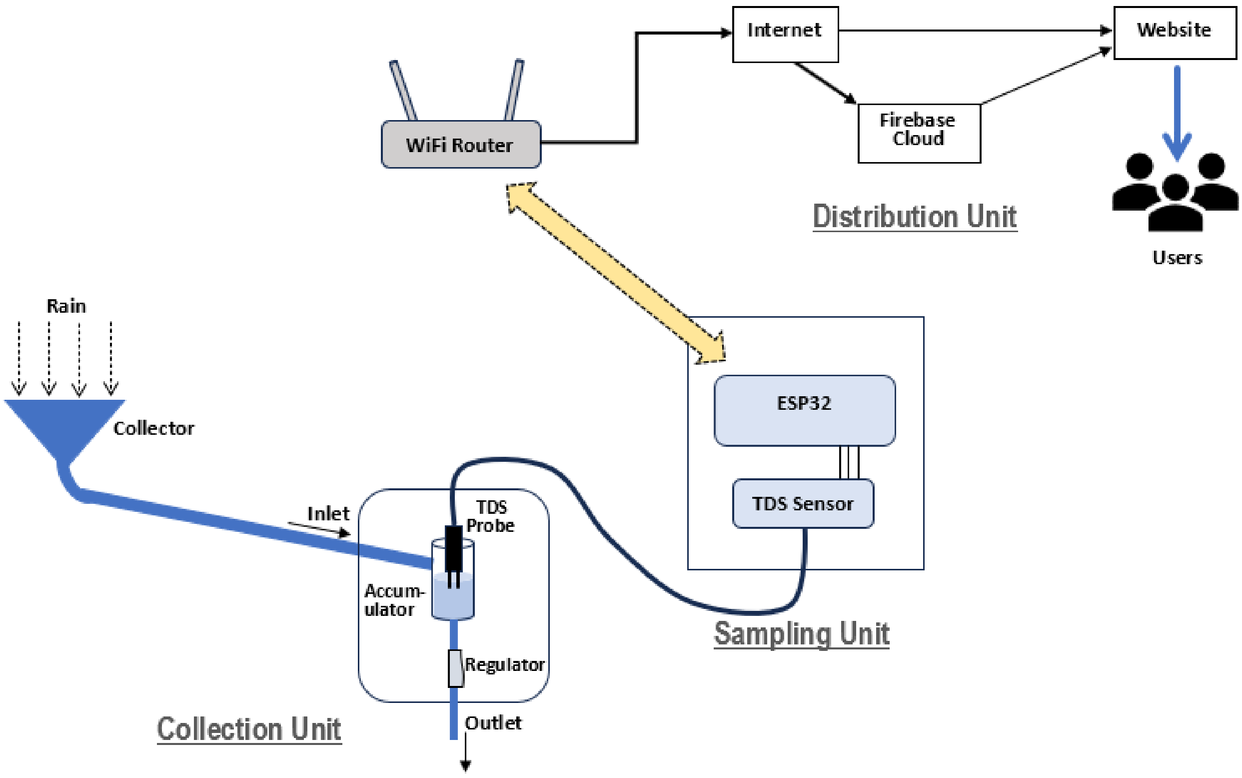
Technical details for RGDC Monitor: Rainwater TDS Unit

Project Summary: This project aims to develop and install a detection device at Ramanuj Gupta Degree College, Silchar, for rainwater TDS value and archive the data in a web location so that this data can be accessed by anyone for research purpose. The device becomes active whenever a detectable amount of rainwater is received, then it samples the data and uploads it to a network location with a time-stamp via WiFi connection. Thus, the TDS data is uploaded in real time which can be downloaded later by using our website https://rgdcmonitor.blogspot.com. This data is useful to track and monitor local environmental changes and pollution level for a short or long period of time. This project has been designed for educational and research purpose and therefore all the codes and software are released as open-source assets with the GNU GPL-3.0 license via GitHub (https://github.com/DwaipayanDeb/RGDC_Monitor).

