Allen E. Paulson College of Engineering and Computing

Department of Electrical and Computer Engineering

EENG 4640 – EE/CpE Capstone Senior Project

Spring 2021

Autonomous FireFighting Drone

John Bailey

Maria Gonzalez Bocanegra

Camiya Felton

Jamison Golson

Table of Contents

Abstract

The Autonomous Firefighting Drone seeks to help combat and extinguish fires that commonly ravage homes and forests around the world. Spontaneous fires are far too common disasters with a lack of revolutionary solutions. UAV drones are rapidly becoming more and more common from hobbyist flying to military assignments. Through the use of UAV technology the team looks to achieve a solution that will be able to extinguish a fire along with preventative measures. In order to extinguish a fire autonomously, the drone will be able to see the fire through the use of a camera before then completing an automated mission to maneuver to the location before activating a balloon release system to put out the fire. The project was created by using Raspberry pi 3b, PI cam2, 16-channel PWM servo driver, VL53l0X range finder, a gps, radio, and Pixhawk flight controller. Through the use of these components the UAV will be able to safely navigate an area and extinguish the fire.

1. Introduction

The aim of this literature review is to identify and explore previous projects that have previously implemented the use of Unmanned Aerial Vehicles for firefighting. This review investigates the capabilities of using Unmanned Aerial Vehicles to rapidly address the complex problem of rapidly increasing fires. Furthermore, this project seeks to pair the Unmanned Aerial Vehicles with a video streaming camera and artificial intelligence to help automate the fire detection process. The implementation of artificial intelligence, and more specifically Computer Vision, will allow for faster fire detection rates as well as to provide automation. In order to have a well-rounded understanding of how to approach this problem.

1.1 Problem Statement

The purpose of this project is to research, design, and fabricate an autonomous drone that will aid firefighters when there is an emergency. The use of drones or other UAVs have been on the rise and this technology could be used to help against an issue that is always prevalent. This project will create a drone that will be able to seek out fires and possible fire hazards that can help alleviate the amount of volunteers that would be put at risk. The drone uses a combination of different sensors in order to simulate a flight controller, an accelerometer, gyrometer, barometer, and distance sensors. Our goal at the beginning of the project was to create and code our own flight controller by using a combination of these sensors but due to time constraints we decided to utilize a Pixhawk flight controller to complete the task. Currently firefighters are at considerable risk whether it is physical or mental stress and pain. Firefighters are exposed to burns, smoke inhalation, and are at risk to falling debris. Issues from fighting fires can also bleed to the home-life in the form of heart disease, lung damage and cancer.

With the added help of drones we will be able to reduce the risk that people are placed into while fighting off infernos. Drones are becoming more and more common through surveillance, observation and different types of applications. Drones will be able to reduce the amount of human error while assessing a situation and reduce the response time needed to solve the problem. Drones will also allow the user to focus on other imperative tasks that might require a human's concentration and more immediate thinking by anyone in the area.

As we research the project and complete the project we will find the best solution to create a safe and affordable autonomous drone. Fires are a constant threat to civilization and in recent years the amount of fires that have occurred has increased. There are 358,300 home based fires on average per year and there were 499,000 structure fires for the year of 2017. This type of tragedy causes the death of civilians every two hours resulting in 2,620 deaths a year. When wildfires are taken into account there were 1,291,000 fires in the U.S which resulted in 16,600 injuries and 3,704 deaths for the year of 2019. Fires also contributed to $14.8 billion lost in 2019. With the sheer amount of people that are affected by fire fires along with global warming on the rise the risk of fires becoming more prevalent, being able to use drone technology to help combat this issue would be an ideal situation.

1.2 Project Limitations

During the course of completing this project there were a great deal of hindrances encountered every week for the project. Funding was the primary issue when completing the project due to the fact that up-to-date materials are required for the project to run smoothly. The materials that were either provided to us or some of our own proved to not work for our project specifically and this caused us to have to order more parts constantly. Due to the lack of funds towards the start date of the project critical tasks such as motor testing and flight stability were pushed back until the proper equipment were delivered to us.

