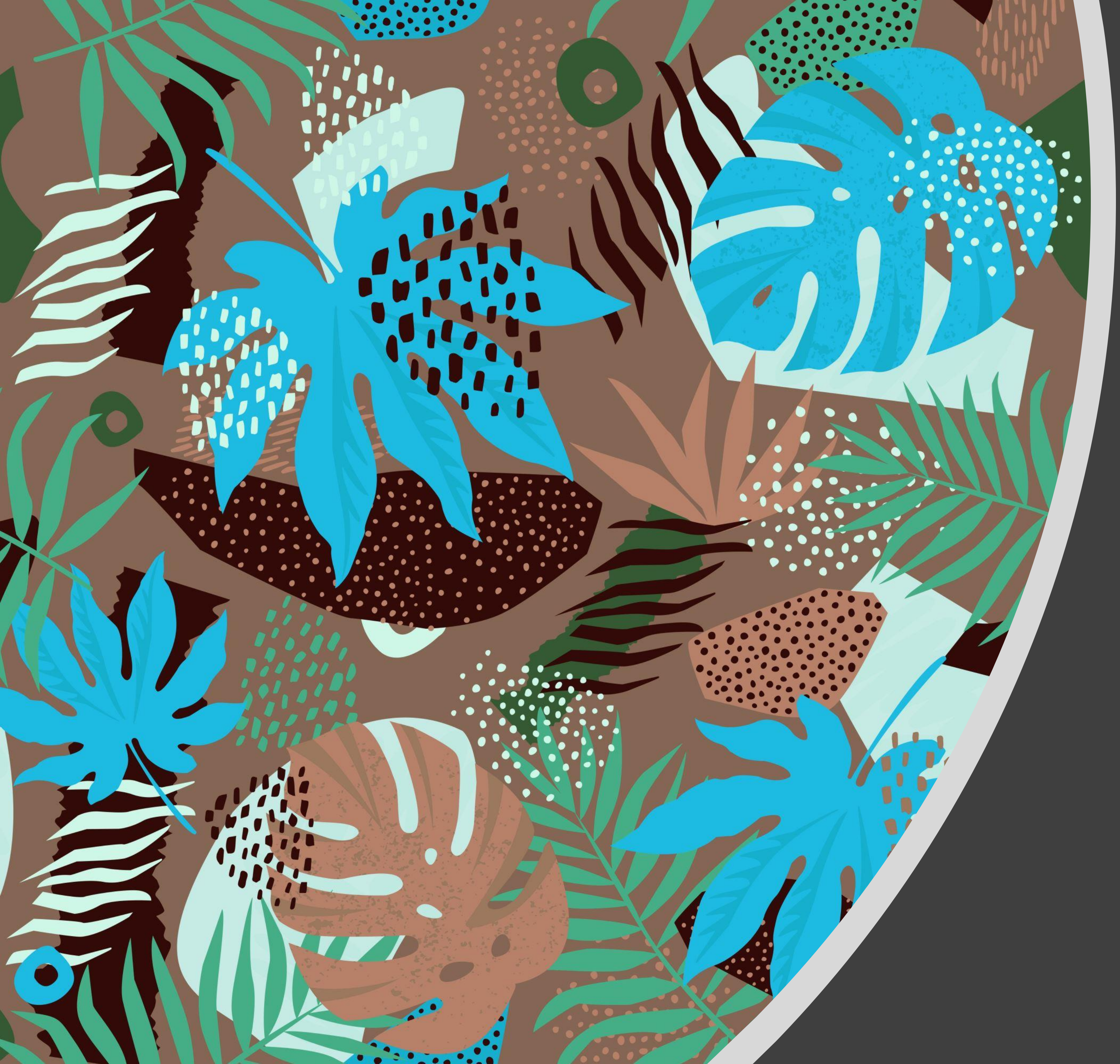


Wildfire Detection System

By Quad Core Crew

Introduction

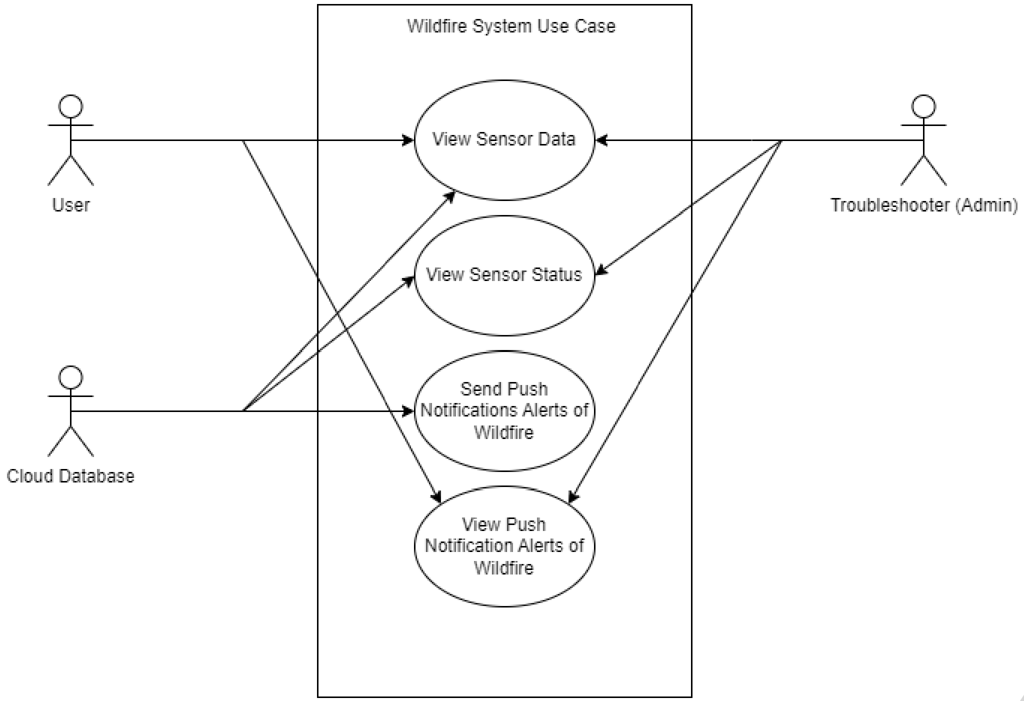
- A problem of today? Destruction of forests due to forest fires. Not only that, but destruction of homes, livestock, and wildlife
- This puts an economic burden on society, government, insurance rates, etc.
- The new normal?