Blueprint: The blueprint of the whole project is schematically given below

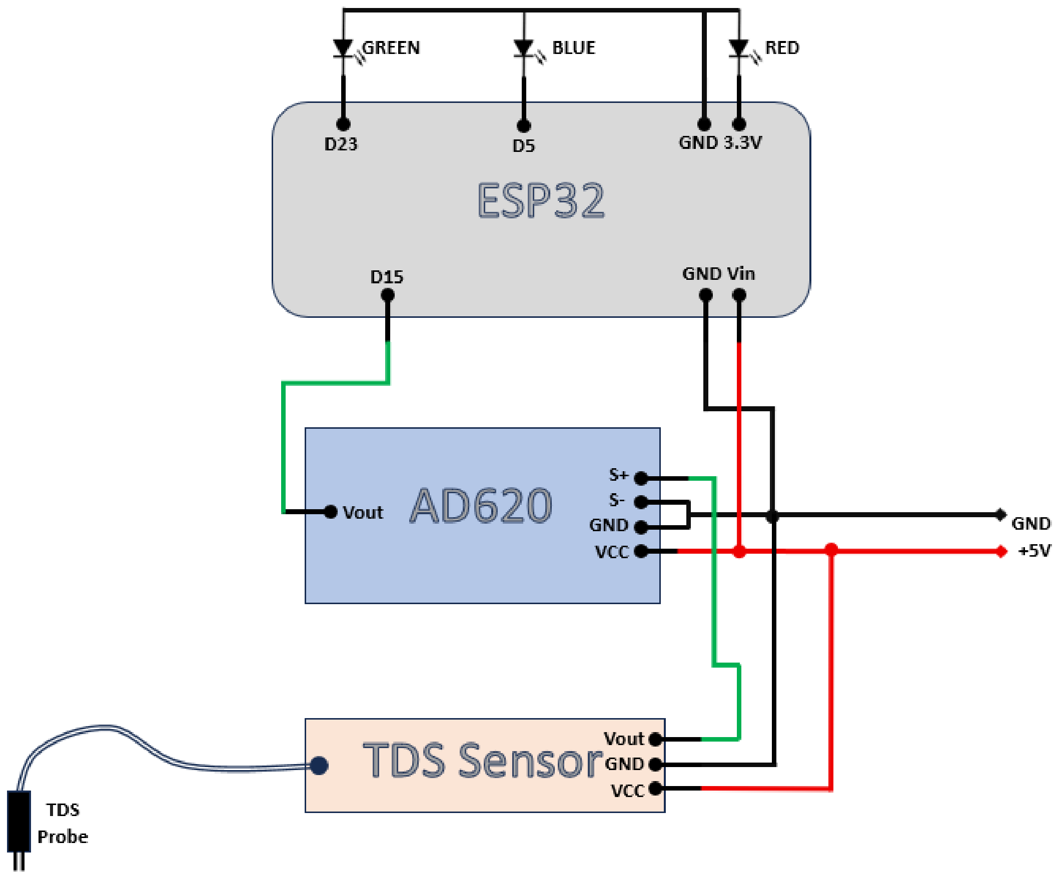


The project can be divided into mainly three units – (i) Collection Unit, (ii) Sampling Unit, and (iii) Distribution Unit.

The Collection Unit is the hardware part that collects rainwater during rainfall and diverts it to the Sampling Unit where the actual TDS value is determined by using TDS sensor and ESP32 microcontroller. The Distribution Unit is the software part that processes the collected data, uploads to Firebase cloud, and makes it accessible to mass users via a website.