2. Project background and Literature review

The first feature this literature review is focusing on is the use of Unmanned Aerial Vehicles for fire fighting and extinguishing. The implementation of the UAVs comme switch several perks to the field. For example, the risk of human lives to combat fire can be drastically reduced due to remote control of the UAV itself. As discussed in [3], the UAVs are implemented around the perimeter of the fire. Since the fire has an undetermined flight path, this paper explores the use of infrared sensors to collect images of regions below the drone. The infrared images are then used to command the drone and follow the path of the fire. Some of the most important aspects this paper touches upon is the communication range of the UAVs. It is primordial to have proper connectivity with the drone to maintain flight control. In the case of the project developed in this proposal, an RFID scanner will be used for the drone to return to home base once a certain limit of connectivity has been reached. Furthermore, the team will implement a Pixie camera and a thermo sensor to detect the fire and the path it has taken. The team is also exploring computer vision techniques such as Convolutional Neural Networks to detect fire through the Pixie camera and allow for UAV automation.

[4] is another paper that explores the use of Convolutional Neural Networks and UAVs for natural disaster assessment. This research leverages the use of a Matrice 300 RTK as their primary drone to conduct assessments of natural disaster-damaged roads. The research focuses mainly on classifying among six different types of damages on roads. However, the scope of this paper can be expanded to include fire hazard detection. With respect to this paper, this proposal seeks to enhance and build upon their results. One key improvement the team seeks to achieve is to conduct real-time fire detection (classification) in contrast to sending the video feed to ground-station and classifying after the fact. With real-time fire detection in mind, the project also seeks to incorporate drone autonomy. To achieve this goal, and as mentioned before, the drone will be incorporated with several thermo sensors as well as computer vision navigation, in contrast to supervised navigation in [4].

In [cite3], the paper dives into the design of the UAVs and features that have to be taken into account so as to create a successful model. [5] poses an argument for frame materials that can withstand the heat expelled by fire, the fire extinguisher payload (2.5 lbs.), and sustain proper speeds. Additionally, the team believes that wind speed should also factor into the design considerations. [5] also considers the structure configuration of the UAV, suggesting a hexacopter. The hexacopter design allows for system redundancy in the case that a motor/propeller fails. It also allows for better payload management. The team is interested in pursuing this design avenue. Another design option kept in mind is the quadcopter design. This design is heavily used by the industry and is an option the team is willing to pursue due to the mechanical simplicity and cost efficiency. Additionally, other key points [5] touches upon is range communication and collision avoidance. As mentioned before, this project will seek to address these features with an RFID scanner and computer vision algorithms, respectively.

2.1 Sensors and Software in UAVs

With the advancement of technology and the field of robotics the amount of sensors and softwares that allow an autonomous vehicle to sense the world around itself have skyrocketed. The culmination of these sensors allow the vehicle to act independently of a driver/pilot while still being completely receptive to its environment with a rapid response time. The large majority of sensors used on the drone were to work in collaboration with each other in order to create a flight controller. The sensors use would be able to measure the changes of varying dimensions:

- ❑ Altitude (Barometer)
- ❑ Acceleration (Accelerometer)
- ❑ Change in Orientation (Gyrometer)
- ❑ Temperature (Thermocouple)
- ❑ Magnetic Fields (Magnetometer)

Being able to extract the usable date from the sensor values allows us to make a working flight controller. A time of flight of the sensor is another important component for an autonomous device because it allows the UAV to see its surroundings. In order to make the drone for the people and the environment around it the range finder can be used as a distance sensor so that it can always keep a safe distance away from anything that can be injured while also keeping the drone out of harm’s way. The use of these sensors requires fast response time for the most accurate date acquisition so that the UAV can respond accordingly to the corresponding inputs.

