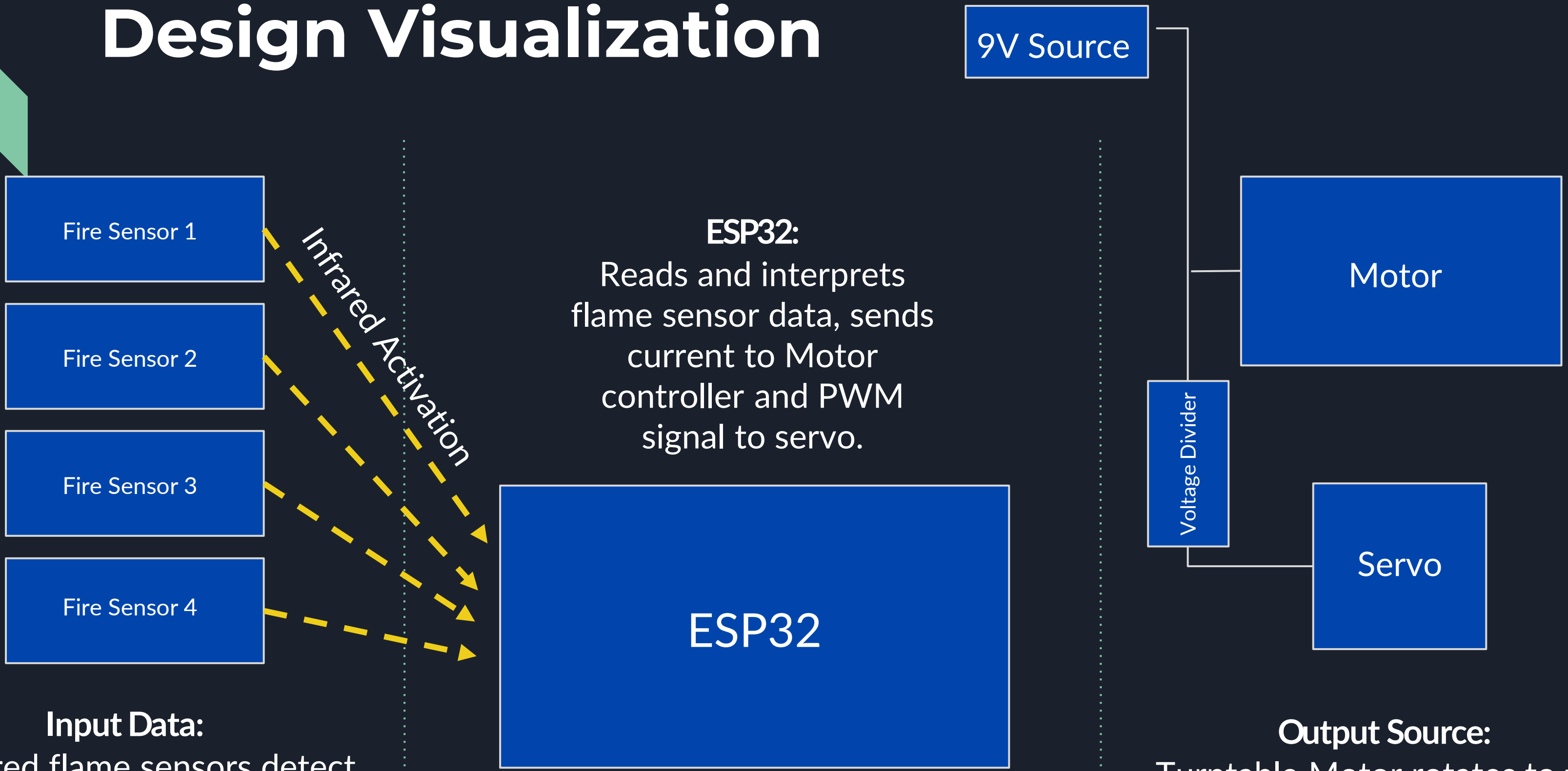
Automatic Fire Suppression System

Kilian Cox and Aashrith Beesabathuni

Design Visualization



ESP32:
Reads and interprets flame sensor data, sends current to Motor controller and PWM signal to servo.

Input Data:
4 Infrared flame sensors detect unwanted flame, delivering position to ESP32

Output Source:
Turntable Motor rotates to the correct burner position, trigger servo activates once target position is reached.