Introduction

- Proposed solution?
- Introducing – the "WildFire" Detection System
  - An outdoor wildfire detection system, utilizing photoelectric and ionization smoke detection sensors
- The "Wildfire" Detection System will be capable of notifying command control of a forest or national park when there is a burning forest fire
- The system will solve the problem of mass destruction of forests and loss of life by providing an alert to command control. This in turn will reduce the damage forest fires have on the environment, the economy and life

Objectives

- Detect wildfires in fire prone areas outdoors.
- Use temperature, photoelectric, ionization, humidity, and wind sensors to determine the presence of fire.
- Use a combination of existing “on the shelf” technology.
- Locate direction and speed of burn.
- Send notifications of fire using cellemetry.
- Locate each unit using GPS and GSM triangulation.
- Rechargeable using solar panels.
- Units positioned in a grid over a large area.



Design

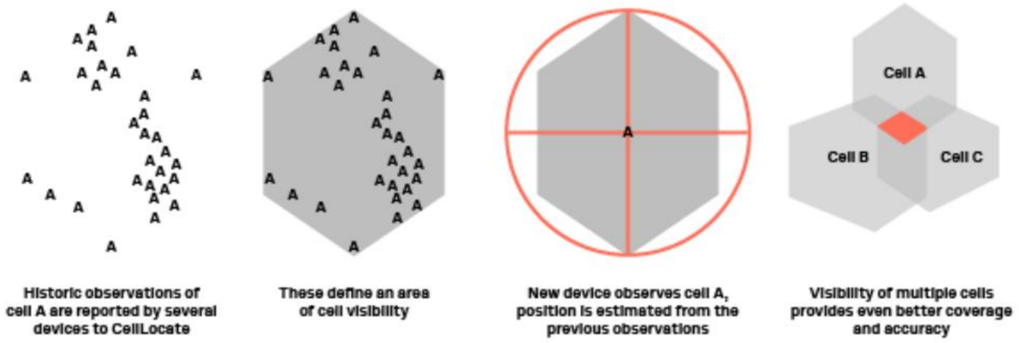
- System will detect the location, direction, and speed of a possible fire and notify users and admins.
- Users can view the alerts and sensor data on the website.
- Admins can view the alerts, data, and sensor status on the website.
- Admins can also view the alerts, data, and status using serial connections to each unit.
- The cloud database service will store all the information passing from the units to the website.
- The service we are using is AWS IoT and DynamoDB.
- The database can read data and status and send alerts.

Implementation – Location

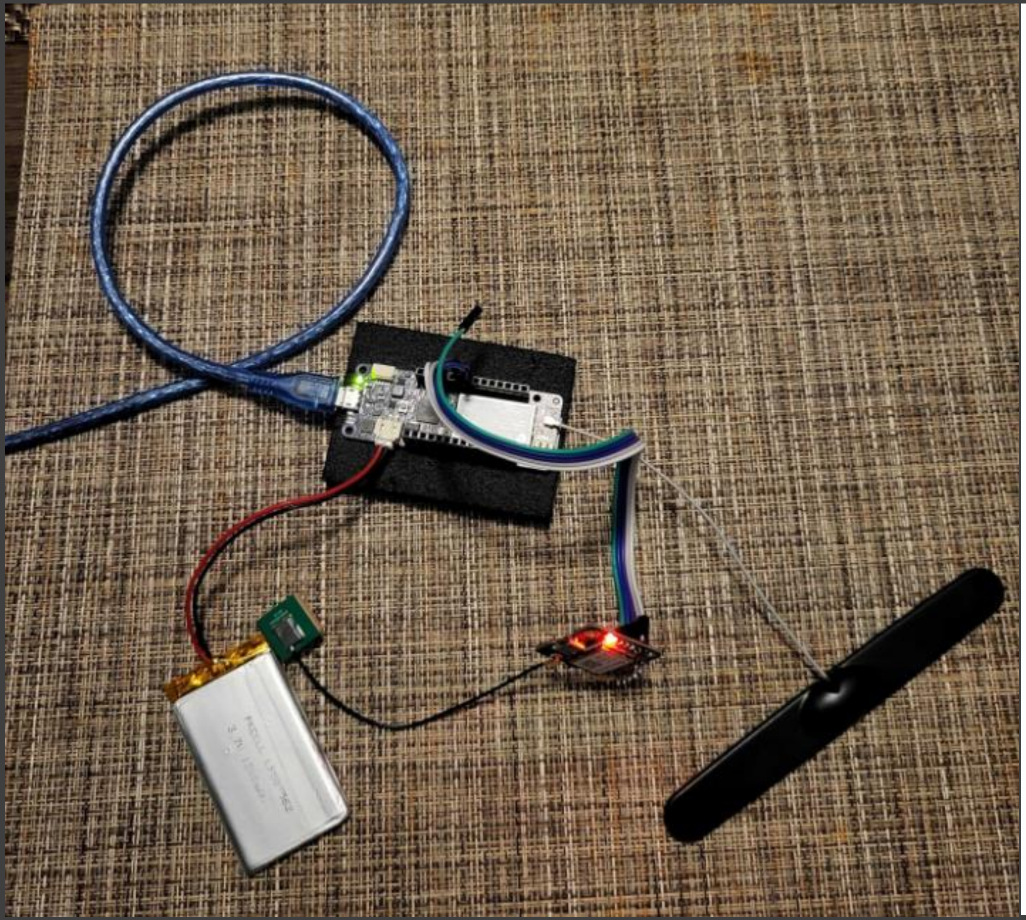
- How are we locating the "WildFire" device?
  - GPS and Triangulation using cell towers
- GPS
  - GPS module, NEO-6M is interfaced to our microcontroller, the Arduino MKR GSM 1400 through serial communication
  - The program tries to find coordinates using the GPS module
  - GPS coordinates are our default coordinates
  - If no coordinates can be found in 4 minutes, the module times out and no GPS coordinates are set. Coordinates found by triangulation of cell towers should be used instead