2.2 Fire extinguishing Methods

The abundance of different types of fires and how they start change how they can safely be extinguished. The location of the fire also greatly affects what type of fire suppressing method. The common house fire can be put out with a water sprinkler system with little to no issue but when there are other classes of fires present the methods to subdue the flame change.

There are five types of classes a fire can burn from. Class A fires are the most commonly found fires because they form combustible materials like wood, cloth, paper, and plastics. Class B fires are formed from flammable liquids such as tars, oils, paints, solvents, lacquers, alcohols, and gasses. Class C fires stem from electrical sources and equipment. Combustible metals fall under the category of class D fires. The more common metals in this class are magnesium, titanium, sodium, lithium, and potassium. The final class of fires, class K, are fires that are started by cooking appliances and materials used for cooking. This could be the flammable materials caused by animal fats and materials.

There are multiple ways that these types of fires can be extinguished. Water is the primary method of suppression and is by far the most common. One issue with a water based fire extinguisher is the issues with use in freezing environments. In order to remedy this situation the extinguishers are combined with antifreeze in order to prevent freezing. Water based extinguisher utilizes a water mist to suppress fires in lieu of a stream to reduce damage done to surrounding objects. Water is used for class A fires, a film-foaming suppressant are used for both class A and B. This method has a foam output that is able to lay on the surface of a liquid which helps prevent the fire from restarting. Carbon dioxide has the advantage of not leaving residue after being used. Due to the lack of residue from the discharge it is used where one does not want to damage components such as a computer. Halon based fire extinguishers are ideal for cold environments but are rarely used due to their negative environmental impact. Dry chemicals and wet chemicals are the next types of fire suppression used. Dry chemicals use moisture absorption to help reignition. Dry powder extinguishers are used for class D and flammable metals..

3.0 Project Description

3.1 Project Goal

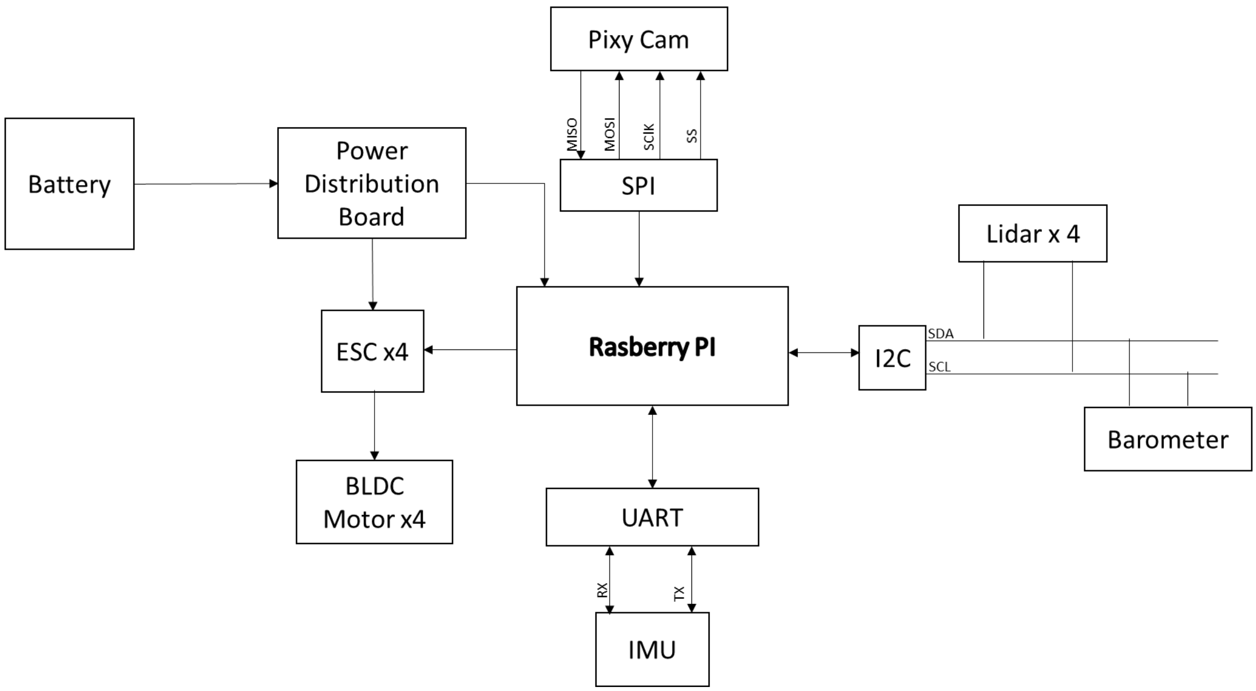
The goal for the project since its inception was to create a firefighting drone that will be able to work autonomously. In order for the drone to complete the mission, the drone will now use a grenade release system to suppress the flames.

