health4everyone
Team Romania

Vent&Fire Monitoring Station

The device monitors the concentration of oxygen and volatile organic compounds in the air and, based on them, triggers an alarm notifying excess oxygen or the need to vent the room. We used a control unit, an oxygen concentration sensor, a volatile organic compound concentration sensor, and an alarm to complete the device.

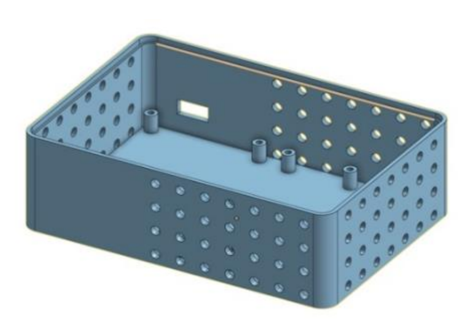
This device can be used independently or on a network (using one of the communication networks, WiFi, Lora, LTE).

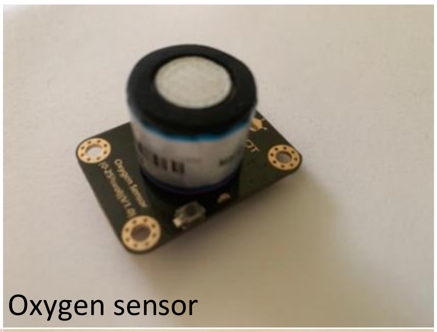
We have created a central device that receives information from monitoring devices over one of these networks for network operation. Thus, when a monitoring device triggers an alert, it will be identifiable. This solution will be advantageous when monitoring is carried out in restricted access areas (hospitals, buildings) where there is no permanent staff able to respond to the audible alerts provided by the monitoring devices.

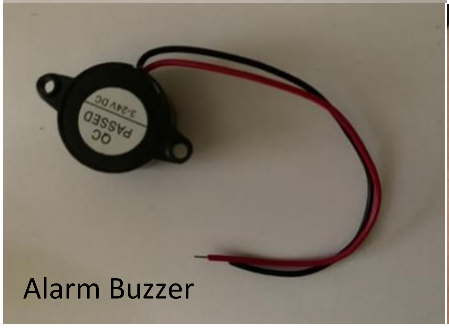
What you need:

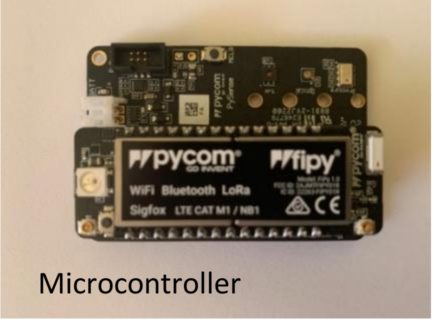
- VOC Air Quality Sensor (DFRobot I2C Oxygen Sensor)
- Oxygen sensor (DFRobot SGP40)
- PySense 2.0
- Microcontroller (FiPy 2.0)
- Alarm Buzzer
- 3D printed box
- Wires

