Temperature and humidity monitoring @HHES

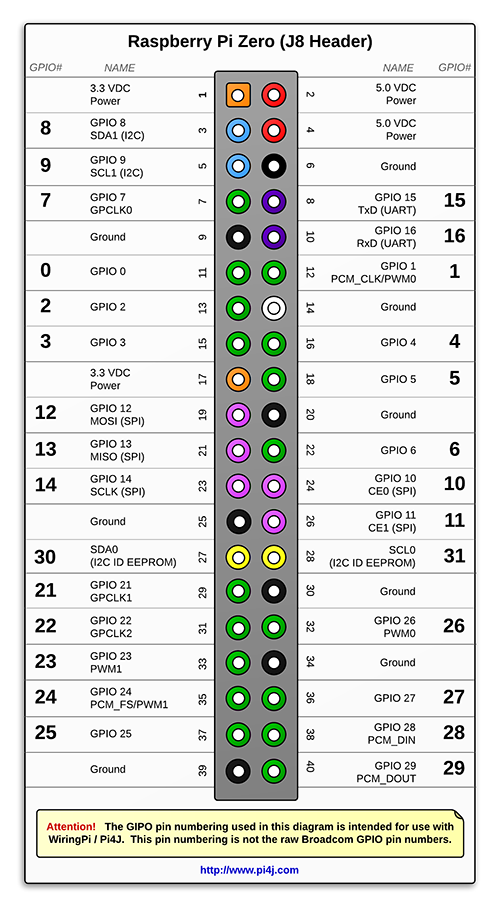
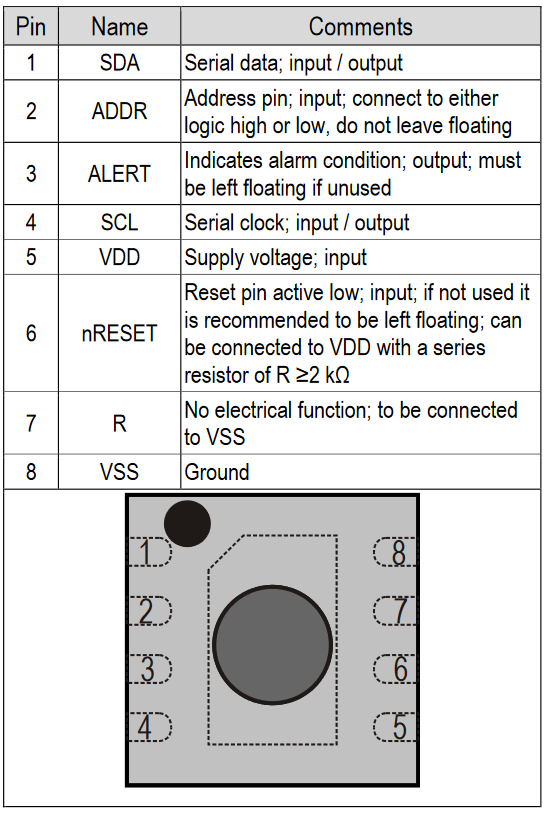
**Sensor:** [**SHT31**](https://sensirion.com/products/catalog/SHT31-DIS-B)

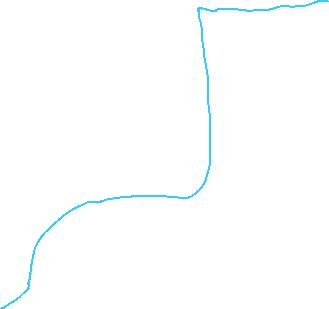
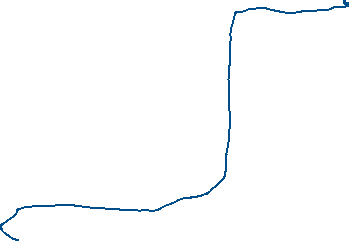
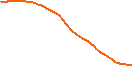
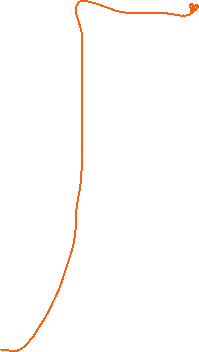
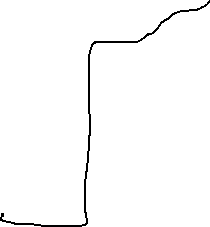
**Client:** [**Raspberry Pi Zero**](https://www.raspberrypi.com/products/raspberry-pi-zero/)

**Server:** [**Raspberry Pi 3 Model B**](https://www.raspberrypi.com/products/raspberry-pi-3-model-b/)

**Tutorial:** [**Reddit**](https://www.reddit.com/r/raspberry_pi/comments/x24aem/raspberry_pi_and_reading_sht3x_sensors/)

Wiring of the sensor





|  |  |
| --- | --- |
| **SHT31 PIN** | **RPI Zero PIN** |
| **1 (SDA)** | **3 (SDA1)** |
| **2(ADDR)** | **1 (3V3)** |
| **3 (ALERT)** | **NC** |
| **4 (SCL)** | **5 (SCL1)** |
| **5 (VDD)** | **1 (3V3)** |
| **6 (nRESET)** | **NC** |
| **7 (R)** | **NC** |
| **8 (VSS)** | **6 (GND)** |

Functional overview

**Communication with the sensor:**

**Protocol:** I2C

**Sensor address:** 0x45

**File:** [**sht31.py**](sht31.py)

**Function:** get\_measurement()

**Communication with the server:**

**Protocol:** TCP

**Server socket:** 192.168.1.43:51378

The server serves all the clients i.e. Rpi Zeros that are responsible for sending in measurement data.

If the client cannot get to the server, it sleeps for a random period before retrying. The randomness is implemented because the sensors take periodic measurements, so it could be the case that the sending would synchronize and a sensor would be forever blocked from sending.