Methodology and Project Description

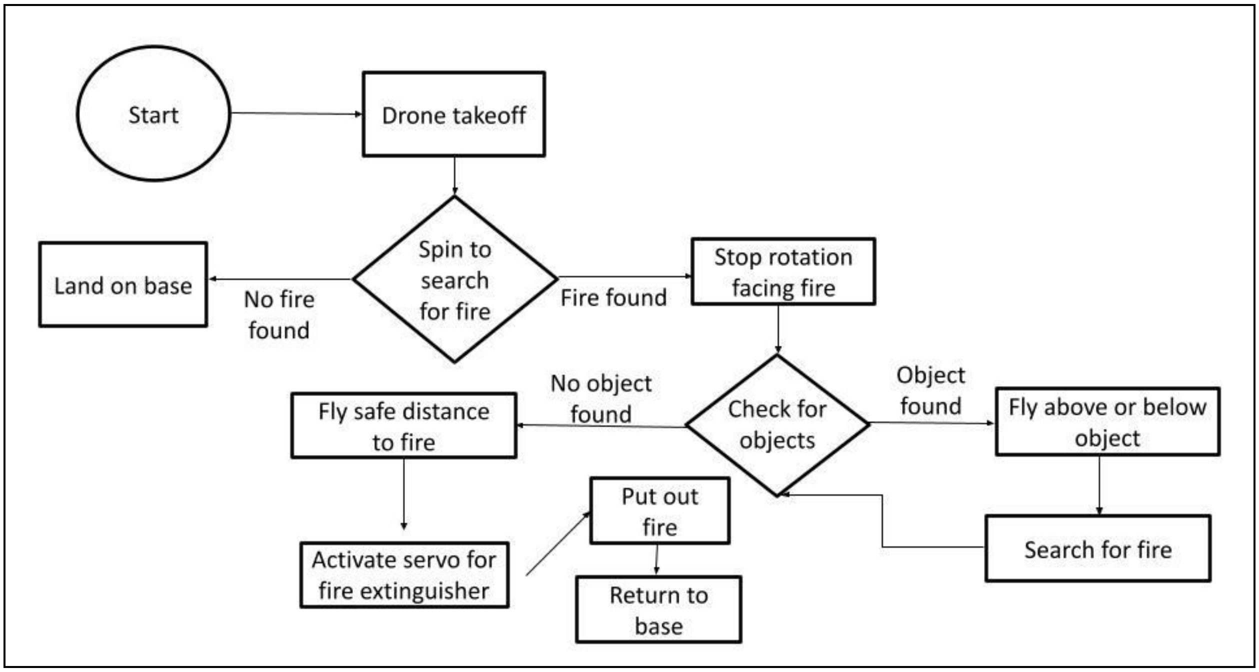
This research project will explain the benefits of creating an autonomous drone that will be able to seek out fires and extinguish them appropriately. The drone will utilize cameras and thermal sensors in order to replicate how an infrared camera would operate. We aim to prove the effectiveness of an autonomous drone by measuring the time it takes to locate the problem area and extinguish the flame. We will also be able to calculate the efficiency of the drone by recording the amount stored materials vs the amount of materials used during each trial.