VOC Air Quality Sensor

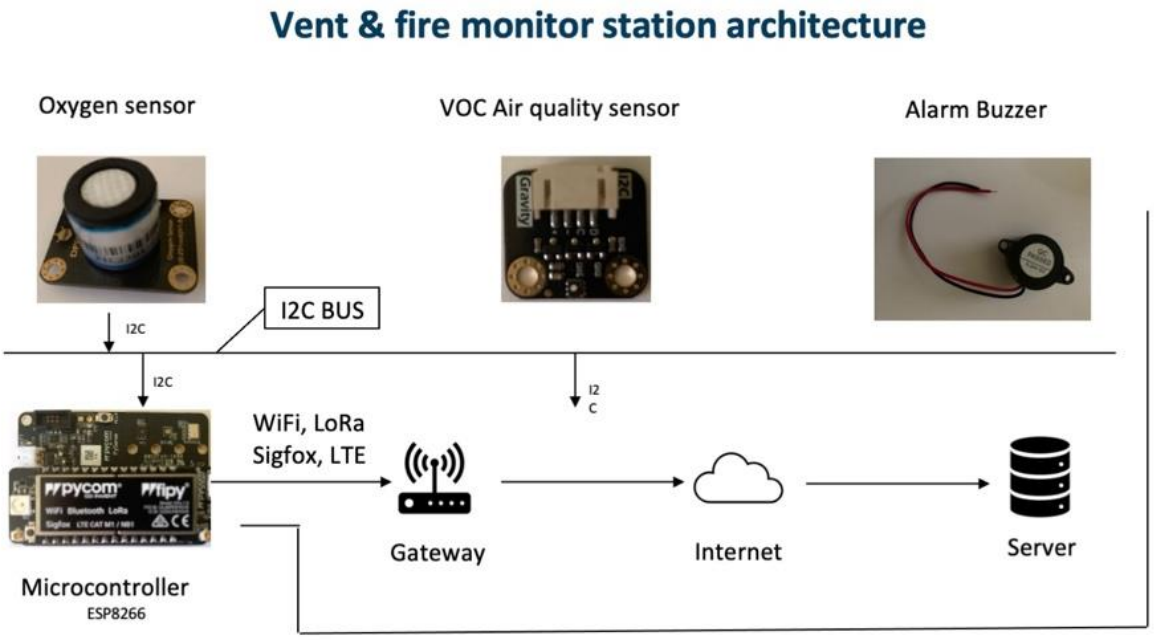



Oxygen sensor


Alarm Buzzer


Microcontroller

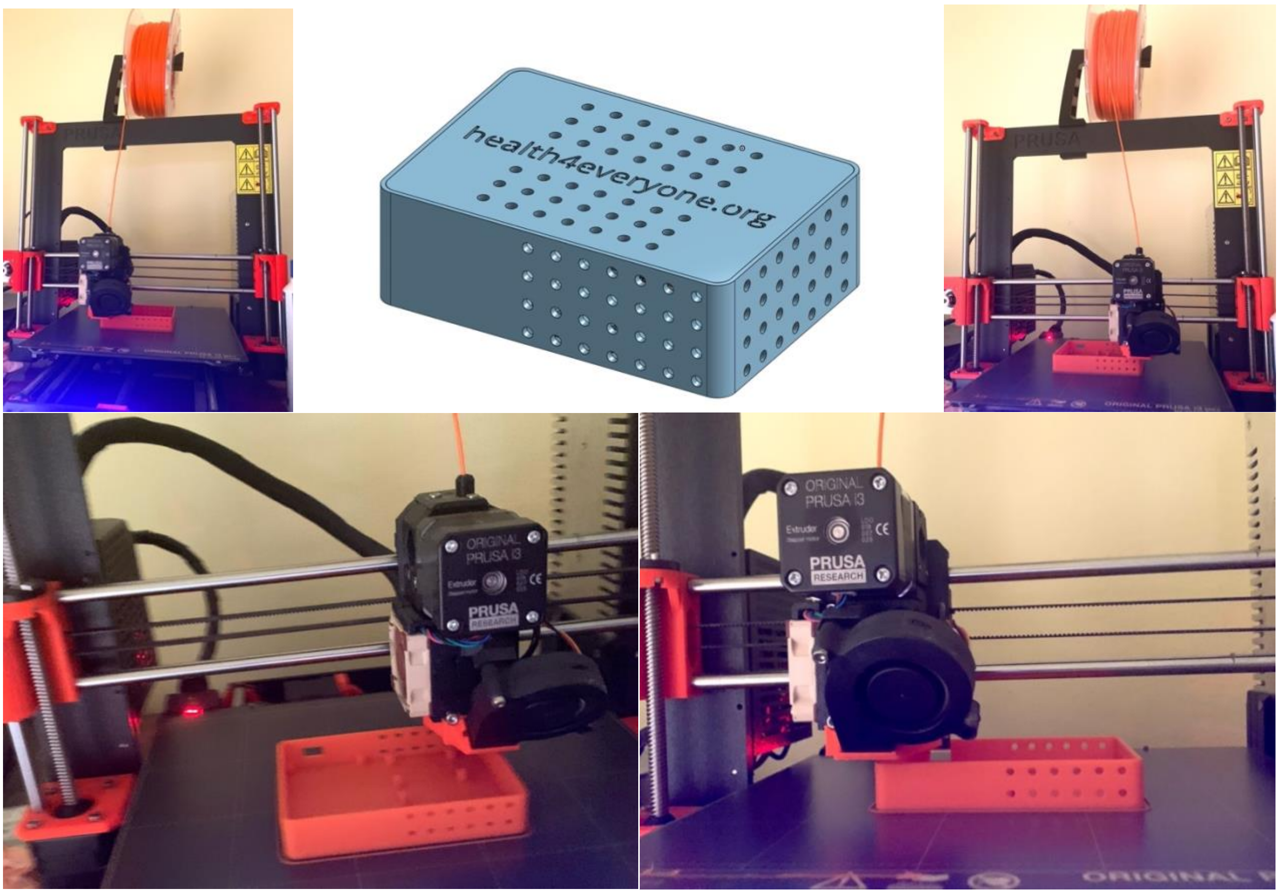
Architecture



You can use as many sensors or any sensors as you want but keep in mind to wire the readings and send them to the monitoring application.

