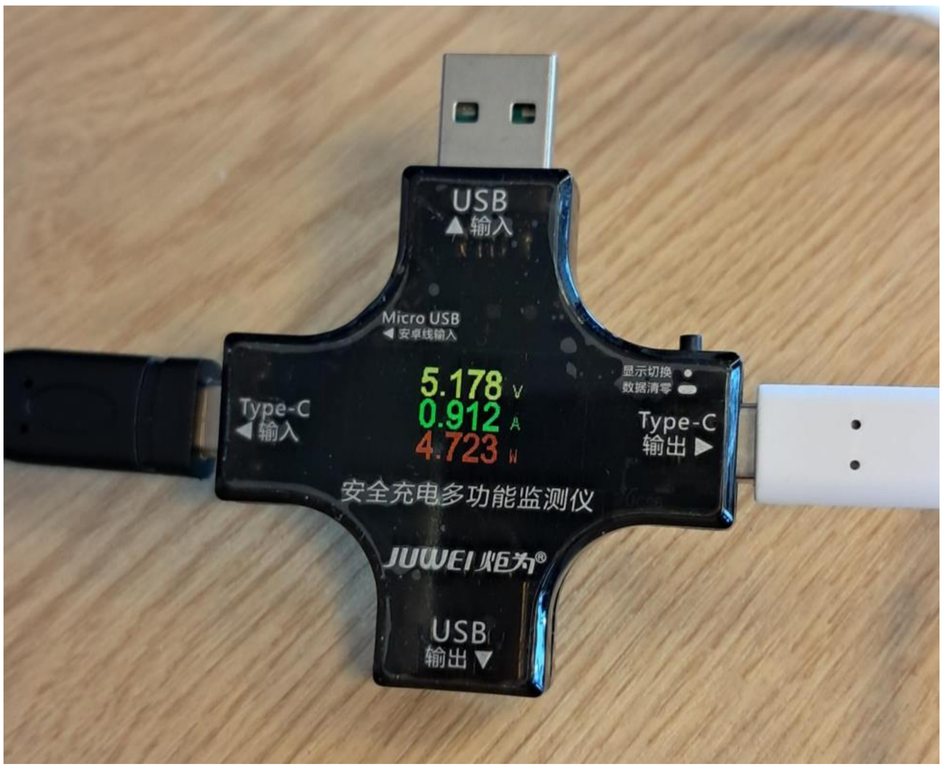
Exercise 3

Initial measurements

Upon switching on the Raspberry Pi, the power draw steadily rises as the different circuits start to turn on. The power fluctuates significantly but settles to the idle state of around 4.7 W, with occasional upwards fluctuations to 4.9 W. We expect this, since when the Pi is idle, the external loading is constant.