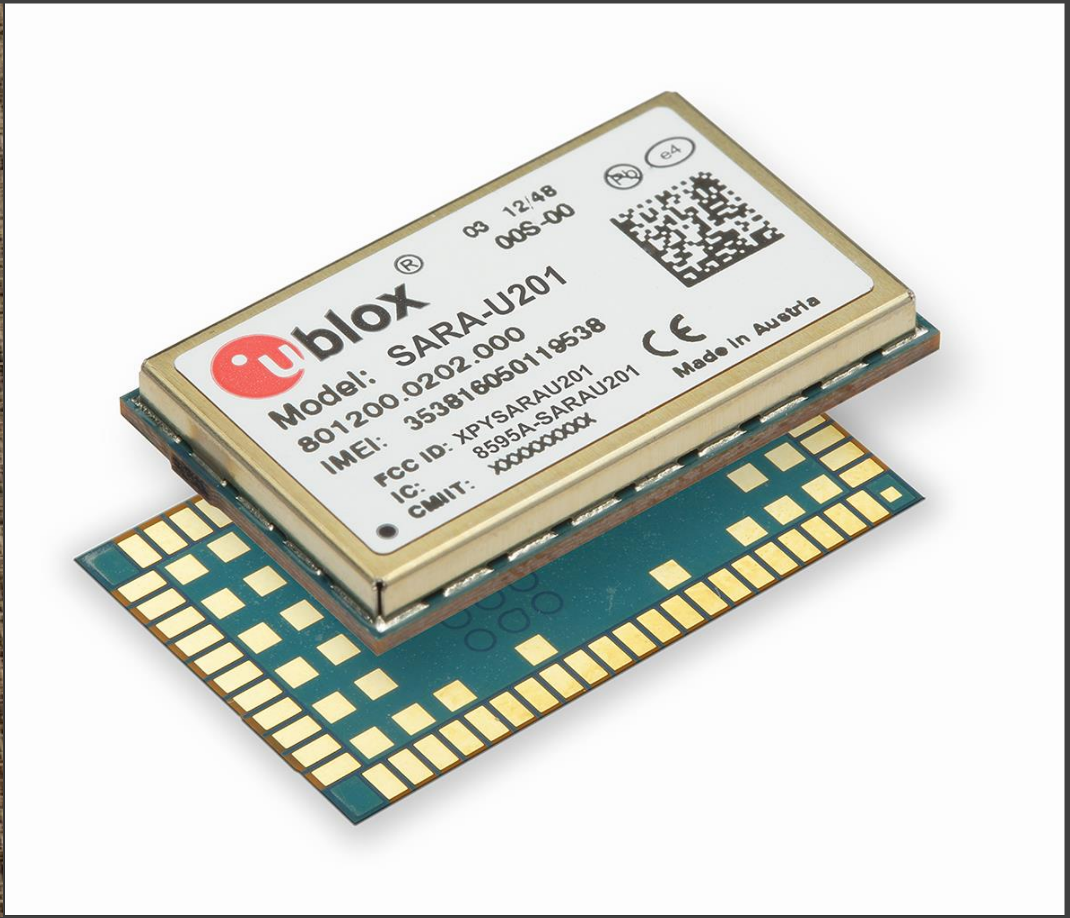
Implementation - Location

- Triangulating cell towers
  - uBlox Sara-201 GSM module has proprietary service called cellLocate
  - When locating, triangulates cell towers and previous pings from other devices near network
  - From this information, if finds out what cell the device is in
  - Further optimizations are done to the location coordinates as cellLocate sees what cells the device is in
  - Finally, a more accurate coordinate is output. If there is no coordinate, routine times out after 4 minutes.