The Electronics:

The main two electronic components are the ESP32 microcontroller and the TDS sensor (Model: Seeed Studio Grove Analog TDS Sensor). Apart from these, a voltage amplifier cum zero level shifter module (AD620, Make: Robu) was used in between TDS sensor and microcontroller for efficient detection. The sensor has a detection probe that is dipped inside the water to be tested for TDS, and it gives an along voltage at the output which is supplied to the ESP32 at the ADC pin 15. The maximum output voltage from the TDS sensor can be 2.3 V which is well below the maximum input limit of ESP32 device (which is 3.3 V). However, an amplifier is needed because when the water TDS is very small, the analog voltage output is below the minimum detection limit of pin 15 of ESP32 (typically 100 mV, but we found it experimentally to be 114 mV for our device). Therefore, the zero level of the amplifier (the steady output DC voltage when input is grounded) was set just above the said threshold value so that small TDS signals are also detected. The circuit connections are given below



Here *Vout* in TDS Sensor is the analog output voltage that appears when the TDS Probe is inserted in water. There are two inputs S+ and S- given in AD620 amplifier for AC signals. For DC signals like in our case, the S- is grounded and the signal is applied only via S+. There are two potentiometers in AD620 module one of which controls the gain, and the zero level can be adjusted with the help of other. The output of the amplifier is applied at pin 15 of ESP32. The Red LED in ESP32 indicates power, Blue indicates WiFi connection, and the Green indicates a successful upload.

Calibration:

In principle, both the amplifier and TDS Sensor should have a linear response to the input signal, however practically they may deviate from the linear behavior to some extent and therefore should be calibrated before use. The AD620 module is known to have an excellent linear profile but at small voltages we found small deviations as depicted in Fig.3. It can be seen that here the zero level is set to 150 mV, meaning, zero volt input shows 150 mV at output.

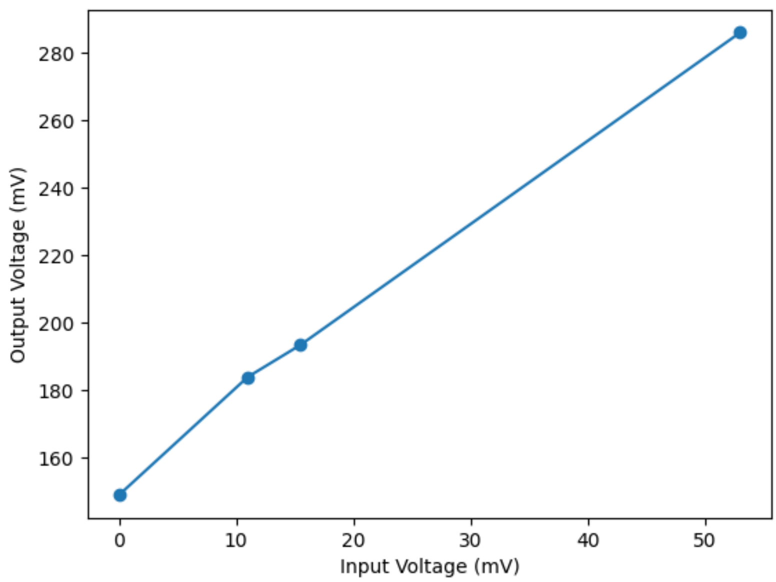


Fig.3 Calibration data for AD620 at small voltages

Even ESP32 ADC at pin 15 is not completely linear, and therefore the final voltage recorded at the device must be calibrated with respect to some standard TDS device so that a reliable value is obtained. We use a standard TDS meter (KONVIO-NEET) as shown in Fig.4. and 21 water samples with different TDS values. We identify three regions in the graph (Fig.5) with different slopes. The resulting equations are

Region-1: TDS_calculated= (((sensorVoltage-151)-3.55)/9.66) ------(1)

Region-2: TDS_calculated= (((sensorVoltage-151)-30.22)/6.84) ------(2)

Region-3: TDS_calculated= (((sensorVoltage-151)-136.75)/4.89) ------(3)