In order to fabricate an autonomous drone it will be completely designed and programmed for all aspects of the UAV. The drone is designed in autocad Inventor before going through a prototype phase. The Drone will be a quadcopter with a x-style frame and DC brushless motors with topside propellers. In order for the UAV to recognize the environment it will be fitted with Time-of-flight sensors, and a camera. The sensors will allow the drone to have an awareness of the environment around and be able to react accordingly with little human input. The Time-of-Flight sensors will be used to tell whenever an object is in its vicinity or along its flight path and stop the drone from moving. To help flight control of the system a PID controller will be implemented using a gyrometer, barometer, magnetometer, and accelerometer. These sensors will help measure where the drone is in the world. The barometer measures the pressure around the drone in order to calculate the altitude, and the accelerometer will measure the speed of the drone. The gyrometer will record the orientation of the drone, and the magnetometer will measure where the drone is in relation to the magnetic fields. For the drone to extinguish fires, we will be using a spray canister. The spray canister discharges four times longer than traditional fire extinguisher products. It is safer and more effective than water as water has proven to make some fires worse and cause them to spread. The extinguishing formula found in the fire spray can extinguish grease, paper, fabric, wood, or electrical fires and it is biodegradable and easy to clean. It is compact and easy to use and so will be fitted in the drone where it will be easily replaceable for refilling. To operate the extinguisher a servo will be connected that will be activated when the drone is in range of a fire. There will be a pin attached to the servo that when activated pops the balloon. As mentioned before, the drone should be able to operate with very little human input. Someone should be able to turn on the drone and when the time comes, put a new "grenade" into the drone a safe distance away from the fire; the drone will take care of everything else accordingly. This process should go on until the fire is completely extinguished.