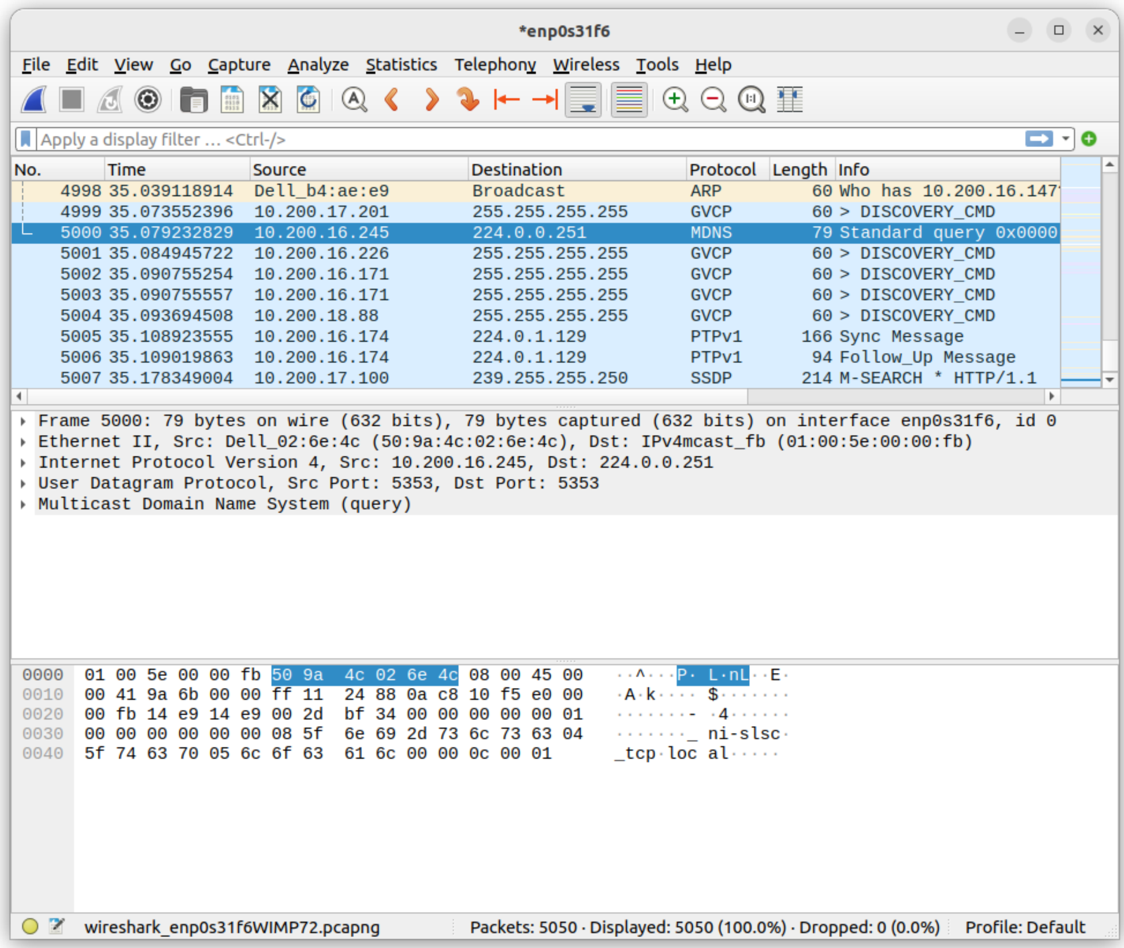


A picture of the voltage, current and power measurements on the meter

Network activity

Here we look at the effect of network activity on energy usage. Based on the earlier lecture on sustainable networks, we are expecting it to have at most a minor effect since a lot of wired network devices have relatively high idle power draw.

We are now instructed to emulate network activity on the ethernet cable. To do this, we need to decide the rate at which to send packets. Using WireShark, let's look at the local wifi network:



5000 packets were received in 35.08 seconds (seen highlighted in the screenshot), which maps to an average rate of 142.5 packets/second!

Now we can use *ping* to flood the cable with traffic:

averaged over 100000 packets	Voltage / V	Current / A	Power / W
Idle	5.17	0.92	4.8
Pi pinging LM	5.16	1.00	5.2
LM pinging Pi	5.17	0.97	5.0

We can see that the voltage remains the same, and the current is the quantity that changes. This makes sense – the voltage input should be regulated to prevent degradation of dielectrics in capacitors, and to preserve logic levels.

It takes more power for the Raspberry Pi to send data than to receive data. This also makes sense, since this involves generating a changing electric field. A receiver does not need to expend energy doing this.

Switching to using *iperf* with the Raspberry Pi as the client,

averaged over 30 seconds	Voltage / V	Current / A	Power / W
Idle	5.17	0.92	4.8
rx-usecs = 57	5.155	1.11	5.7
rx-usecs = 0	5.154	1.12	5.8

The parameter *rx-usecs* is the time (in milliseconds) that the devices wait before generating an interrupt, which is the alert that the device gets when data is received. Setting this to zero means it must check more often, so the processor is doing more checks and hence drawing more power (but only very slightly).

CPU activity

We will now use *stress-ng* to load the Raspberry Pi's CPU.

averaged over 30 seconds	Voltage / V	Current / A	Power / W
Idle	5.17	0.92	4.8
2 cores	5.14	1.26	6.5
4 cores	5.138	1.30	6.7

The stressed power draw is much greater than the idle draw (by over 33% in both cases).

Theoretical Calculations

Let's use electricity maps to find the carbon footprint of running the Raspberry Pi in various power grids.

