# Raspi-Driver-SPS30

Software to read out [Sensirion SPS30](https://www.sensirion.com/en/environmental-sensors/particulate-matter-sensors-pm25/) PM sensor values over I2C on Raspberry Pi.

This software is licenced under GPLv3 by [UnravelTEC OG](https://unraveltec.com) (https://unraveltec.com), 2018.

## Prerequisites

Maximum I2C speed for SPS30 is 100kHz (Default on Raspbian).

Recommended maximum I2C cable length 20cm.

Sensor needs 5V, on I2C bus 3.3V is fine. Seems to work without external pull-ups on Raspberry Pi 3B+ on i2c-1.

### Pigpiod & Python

As the SPS30 needs complex i2c-commands, the Linux standard i2c-dev doesn't work. A working alternative is pigpiod (we are using it through python).

```

aptitude install pigpio python3-crcmod python3-pigpio i2c-tools

```

Atm, IPv6 doesn't work on Raspbian correctly with pigpiod, so:

```

sed -i "s|^ExecStart=.\*|ExecStart=/usr/bin/pigpiod -l -n 127.0.0.1|" /lib/systemd/system/pigpiod.service

systemctl restart pigpiod

# Test (should return an int)

pigs hwver

```

## Run program

```

python sps30-service.py

```

## Installation

To install it as a background service run ./install.sh (install dependencies, e.g. via apt, first).

This service writes a file - suited for scraping by prometheus - (onto ramdisk on /run/sensors/sps30/last) and updates it every second.

# Notes

## Stability

When running on a RPi 0W powered by a USB3.0 PC port, (4.8V resulting), the sensor stops reliably working after 60s, and has to be resettet. Use a proper PSU.

## Data credibility

data values in the first 15 seconds while the fan is spinning up are too high, recommended to ignore them.

## Calculation of pm values

The pm\* values (µg/m³) are summed up - pm10 is every particle <10µm, pm4 is every particle <4µm, etc. If e.g. there are no particles > 2.5µm, pm2.5 - pm10 are the same.