Implementation - Location

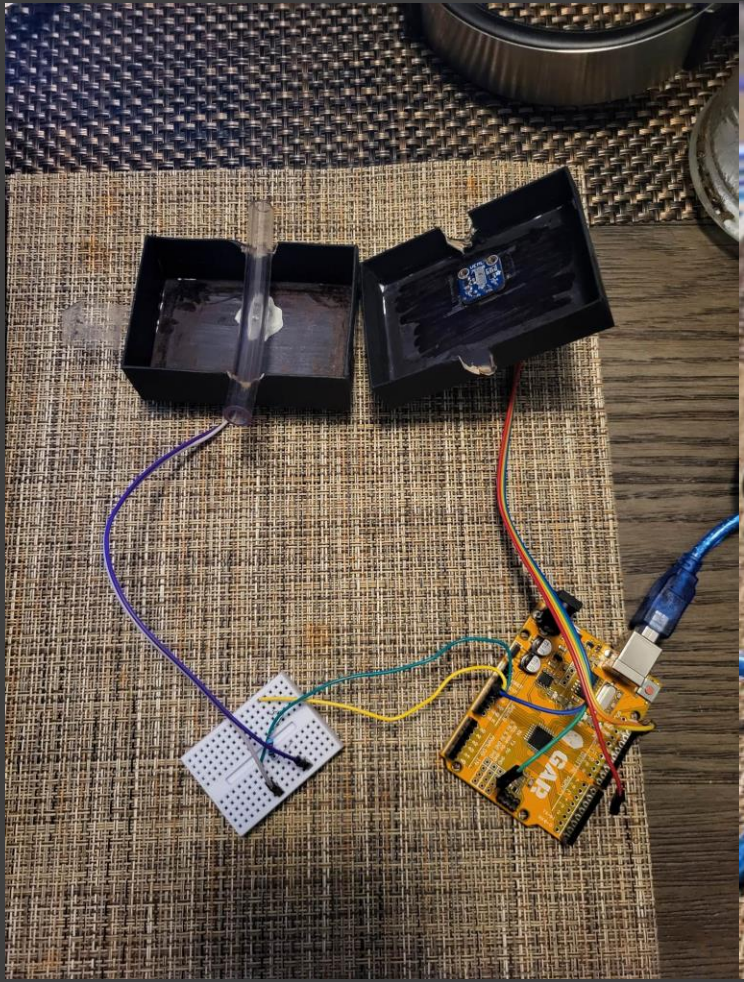


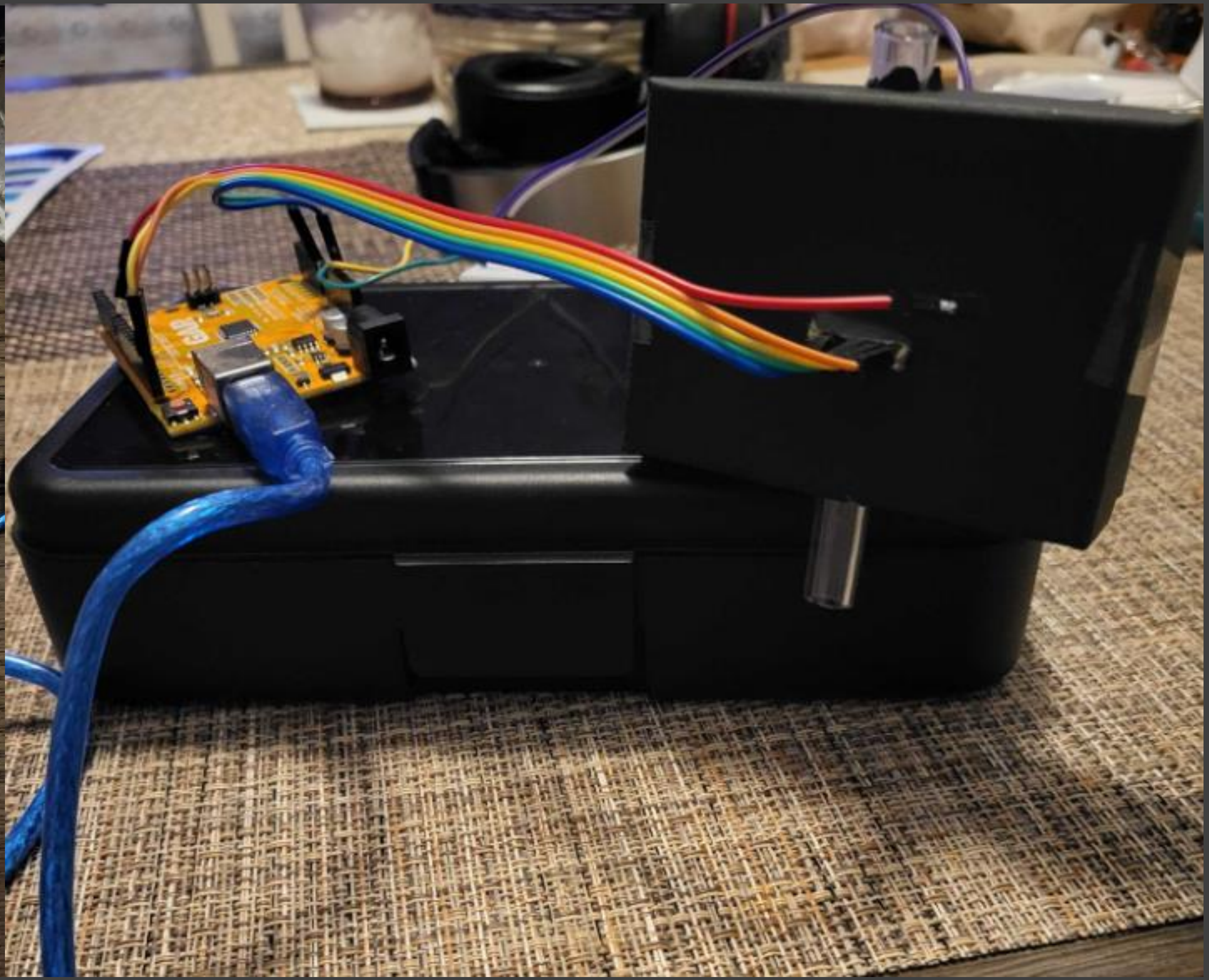


Implementation – Photoelectric sensor

- The photoelectric sensor is a box containing a white LED facing a VEML7700 LUX sensor. A clear plastic tube is sandwiched between the LED and the LUX sensor
- When smoke travels up the plastic tube, the LUX values change, indicating presence of smoke.
- Combustion particles are not detected with this implementation of the photoelectric sensor.
