

Phytoplankton Air Systems

Algaerithms Inc.



Dhairya Pal (N01576099)
Julian Imperial (N01638310)
Dharmik Shah (N01581796)
Sanskriti Mansotra (N01523183)

Project Overview

Phytoplankton → Oxygen (Photosynthesis)

Algae tank placed under a water reservoir that supports the algae's growth environment.

Four sensors are embedded in the system:

- A light sensor to monitor brightness for optimal photosynthesis,
- A proximity sensor to detect nearby human interaction,
- A turbidity sensor to assess the health of the algae culture by analyzing water clarity, and
- An ENS160 sensor, which estimates the ambient carbon dioxide concentration to infer the rate of CO₂ consumption.

The Product



Water reservoir

Carbon dioxide sensor

Phytoplankton solution

Light sensor

Turbidity sensor

Proximity sensor

ENS160

- ENS160 can very efficiently estimate eCO₂ (equivalent carbon dioxide) levels using its metal oxide (MOX) gas sensing array and embedded AI algorithms.
- Unlike basic CO₂ sensors, ENS160 can estimate volatile organic compounds (TVOCs) and eCO₂ in a compact form factor.

SparkFun Breakout Board specs

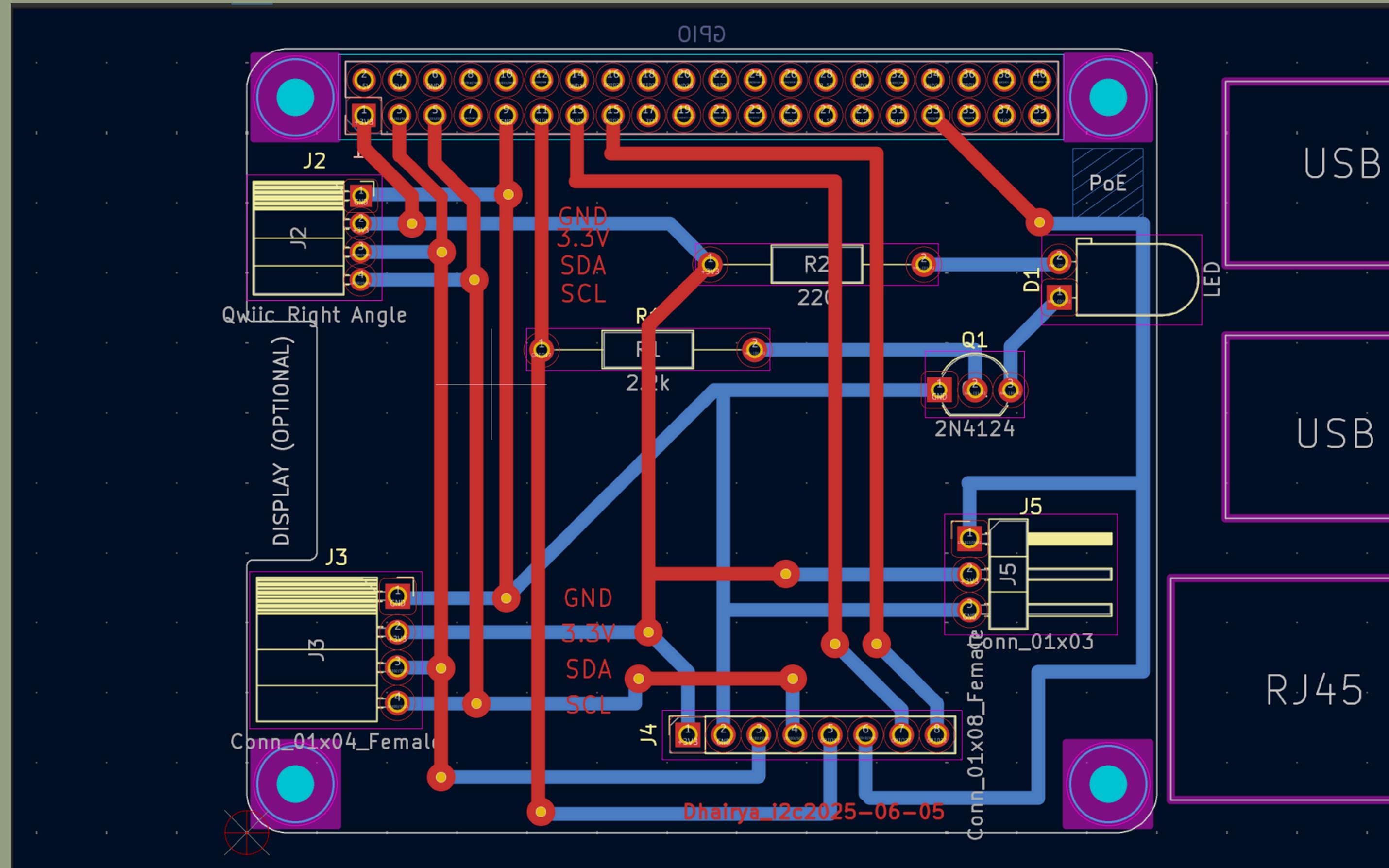
Sensor Type: ENS160 MOX-based gas sensor