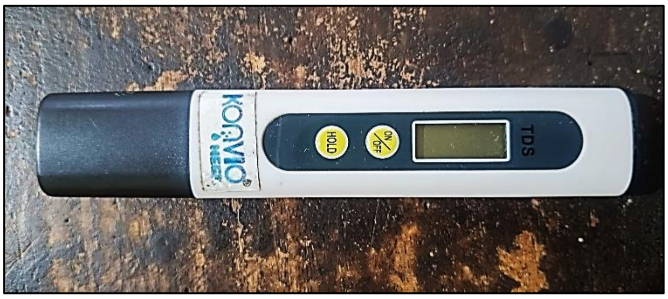


Fig.4 TDS meter used for calibration

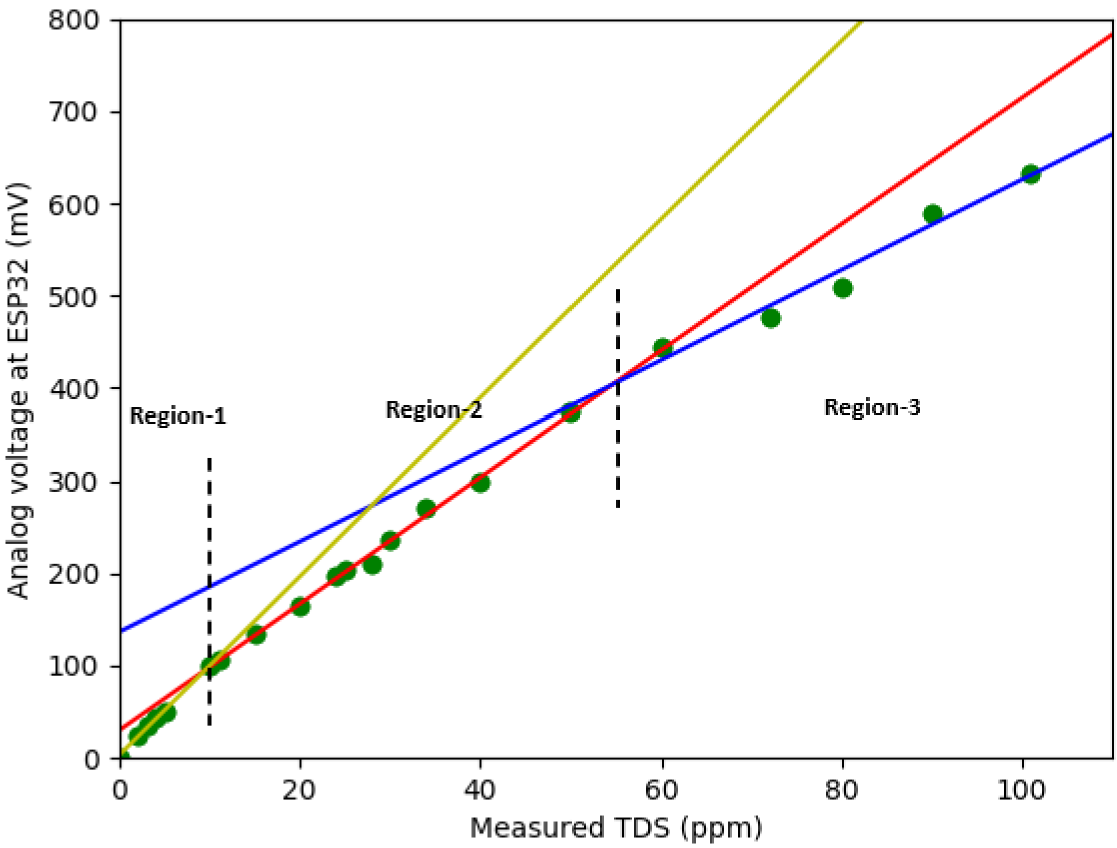


Fig.5 Sample TDS values measured with a standard TDS meter have been plotted against the analog voltages recorded by the ESP32. There are three regions with different slopes of fitted straight lines

Software and Distribution:

The ESP32 code used in this project can be downloaded from the Github location mentioned earlier. The ESP32 connects to a local WiFi to upload the data to Google Firebase realtime database – a platform that anyone can use for free. A simple android app also have been developed by using MIT App Inventor to change the WiFi credentials as required via bluetooth. A free website was created on Blogger to distribute the data. It pulls the data from Firebase using a javascript code and compiles into a csv file for download. The users must share their information before downloading starts for our records.