- The LUX sensor communicates to the microcontroller via I2C

Implementation – Photoelectric sensor





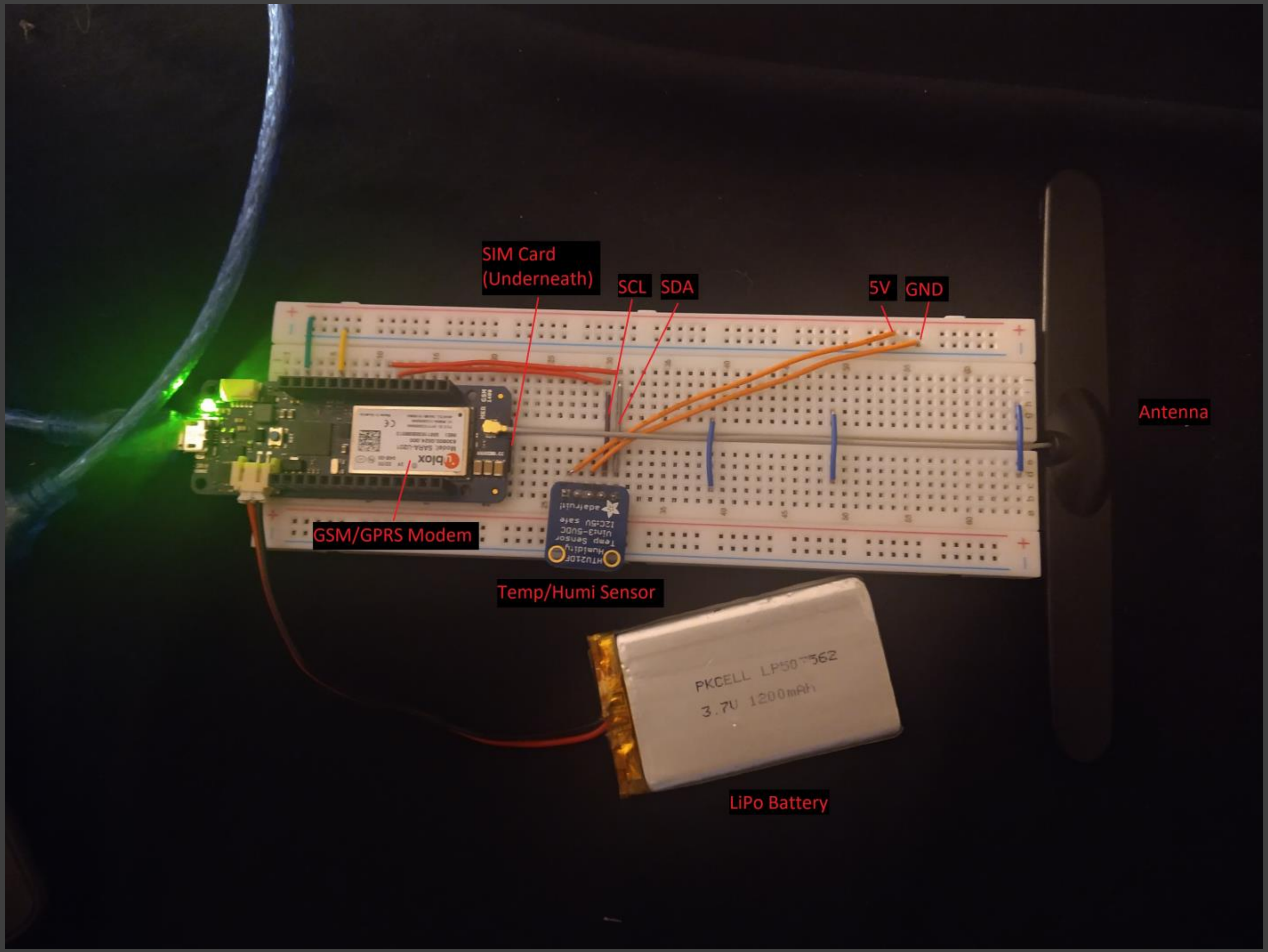
Implementation – Temperature / Humidity Sensor

- Uses HTU21D-F temperature and humidity sensor.
- Originally used DHT20, but the sensor broke.
- -40 to 120 degrees Celsius operating range.
- 0 to 100% humidity operating range.
- 3-5V operating voltage.
- I2C connections.
- Exposes a wire with voltage to the air and calculates the temperature and humidity based on the change in the voltage.