3D box

To have a nice box that keeps things together, we designed and printed a box with places for every component. We did that in OnShape that offers free licensing for educational purposes. You can find the files in the repository for what we did..



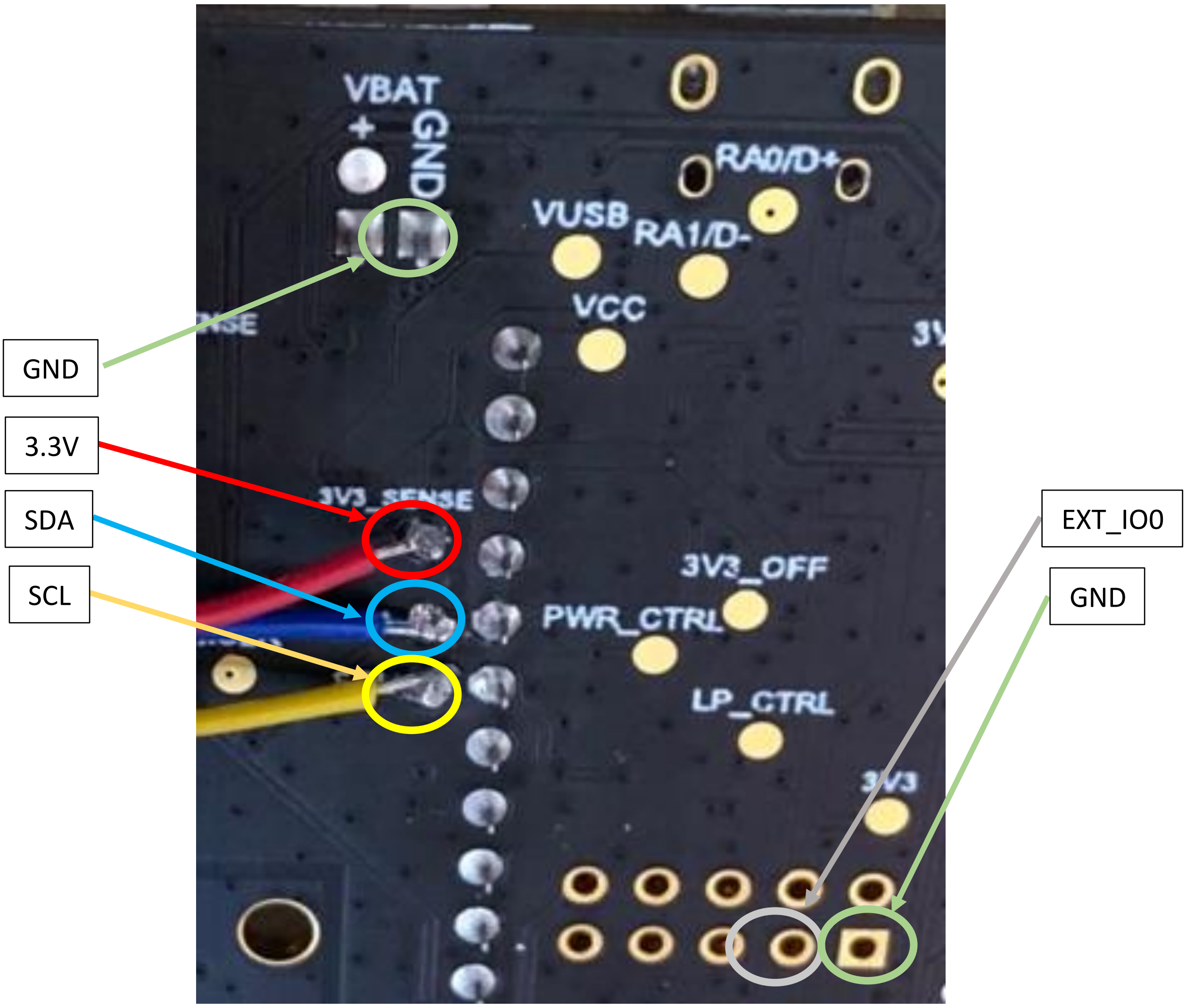
Wiring

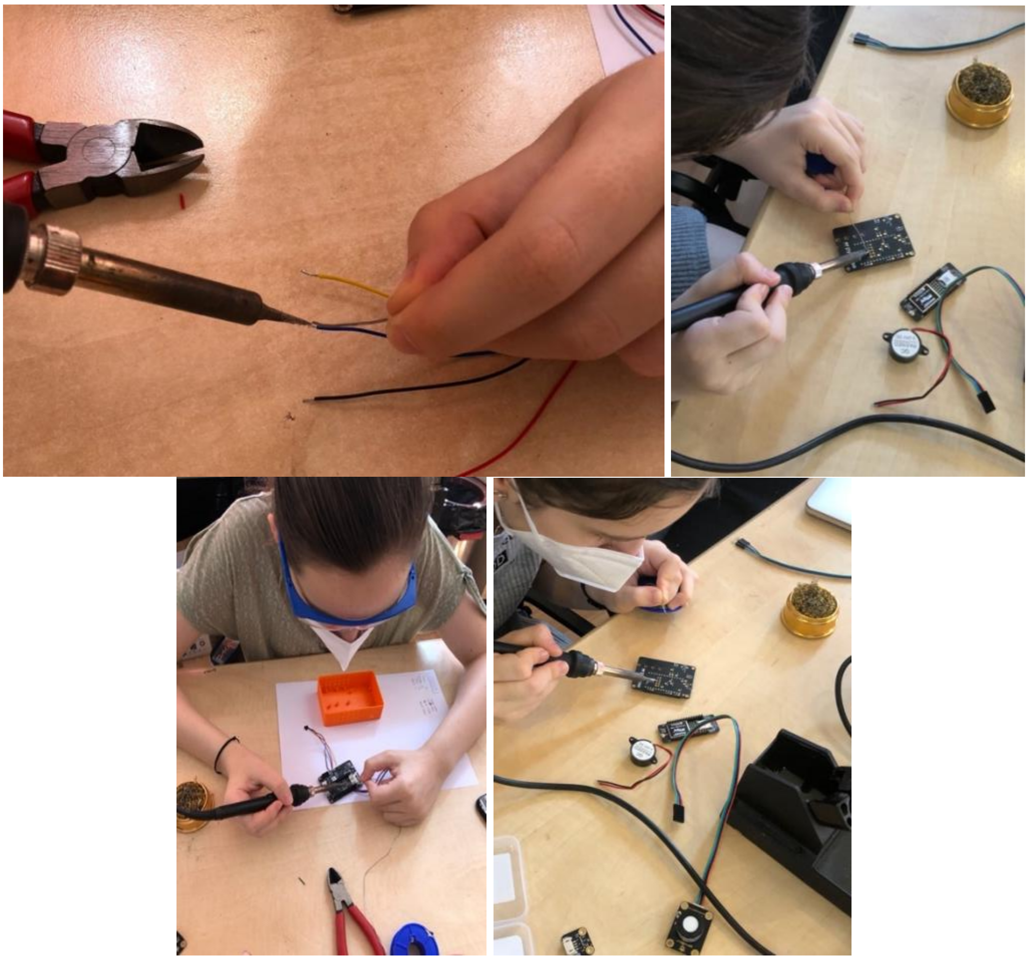
Our sensors work using I2C bus communication protocol, so we need to connect them to the correct bus for that to work. PySense 2.0 module that we used includes some sensors that provide information like pressure, temperature, humidity, light, and in addition to that, we added the oxygen sensor and a VOC sensor.

For added sensors to work, we need to connect them to power and I2C communication bus. To do that, we soldered four wires that will carry 3.3V, GND, SDA, and SCL signals. We used a Sparkfun Qwiic Multiport adapter and some qwiic wires to make the soldering part easier, but that is optional.

Here are the pins we used on the back of PySense 2.0 shield. We decided to use 3V3_SENSE pin since it can be turned on/off if required. Also, you can use any GND for the fourth wire; we used one that is not in the picture. Finally, keep track of all signals, so each sensor is correctly connected to power and signals.

We also used EXT_IO0 pin for the buzzer and another GND to connect the alarm.





Assembly

After we did all the wiring, we started mounting all the parts in the case we printed. You'll notice all parts have mounting holes and spacers based on specific sensors. We used M2x5mm screws for that. You can see the result below.