Breakout Board: SparkFun Qwiic ENS160 (I²C)

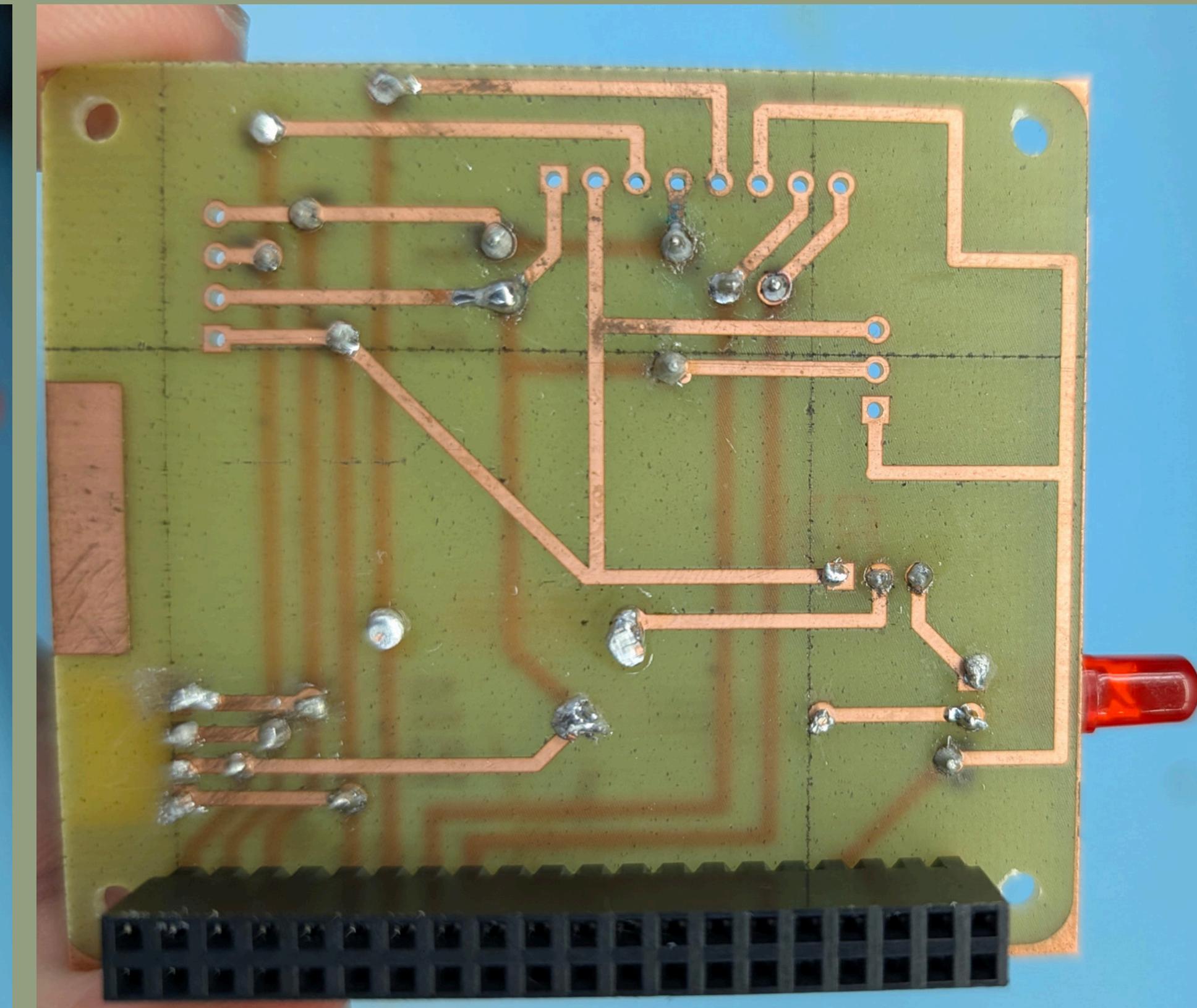