3.1 Project Block Diagram

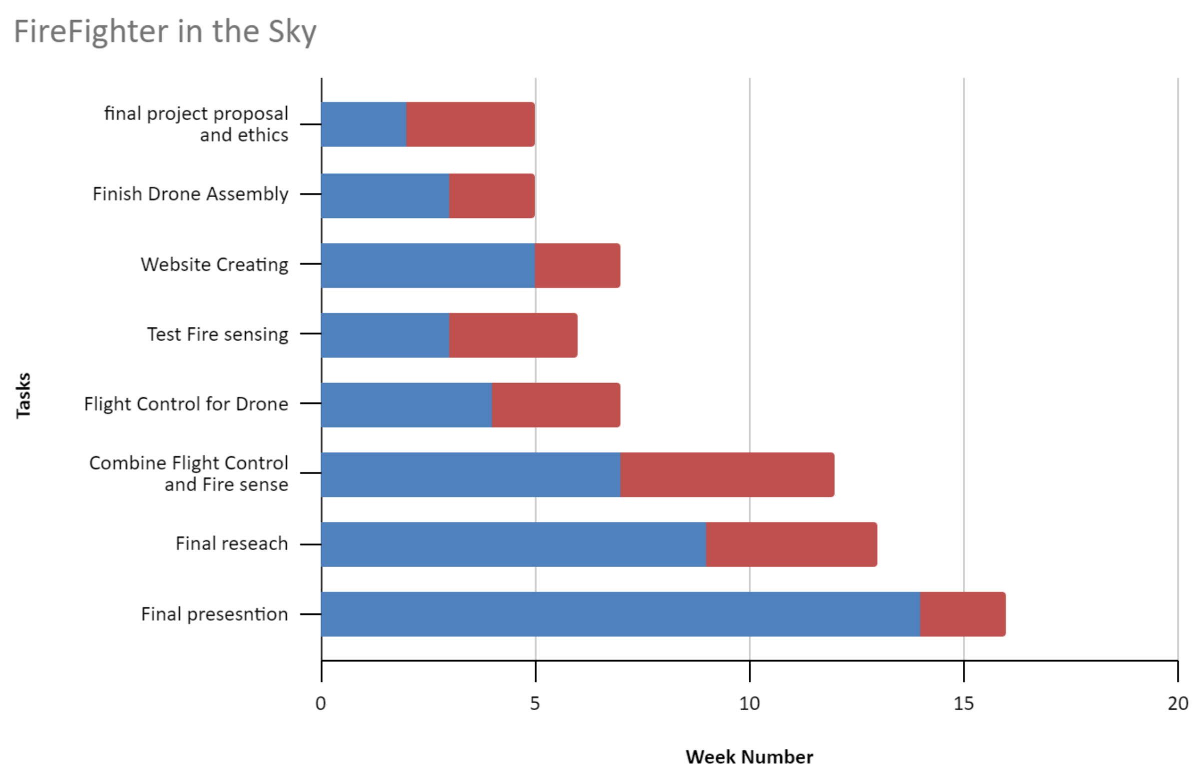


Figure


Start
Drone takeoff
Spin to search for fire
Land on base
No fire found
Fire found
Stop rotation facing fire
Check for objects
No object found
Object found
Fly safe distance to fire
Fly above or below object
Put out fire
Activate servo for fire extinguisher
Return to base
Search for fire