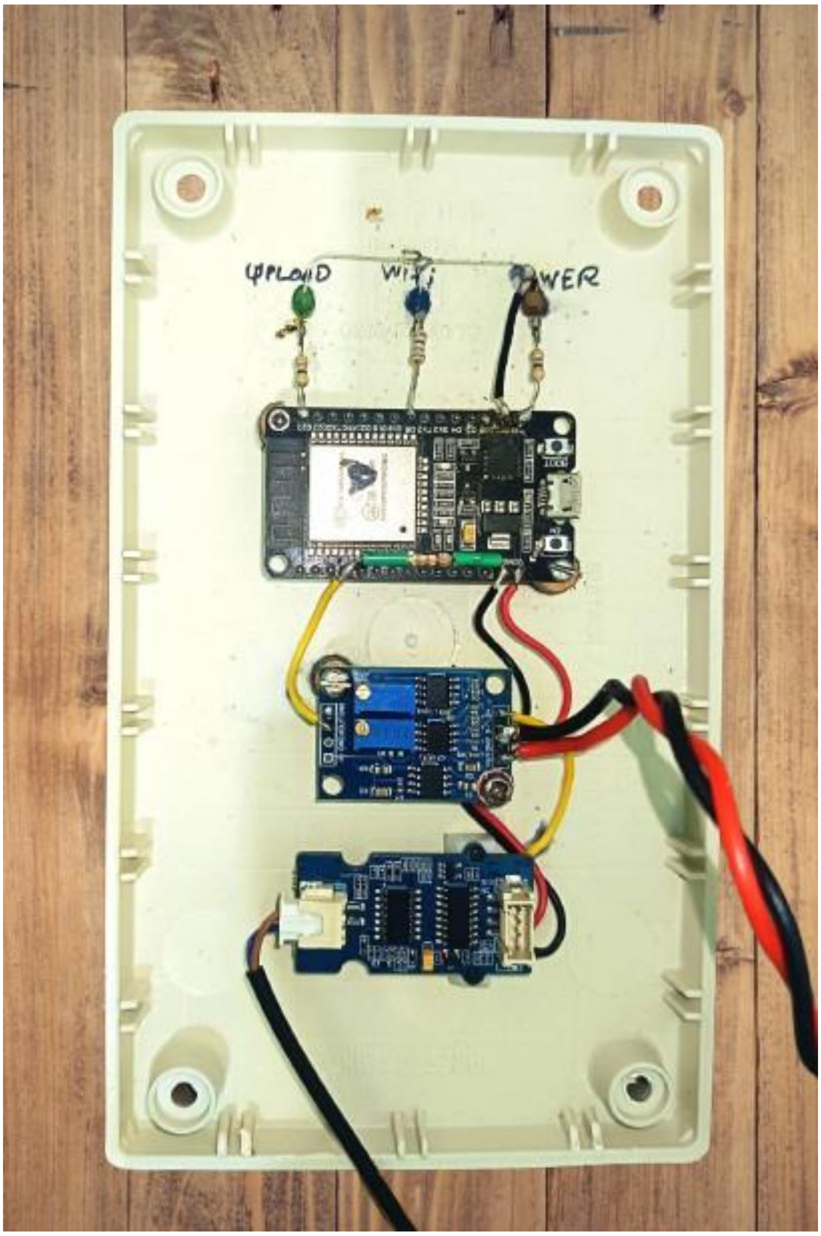


Fig.6 The circuit