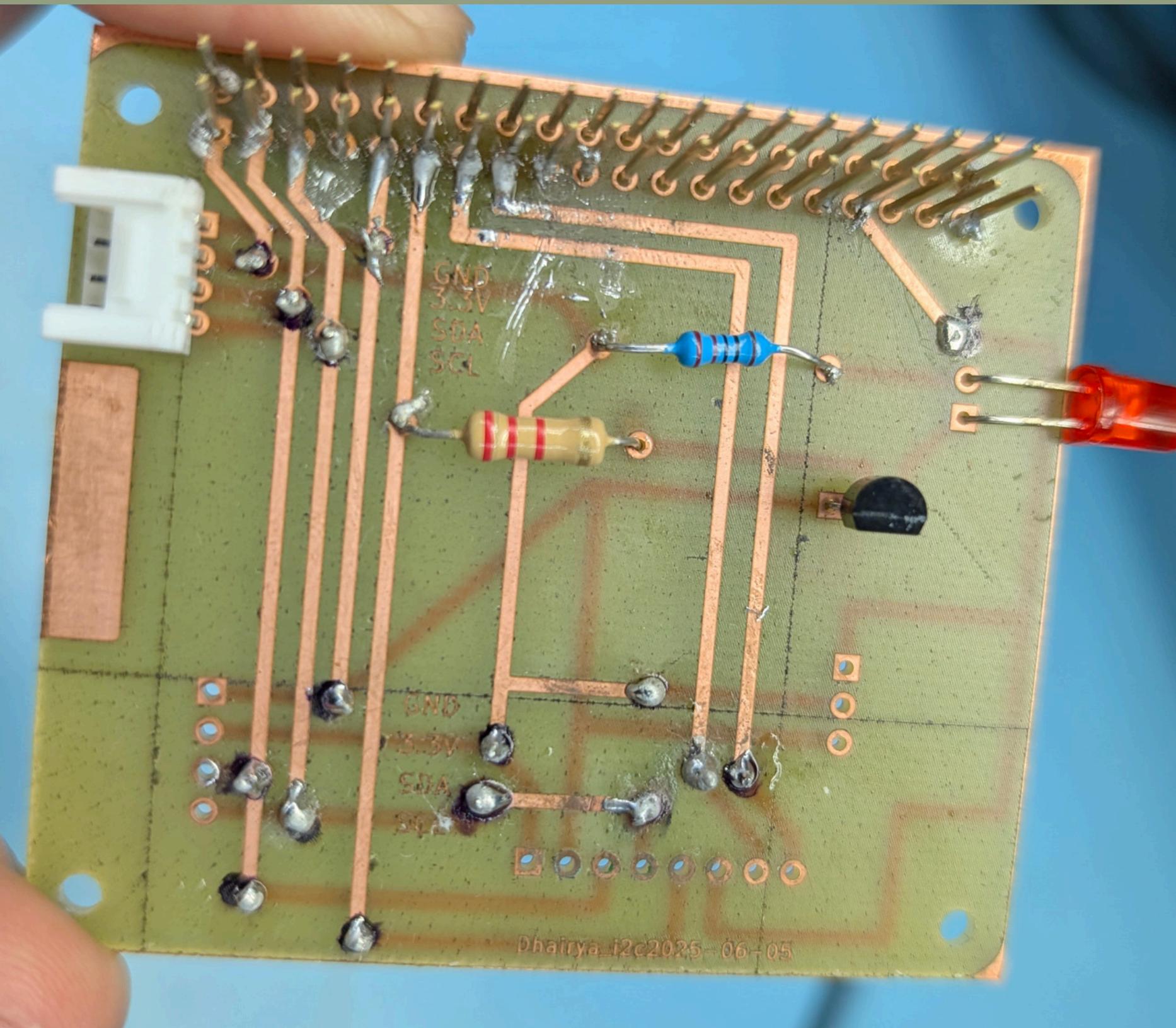
Key Features:

- Measures eCO₂, TVOC, AQI
- Operating Voltage: 1.8–3.3V
- I²C Address: 0x53
- Communication: I²C (Qwiic-ready), fully supported by Raspberry Pi
- Warms up ~3 mins
- Speed ~1Hz (I²C allows faster but ENS160 internal update rate is ~1Hz)

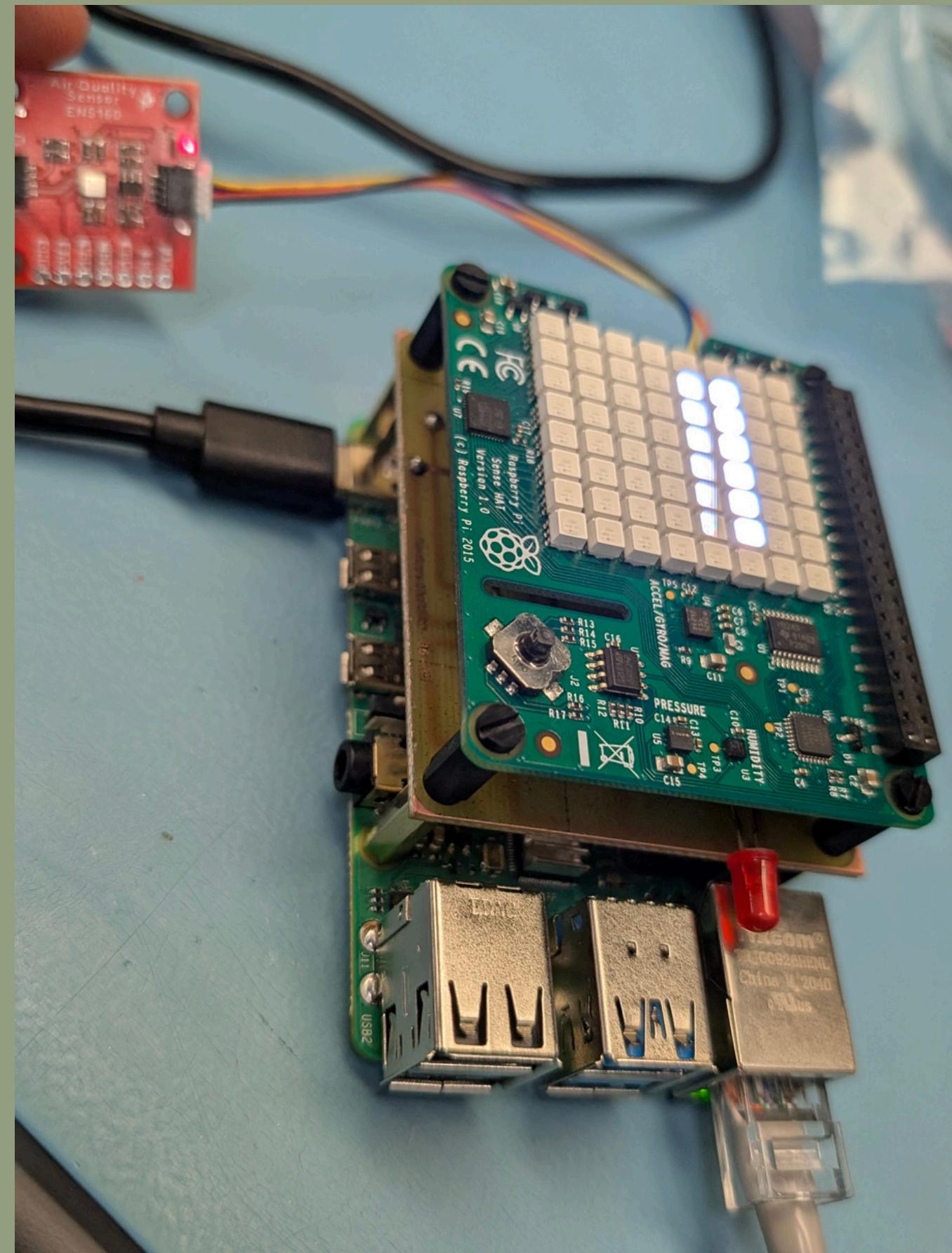
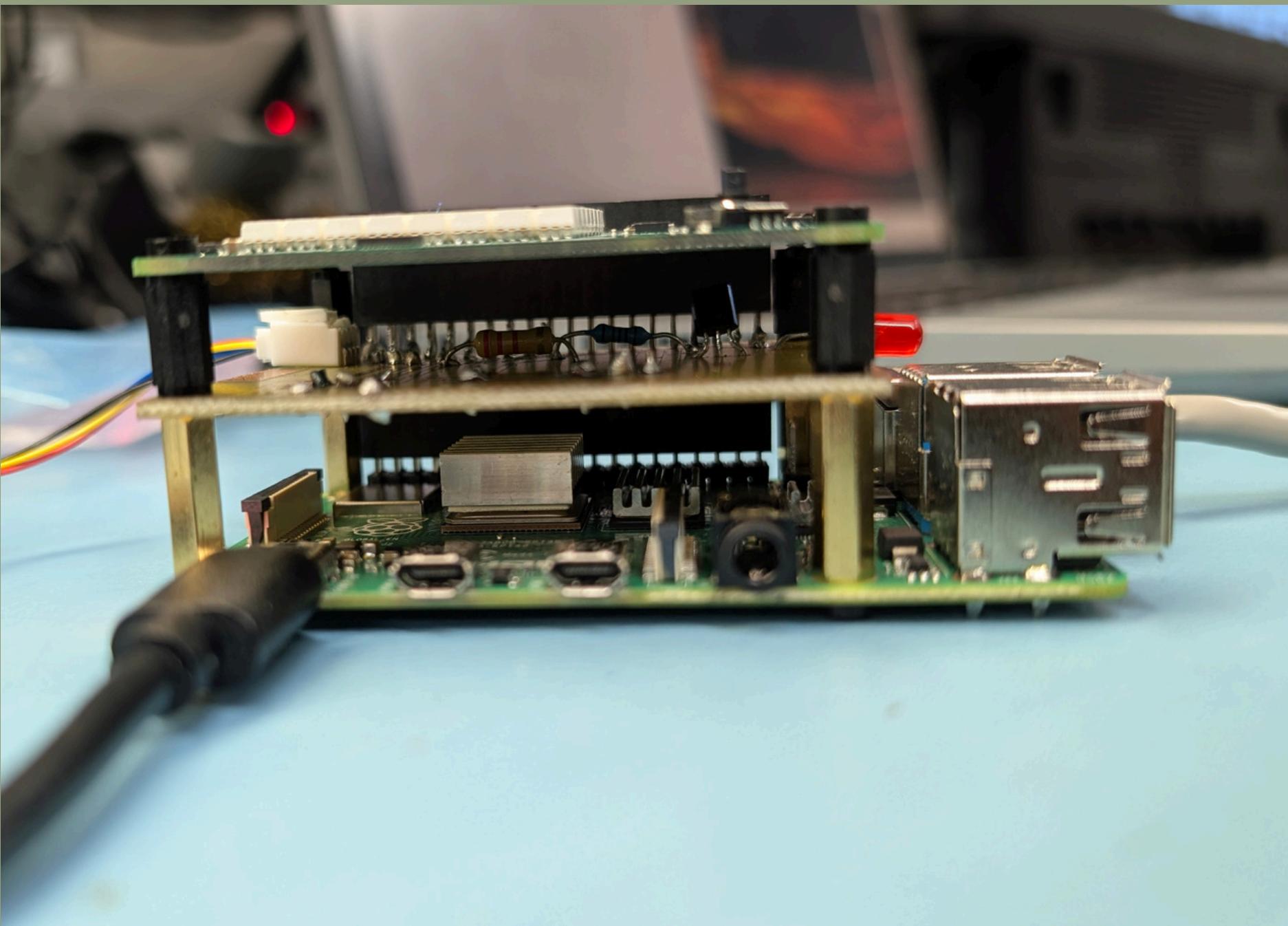
Designed PCB