Computation: Results and Issues

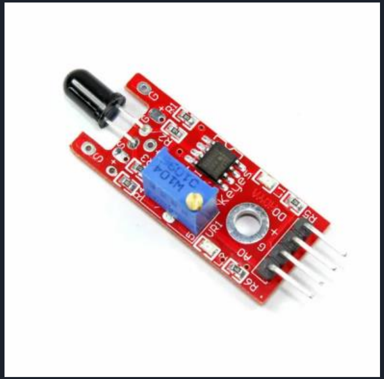
- All of our code was written in micropython, and uploaded through the Thonny software
- Computationally: the system has 3 main functions that are called whenever needed through the program loop:
    - This allowed us to easily call functions later down the line, especially in a program that is very similar for each stovetop, with the distance being changed.
    - Some issues: PWM with the servo, as we had difficulties actuating it, and more importantly, getting it to stop actuating.

```python
def turnOneQuad():
    SpinPin1.value(1)
    SpinPin2.value(0)
    sleep(0.6)
    SpinPin1.value(0)
def shoot():
    servo_pwm.freq(10)
    servo_pwm.duty(10)
    time.sleep(1)
    servo_pwm.duty(1000)
def message(stove):
    feedName = "aashubee/feeds/aashubee"
    testMessage = "Fire Detected at StoveTop " + str(stove)
    mqtt.publish (feedName,testMessage)
    time.sleep(1)
    mqtt.publish(feedName, "Phew! fire extinguished!")
```

Sensing: Main Decisions & Results

Main Sensor: KY-026 Fire Sensor

- This sensor works by detecting wavelengths found in fires, and outputting a HIGH or LOW signal based on whether the fire was detected
  - Sensor ha built in potentiometer, allowing us to turn the “sensitivity” on what constitutes high and what constitutes low
- 4 Sensors placed on 4 different stove burners, each connected to the pins of the ESP32