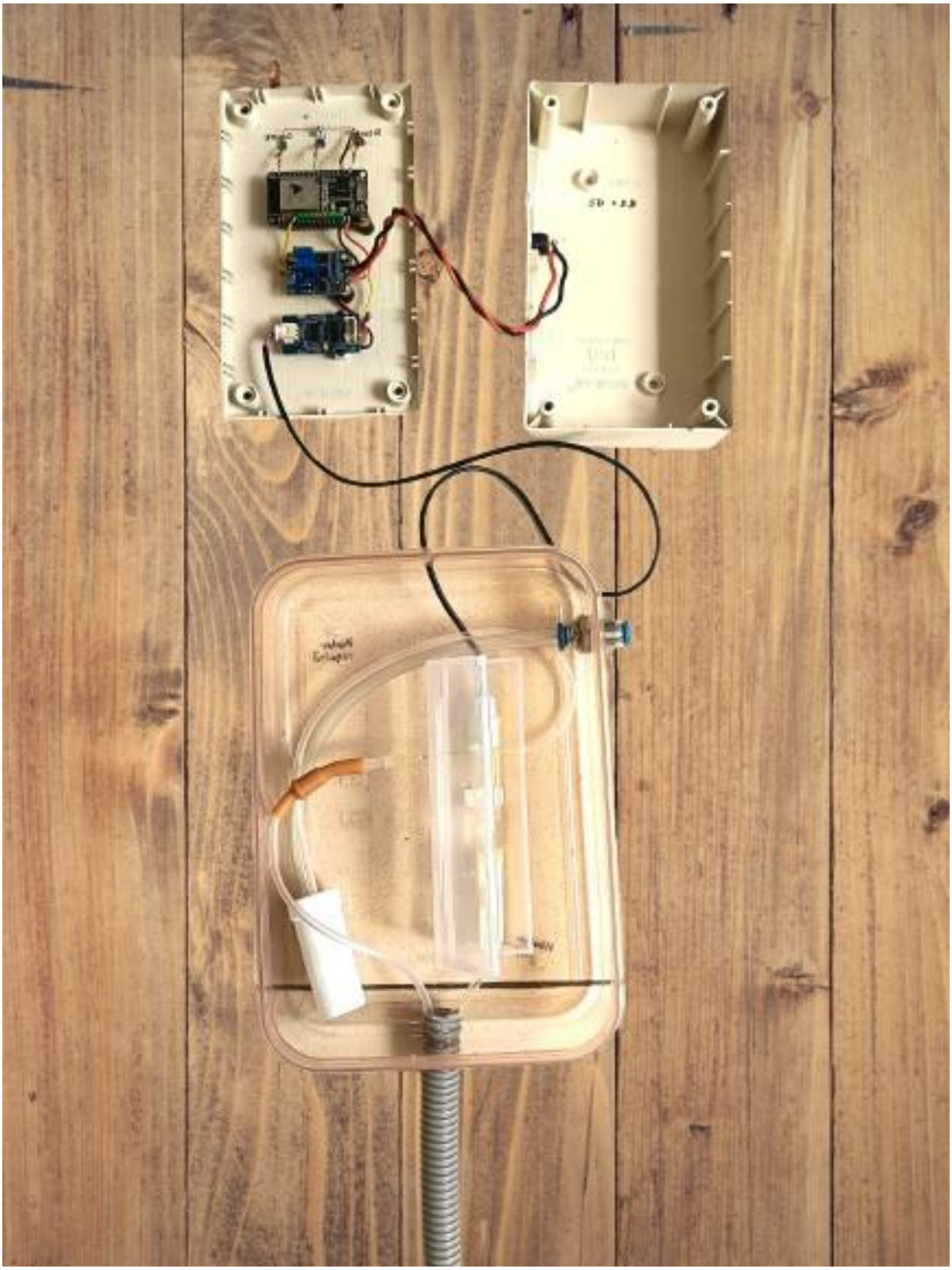


Fig.7 Whole arrangement