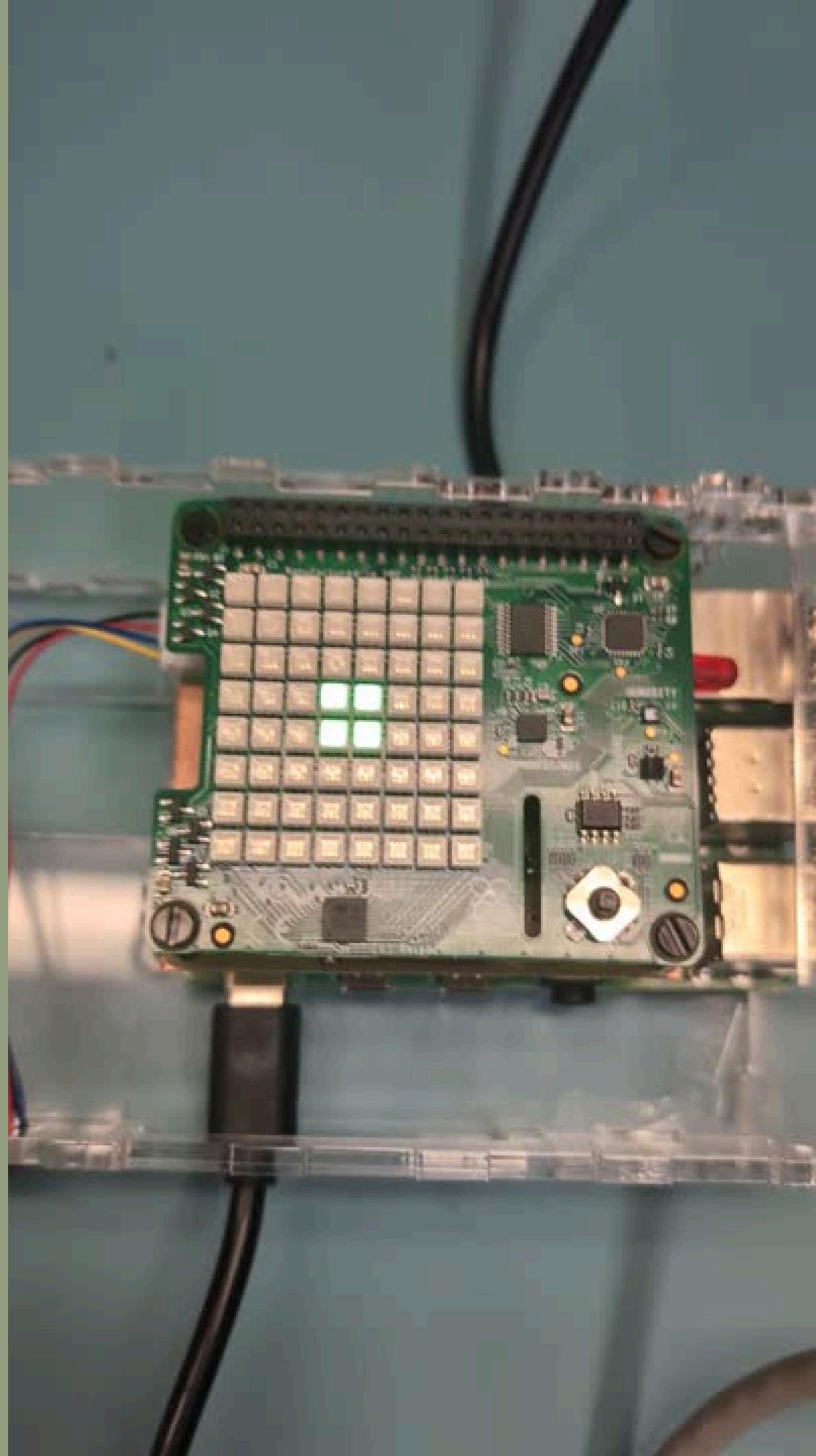
Printed PCB



Assembled hardware



Demo



Code

```
def read_sensor():
    print("Starting ENS160 example...")

    ens160 = qwiic_ens160.QwiicENS160()

    if not ens160.connected:
        print("Sensor not connected. Check your wiring.")
        return

    ens160.begin()
    ens160.set_operating_mode(2)

    time.sleep(1)

    while True:
        ens160.check_data_status()
        eco2 = ens160.get_eco2()
        co2_kg = convert_ppm_to_kg_co2(eco2, 0.005)
        co2_kg = round(co2_kg, 10)
        timestamp = time.strftime("%Y-%m-%d %H:%M:%S")

        update_data = { 'co2_converted': co2_kg, 'timestamp': timestamp }

        device_ref = db.reference('device_001')
        device_ref.update(update_data)

        print(f"Updated co2_converted to {eco2} at {timestamp}")
        time.sleep(5)
```

Sensor updates Firebase values and timestamp

The screenshot illustrates a development environment for a Raspberry Pi connected to a sensor (ENS160). The left window is a web browser displaying the Firebase Realtime Database console. The right window is a Thonny IDE running on the Raspberry Pi.

Firebase Realtime Database Console:

- Project: Phytoplankton Air Systems
- Database URL: https://phytoplankton-air-systems-default-rtdb.firebaseio.com/
- Security Rule Warning: Your security rules are defined as public, so anyone can steal, modify, or delete data in your database.
- Database Structure:
 - device_001
 - algaehHealth: 55