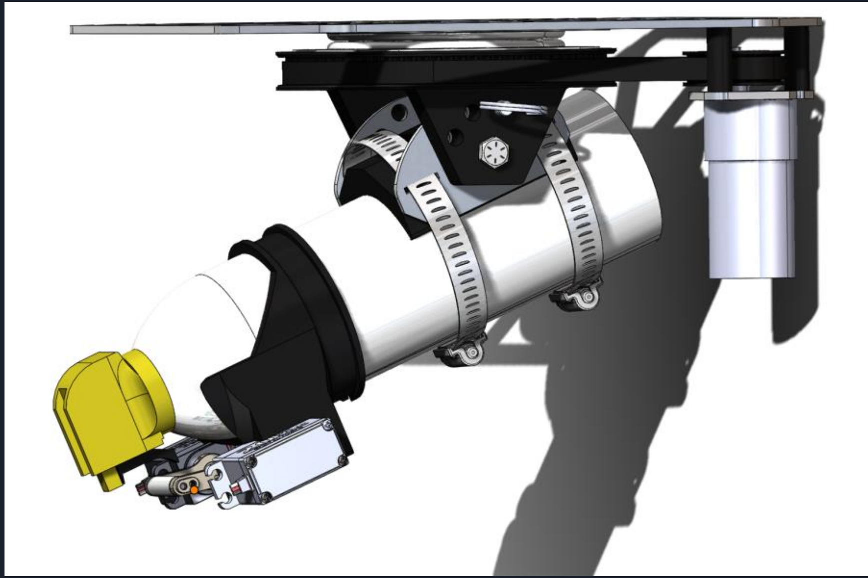


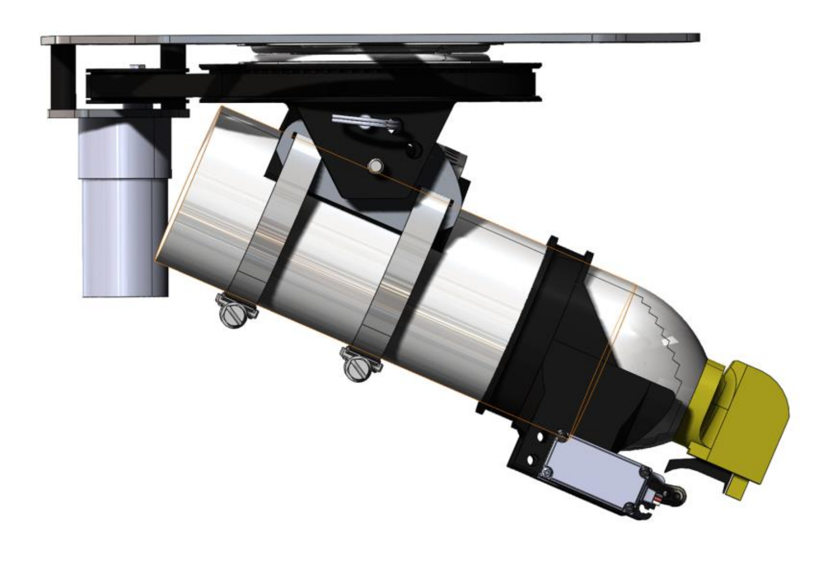
Actuation: Main Decisions

- **Trigger servos** and **turntable motor** are main sources of actuation.
    - 251 rpm motor from kit was deemed feasible enough to work with an **18:80 pulley reduction**
    - The MS24 24KG servo was chosen given it’s **high torque capabilities**, while still being a **relatively light** servo option.
- All design parts were waterjet and 3D printed to achieve the needed geometries.

Actuation: Design

All mechanical parts were **fully caded** in order to achieve dimensional accuracy and a **working assembly**



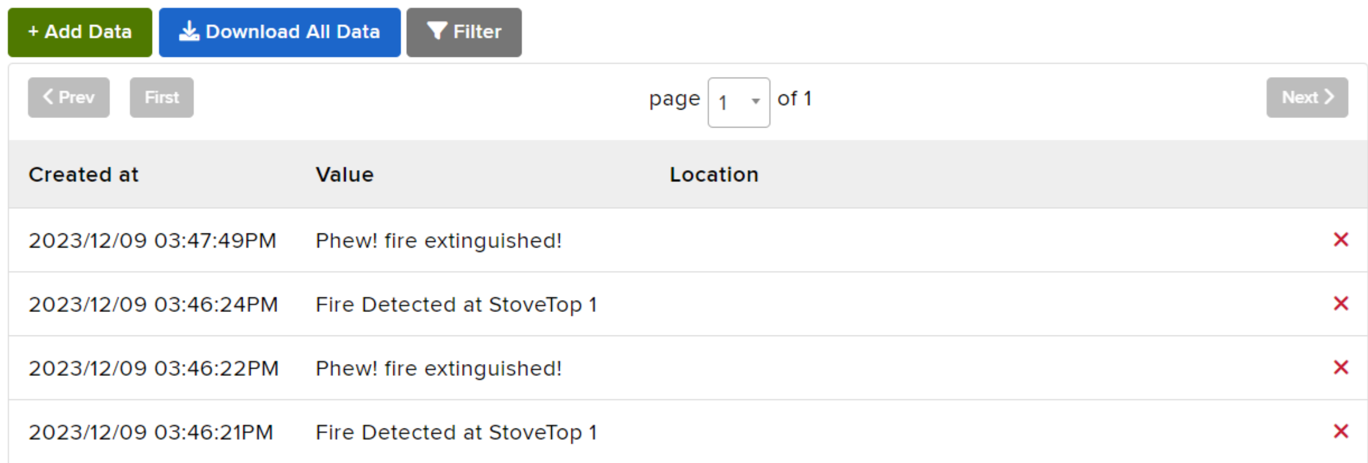


Actuation: Results and Issues

- Overall motor control worked very well, allowing the turntable to achieve any desired location.
  - Time based quadrant selection was our final method of rotation control, as we weren’t able to achieve the encoder-based control from this specific motor.
  - In the future we would probably implement a different motor that allowed encoder control or a separate encoder mounted to the turntable itself.
- Servo control worked less effectively than the motor, but we we’re still able to get the servo to rotate and press the trigger on the extinguisher.
  - Issues with PWM signals and position control were our main issues, preventing us from being able to accurately rotate the servo to the correct position.
  - Mounting and logistical problems also prevented the design from reaching its maximum mechanical potential.

IOT Implementation

- Using MQTT and Adafruit.io, we were able to utilize IOT principles and communicate with the computers
- Allows for quick and easy communication, transferring info whenever fire is detected by the system
- Also alerts users when the fire is put out on the stovetop



Final Conclusions

Demo: Due to safety and cleanliness concerns, we decided to not actuate the servo resulting in the can spraying

Instead, we will light a flame to determine where the motor has to spin to, as a way of showcasing all minimum aspects of the project