Implementation – GSM / GPRS Connections

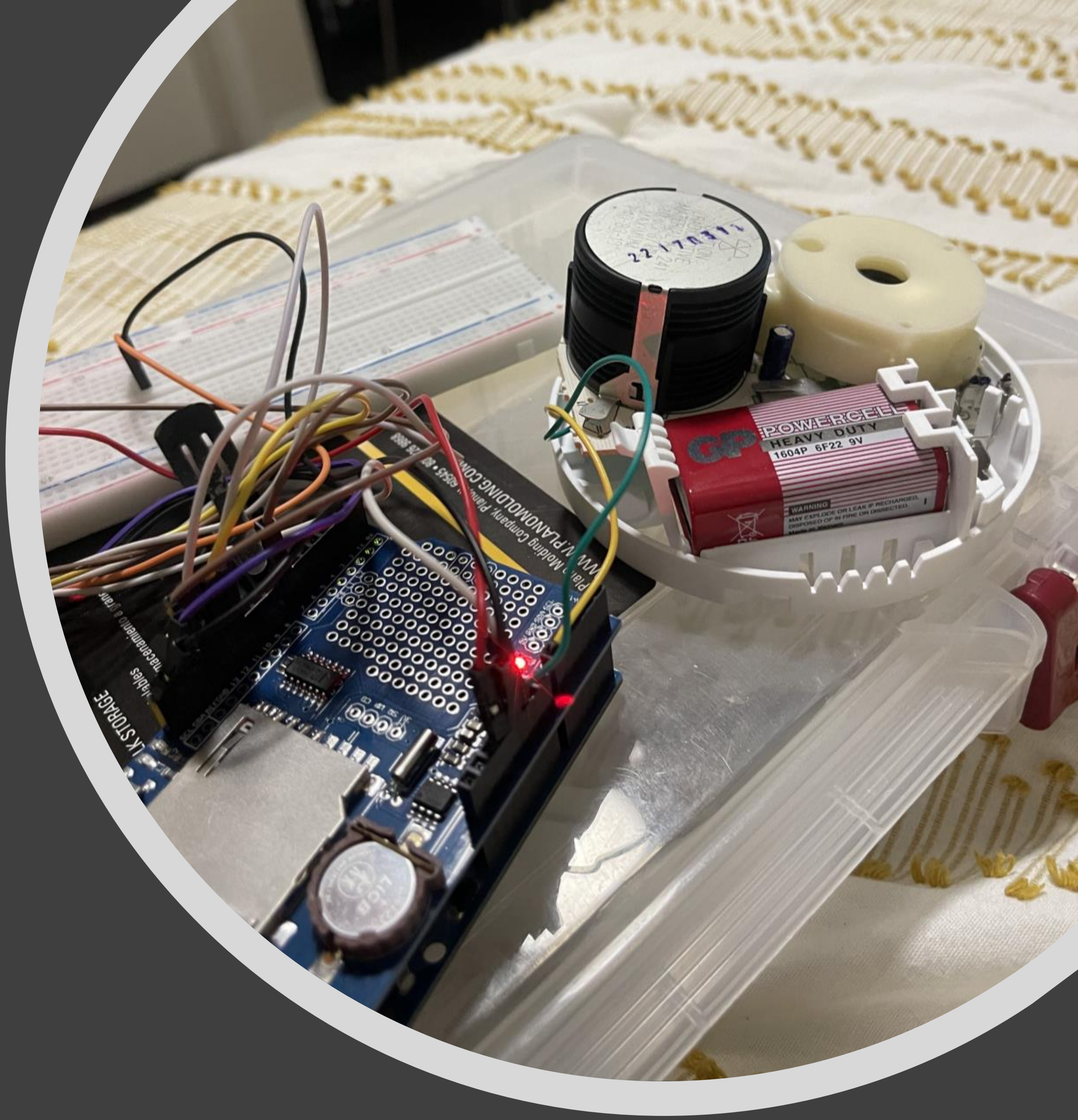
- Arduino MKR GSM 1400 processor is capable of connecting to the internet using cellemetry.
- Soracom SIM card used for connecting to the internet.
- Capable of GSM (2G) and GPRS (3G) cellemetry protocols.
- Uses Arduino GSM antenna for stronger connections.
- Processor establishes connection, gathers data, sends the data to AWS, disconnects, and repeats every 5 minutes.
- Processor produces current spikes when establishing a connection.
- Requires a 3.7V LiPo battery to provide extra current during the spikes or the connection will fail.