 - co2Converted: 0.00000585
 - light: 2
 - proximity: false
 - timestamp: "2025-07-30 20:54:45"
 - turbidity: 160
 - waterlevel: 25
- Database location: United States (us-central1)

Thonny IDE (Raspberry Pi):

```
ens160.begin()
ens160.set_operating_mode(2)

time.sleep(1)

while True:
    ens160.check_data_status()
    eco2 = ens160.get_eco2()
    co2_kg = convert_ppm_to_kg_co2(eco2, 0.005)
    co2_kg = round(co2_kg, 10)
    timestamp = time.strftime("%Y-%m-%d %H:%M:%S")

    update_data = { 'co2Converted': co2_kg, 'timestamp': timestamp }

    device_ref = db.reference('device_001')
    device_ref.update(update_data)

    print(f"Updated co2Converted to {eco2} at {timestamp}")
    time.sleep(5)

if __name__ == "__main__":
    try:
```

Shell output:

```
>>> %Run firebase_upload.py
Starting ENS160 example...
Updated co2Converted to 687 at 2025-07-30 20:54:39
Updated co2Converted to 650 at 2025-07-30 20:54:45
```

Challenges, solutions, and lessons learned

- Using KiCad, InkScape for the first time
 - Did research
 - Kept patience
- Cold soldering for a few connections – the 40 pin header and the Qwiic connector
 - Resoldered connections nicely and removed the extra cold solder.
 - Checked electrical continuity before moving on to connecting the sensor.
- Sensor light wasn't turning on
 - Connector was loose

Be precise and careful while handling hardware, and ask a lot of questions – to prof, and do your research.

CENG317 Hardware
Production Technology

Thank you

Algaerithms Inc.