Figure

3.2 Project Scheduling



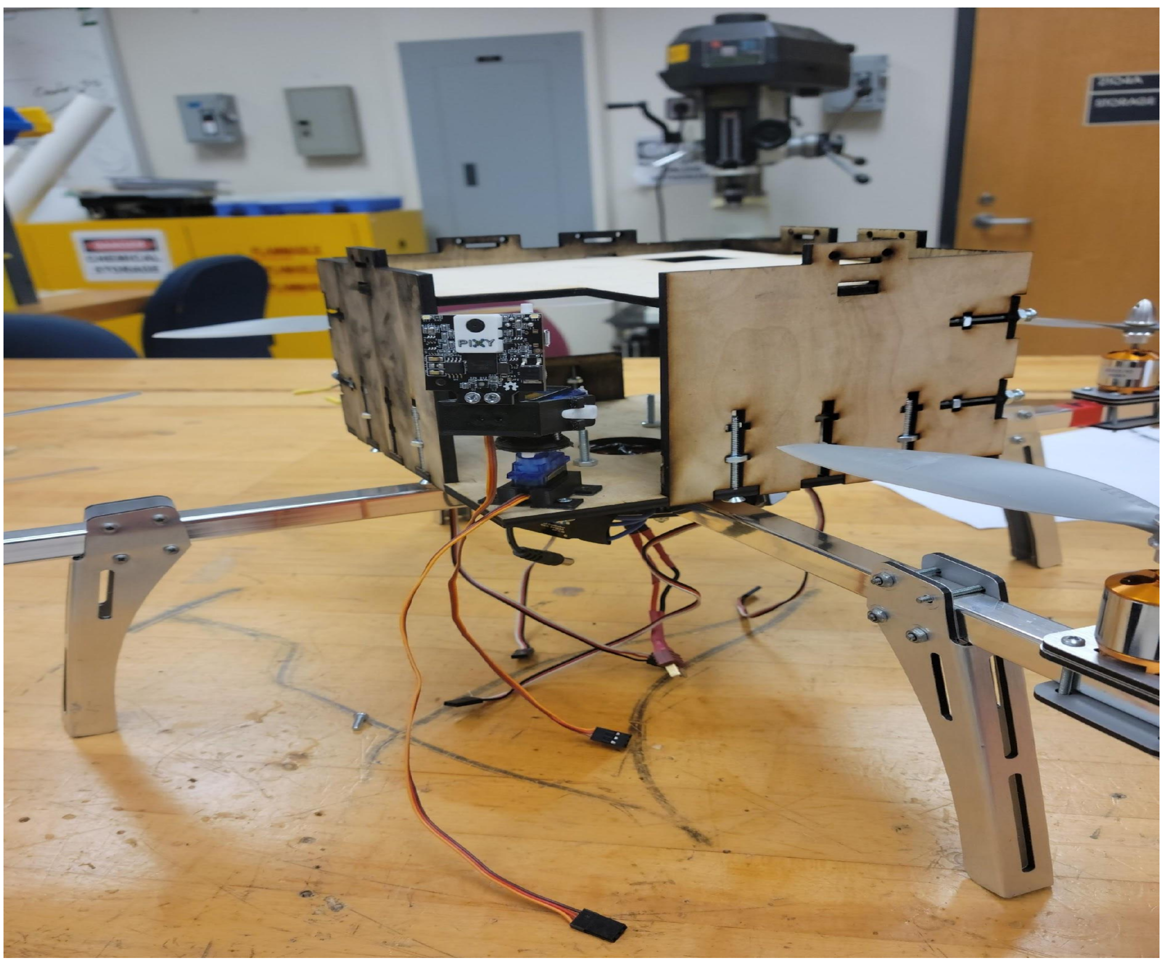
Figure

3.3 Drone Design

While brainstorming for the drone, there were a myriad of design changes for the construction of the UAV. The design changes were based around any limitations that were placed in the project or tasks that would take too long to implement in a short time frame. Besides the overall design of the drone, the electrical components used in the drone also varied.

The main goal for the project was to extinguish a fire during its autonomous mission. In order to do this task we need a body, flight controller, flight computer, a method of extinguishing, and a camera. The first design of the drone included the use of a miniature fire extinguisher connected to the undercarriage of the drone. The extinguisher would be parallel to the ground and be switch activated in order to subdue the fire. This idea was later changed to be a grenade release device. The “grenade” in this situation would be a water balloon for testing purposes. In future implementations this would be changed to a different solution. The grenade is popped from a servo once a set height is reached. Once popped the suppressant, in this design the water, would then put out the fire.

The overall body of the drone went through various design changes in order to fit the needs of fitting all of the components comfortably. The first iteration was a 3 tiered canopy that only allowed space for one microcontroller. We decided to step away from this drone as there was no space for the likes of a camera and a place to store the battery. The design after that included making the canopy one solid piece instead of having plates stacked on top of each other. This also would not allow for a level and vibration resistant frame while in flight as shown in figure .