Implementation – GSM / GPRS + Temp / Humi

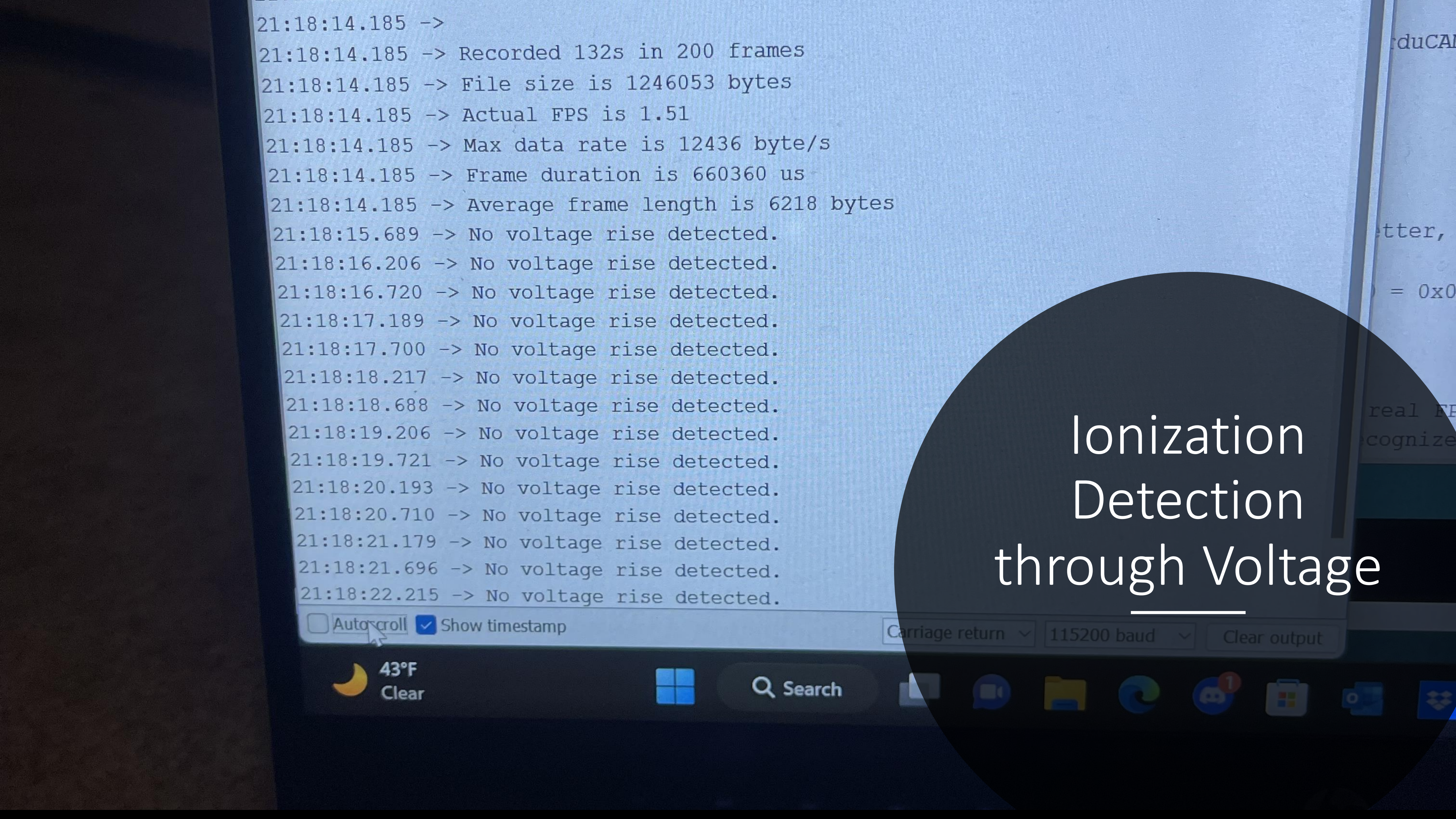


Implementation – Ionization Sensor

- The ionization sensor is covered by a black silicone cover.
- When it detects enough smoke, it'll trigger a voltage change which will then alert the camera that there is a possible wildfire near and start recording a video.
- The voltage change will be higher than >4.60V and will reach <5.0V. If no voltage rise is detected, then it'll display "no voltage rise detected."
- Future plans: connect ionization to power supply and remove unnecessary buzzer.