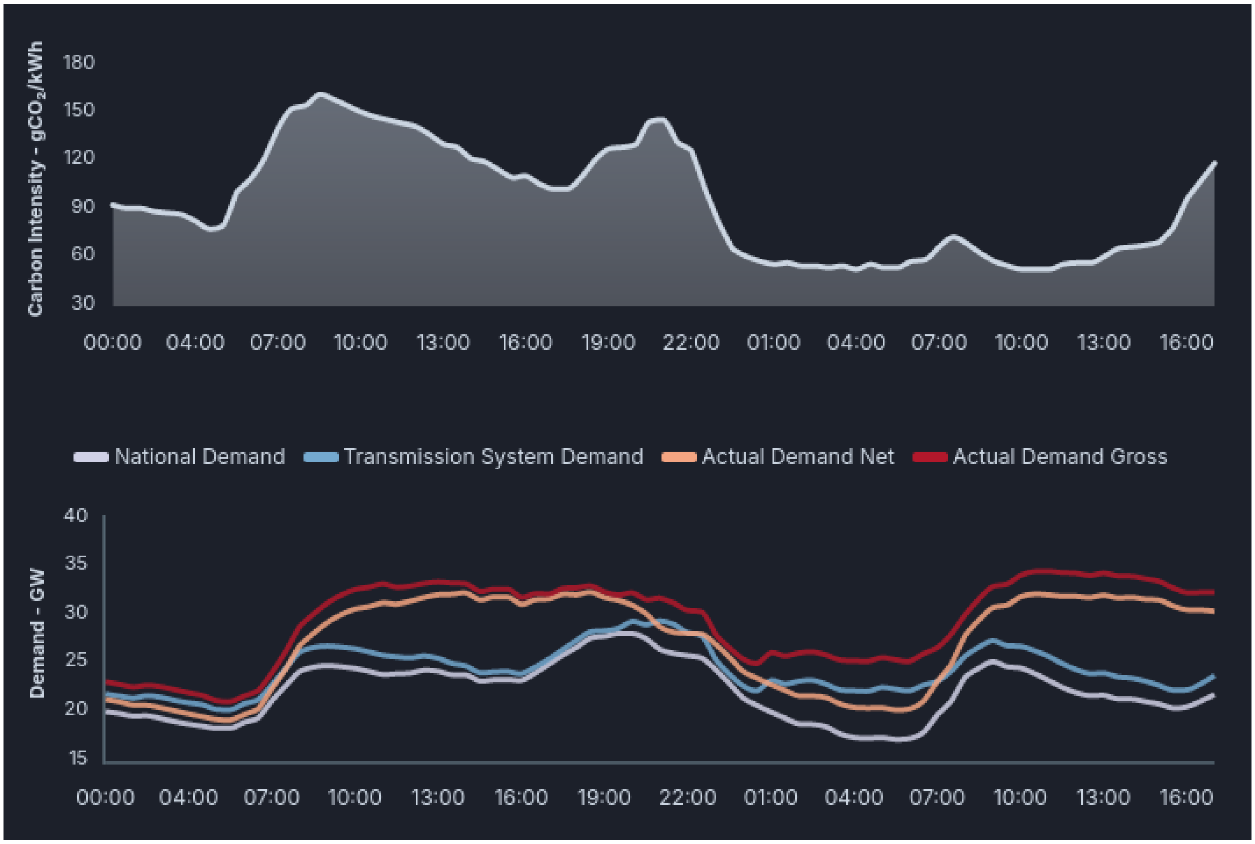
For this exercise, we're going to assume a worst-case scenario from the stress tests, a mean power of 6.7 W. The following table describes the carbon intensity and hence theoretical footprint in various locations:

Power Grid	Carbon intensity / gCO_2eq/kWh	Carbon footprint of RasPi in 1 hour / gCO_2eq	Carbon footprint of RasPi in 1 day / gCO_2eq
California ISO, USA	145	0.97	23.3
Estonia	222	1.49	35.7
Great Britain, UK	93	0.62	15.0
Hong Kong	592	3.97	95.2
Iceland (100% renewable!)	28	0.19	4.5
NSW, Australia	654	4.38	105.2

The carbon footprint is calculated as 6.7×10^{-3}kW × 1 hour × CI, and multiplied by 24 for the whole-day value. Values were taken from https://app.electricitymaps.com/map/72h/hourly on 10th June, 2025.

Suppose there are 30 billion devices connected to the internet. Then if their power draw was modelled as the equivalent of 30 billion Raspberry Pis, what would be their carbon footprint? Using a global mean carbon intensity of 481.5 gCO_2/kWh in 2023 [1] the total carbon footprint would be 96.8 $tonneCO_2eq/h$, or 848 million tonnes CO_2 equivalent in a year. This seems like a very big number, but the total energy related emissions that year were 37.4 billion tonnes [2], over 40 times higher than that value! So really, the contribution to energy-related emissions would be hardly 2.3% in our model.

However, this is almost certainly a bad estimate. The global mean carbon intensity becomes much less helpful when we consider the distribution of devices in space and time. We don't know exactly where everything is and therefore can't assume that devices are evenly distributed across different regions; for example, there are probably many more devices in China than there are in New Zealand, due to the population difference. Furthermore, renewables are often not synchronised with energy demand, so a mean intensity will not account for the extra effect of high-demand, high-intensity periods.



Graph of carbon intensity and demand over time – notice the high demand, high CI period in the working hours of the day (approximately between 8 AM and 8 PM) [3]

Finally, it is completely silly to model all devices as having the same power draw as a small Raspberry Pi. Devices have power draws ranging from hardly a watt to several kilowatts (or even more), so a Raspberry Pi being on the low end is not representative.

References

All references were retrieved 10 June 2025:

[1] https://www.statista.com/statistics/943137/global-emissions-intensity-power-sector-by-country/

[2] https://www.iea.org/reports/co2-emissions-in-2023/executive-summary

[3] https://www.energydashboard.co.uk/live