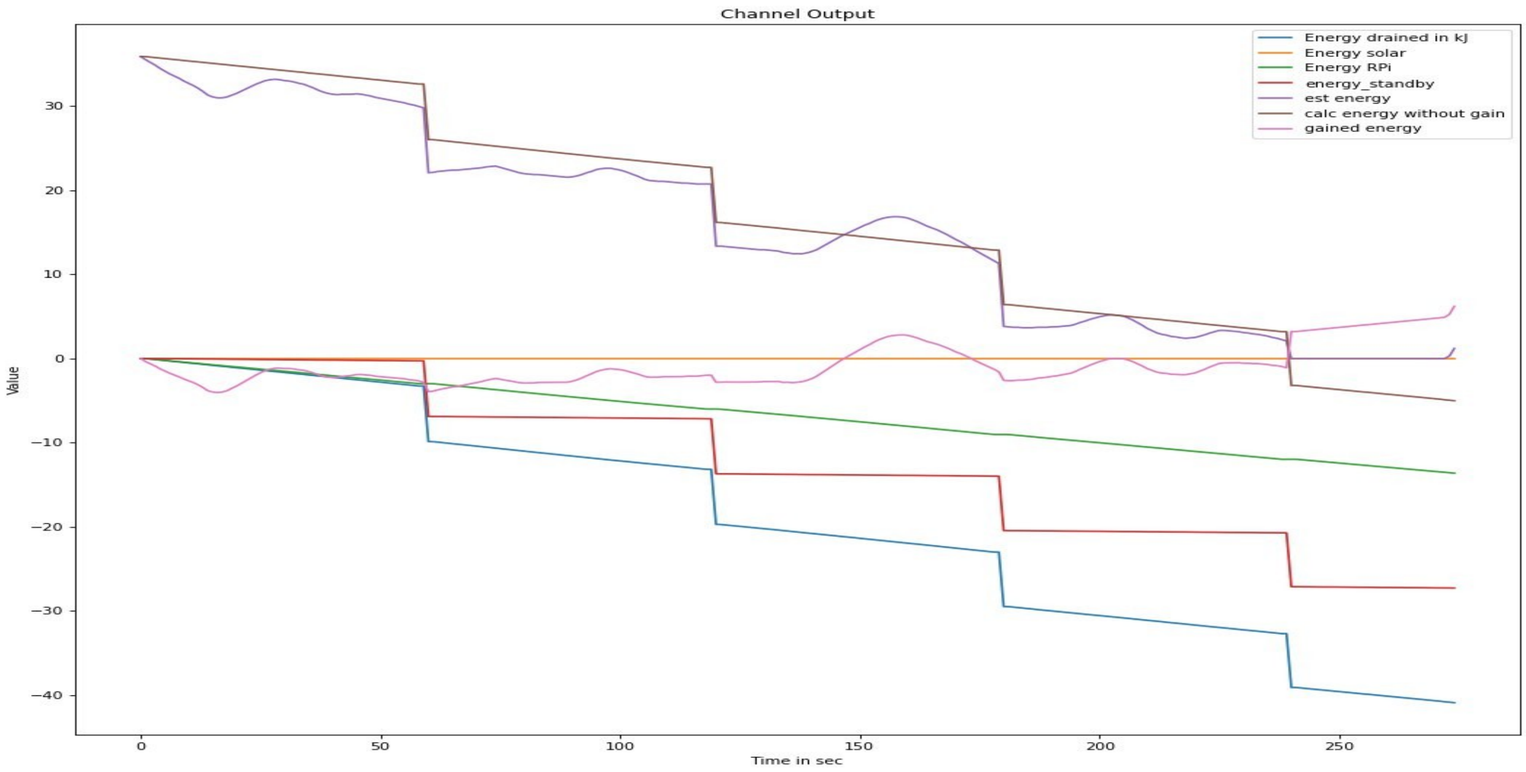
Discussion - Capacity

- Capacity loss by:
- Standby current (Ah standby)
- Rpi current when activated. (Ah Rpi)
- Calculated capacity = Capacity at beginning – loss (standby and Rpi)
- Estimated capacity by battery voltage
- Gained capacity = estimated capacity – calculated capacity



Discussion - Energy

- The same period for energy in Wh (not kJ)
- X-axis is number of measured/calculated values (not seconds)
- No solar power production.
- Brightness can be used to estimated power production, but Rpi can only be active one hour per day and that’s why not record brightness.



Discussion – Last Words

- Right from beginning I ran into two main problems:

  1. Winter in Sweden is dark and not a good time for a solar powered project.

  Operation 24/7 isn’t possible with this design at winter. That’s why wittyPi is needed to shutdown/startup Rpi at scheduled times. Wittypi is also good for power management and can record data like input voltage, output voltage and so on.

- 2. Hall current sensors are good to measure current without electrical interconnecting your circuits. Every current generates a magnetic field, which can be measured by the hall effect (voltage difference).

  The main problem is that currents in this project are far too small and electrical noise to high to produce any meaningful sensor readings.

  Solar Power Production can be estimated by the difference of expected battery voltage and real voltage. If Power is produced, than battery power gets higher.

  Brightness can be another option to estimate solar power.