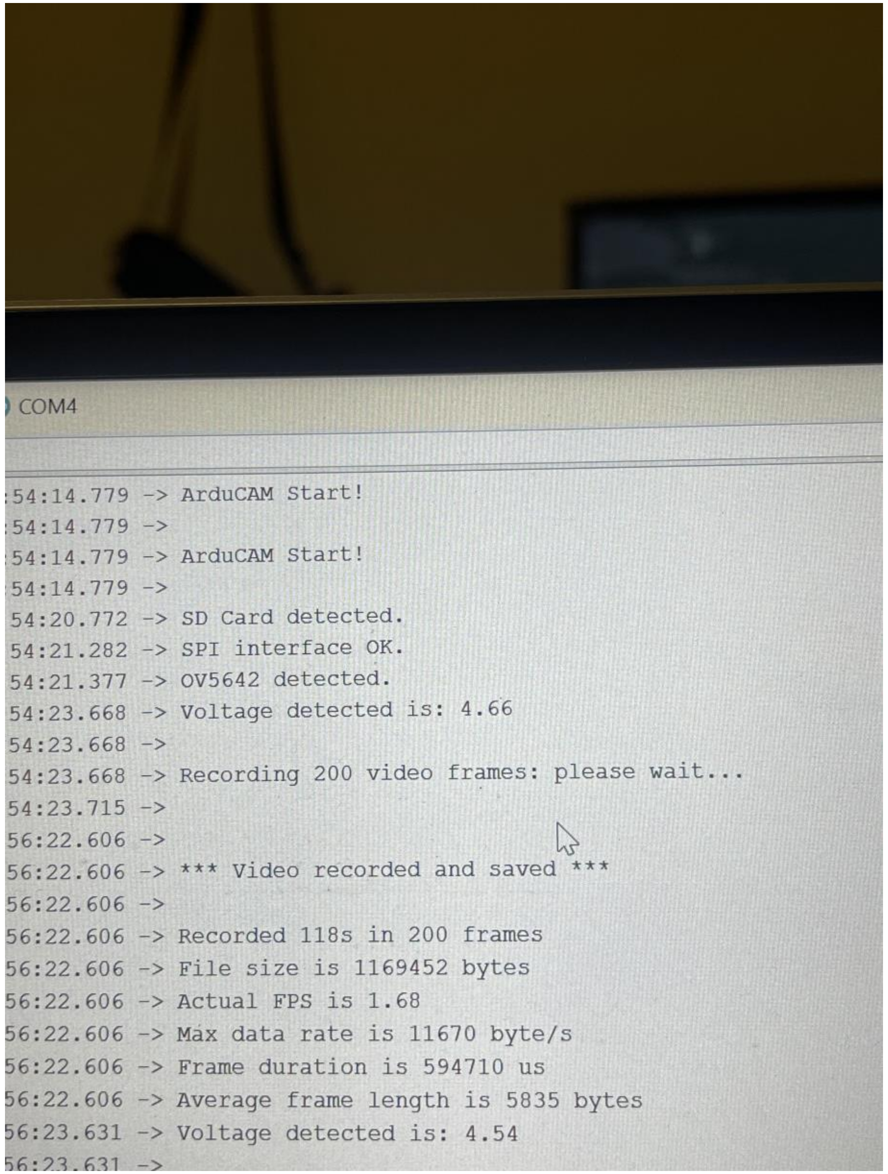
Ionization Detection through Voltage



Implementation - Camera

- Arducam 5MP mini camera is used to record a video of a possible wildfire near.
- Connection: Camera -> SD Card Shield -> Arduino Uno board
- Using the ArduCAM code provided, modifications were made to make sure it detected the model (OV5642) and adding the ionization sensor code into the void loop() section.
- When an increase of voltage is detected, the camera will then start recording a video of 200 frames. Once the voltage goes back to its constant, it'll stop recording and save video on SD card.
- Future plans: send video to the cloud and connect a white LED for night vision.

Camera and Sensor working together



Preliminary Results

- We connected each other's code to allow the hardware device to use multiple sensors and collect data, which is then stored in the Soracom database.
- We linked the website to the Soracom database using an API to fetch data.
- We configured the camera to detect fires and save the images it captures to an SD card.
- We implemented a photoelectric sensor and made progress on using an ionization sensor from a smoke alarm product.

Schedule

- Requirements: Completed on 10/01/2022
- Specifications: Completed on 10/08/2022
- Architecture Design: Completed on 10/27/2022
- Component Design: Forecast completion is 03/01/2023
  - Temperature, humidity, photoelectric, ionization, camera, GPS, GSM, and database completed on 12/16/2022
  - PSU, solar panels, enclosure, wind sensors, cooling system, and PCB design will be completed by 03/01/2023
- Integration: Forecast completion is 04/27/2023

Conclusion

- Approximately 50% of requirements completed.
- Project to be completed by 04/27/2023.
- Have implemented:
  - Temp/humi sensor, camera, photoelectric sensor, ionization sensor, GPS, GSM/GPRS connection, database, and website.
- To implement:
  - Wind vane, anemometer, enclosure, solar panels, rechargeable battery, troubleshooting, cooling system, PCB design, and 3 completed units.

Acknowledgements

- Our professor, Dr. Robin Pottathuparambil.
- Our sponsor, Fil J. Cosentino.

References

- https://www.epa.gov/climate-indicators/climate-change-indicators-wildfires
- https://lastminuteengineers.com/neo6m-gps-arduino-tutorial/
- https://www.u-blox.com/en/technologies/cellular-network-fingerprinting-celllocate
- https://www.nfpa.org/Public-Education/Staying-safe/Safety-equipment/Smoke-alarms/Ionization-vs-photoelectric
- https://electronicsinnovation.com/v2-0-temperature-data-record-on-aws-iot-core-with-esp32-using-arduino-ide-and-mqtt-protocol/
- https://create.arduino.cc/projecthub/andreiflorian/smartagro-e46213?ref=part&ref_id=50608&offset=3
- https://create.arduino.cc/projecthub/andreiflorian/smartagro-e46213?ref=part&ref_id=50608&offset=3
- https://content.u-blox.com/sites/default/files/AT-CommandsExamples_AppNote_%28UBX-13001820%29.pdf
- https://support.soracom.io/hc/en-us/articles/218427317-What-is-an-APN-setting-and-how-do-you-set-it-up-
- https://create.arduino.cc/projecthub/Arduino_Genuino/securely-connecting-an-arduino-mkr-wifi-1010-to-aws-iot-core-a9f365

References

- https://www.elithecomputerguy.com/2020/06/arduino-arducam-motion-activated-video-spycam/
- https://www.arduino.cc/en/Tutorial/BuiltInExamples/ReadAnalogVoltage
- https://github.com/ArduCAM/Arduino/tree/master/ArduCAM
- https://www.microchip.com/en-us/product/RE46C120