Figure

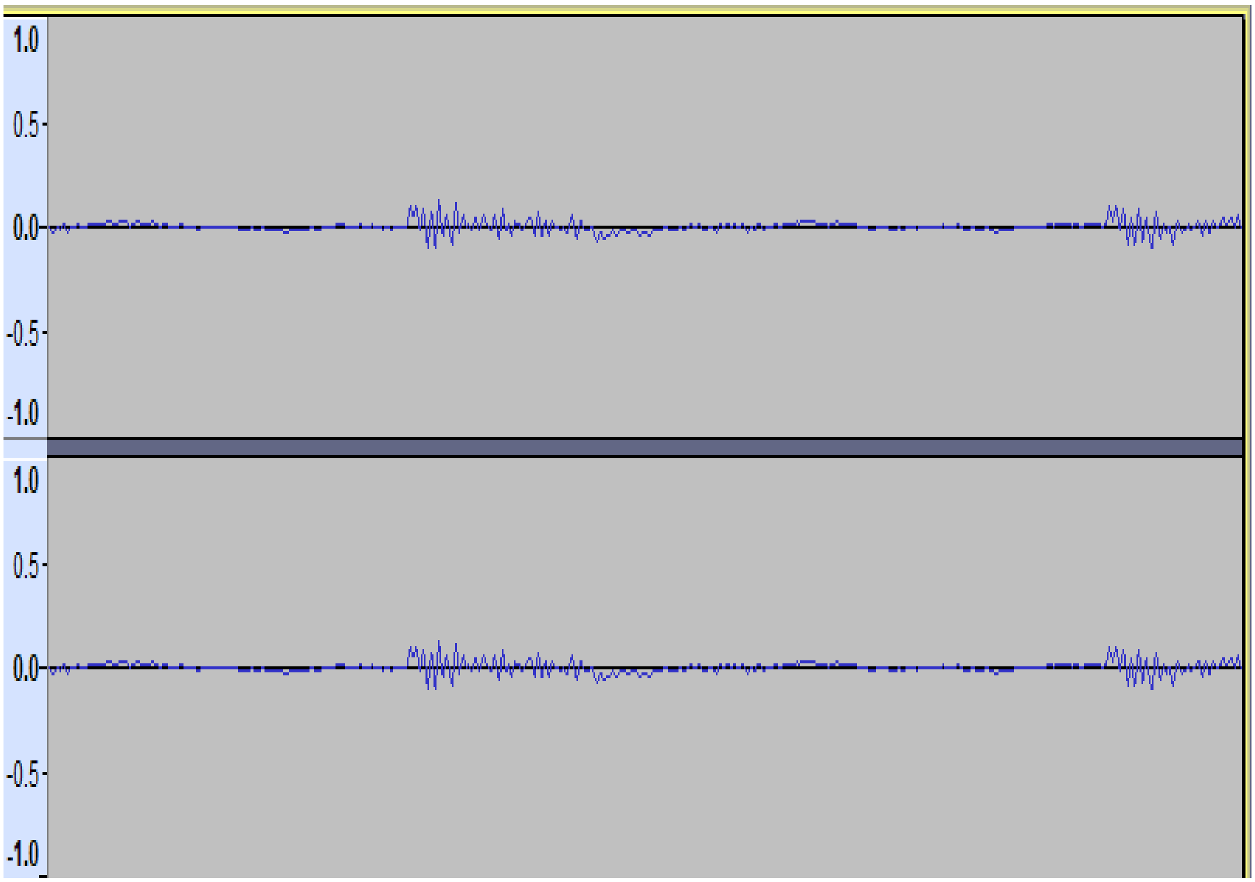
After running trials with this drone prototype the team encountered issues with the motors that were currently being used. The motors used were 950 KV brushless DC motors that were rated for a 2s-3s battery source. This presented issues with the motors not being able to produce enough thrust to carry the load of the drone. With a 3s battery source being 11.1 volts at max charge the 950 KV motors would have a max RPM of 10,545. If the motors were able to support a 4s battery, they would be able to spin at 13,680 RPM. At this desired RPM for each motor they would have over 1049g of thrust per motor using the formula in figure .

$$T = \frac{\pi}{4} D^2 \rho v \Delta v \qquad (2)$$

T=thrust [N]
D=propeller diameter [m]
v=velocity of air at the propeller [m/s]
Δv=velocity of air accelerated by propeller [m/s]
ρ = density of air [1.225 kg/m³]

Figure

The 1049g of thrust per motor would be enough to lift the drone which weighs in at 1909g but that would require the motors to be rated for a 4s battery. The team eventually switched to a set of 700 KV motors that were gifted to us but due to the age of the motors the motors did not turn on properly. Our goal was to get a set of motors with a lower KV rating because the purpose of the drone is to carry a load. High KV motors are used for racing drones and maneuverability, while a lower torque motor would generate more torque for the drone. The propellers chosen for the UAV also had an impact on the flight efficiency. We need propellers with a large diameter but a lower pitch such as the 10” x 4.7” we had equipped. The 10” diameter allows more air to move as the motors spin. The 4.7” pitch is how far the propeller would move forward in the air with each rotation of the motor. Having a pitch that would be larger than what is needed would result in an unnecessary current draw. The third set of motors used were 935 KV motors that were rated for a 4s battery. While this set of motors met all of the team's design requirements the wear and tear of the motor presented other issues for the project. Although the motors would produce enough lift in theory, in actuality the motors would spin at 50% efficiency.

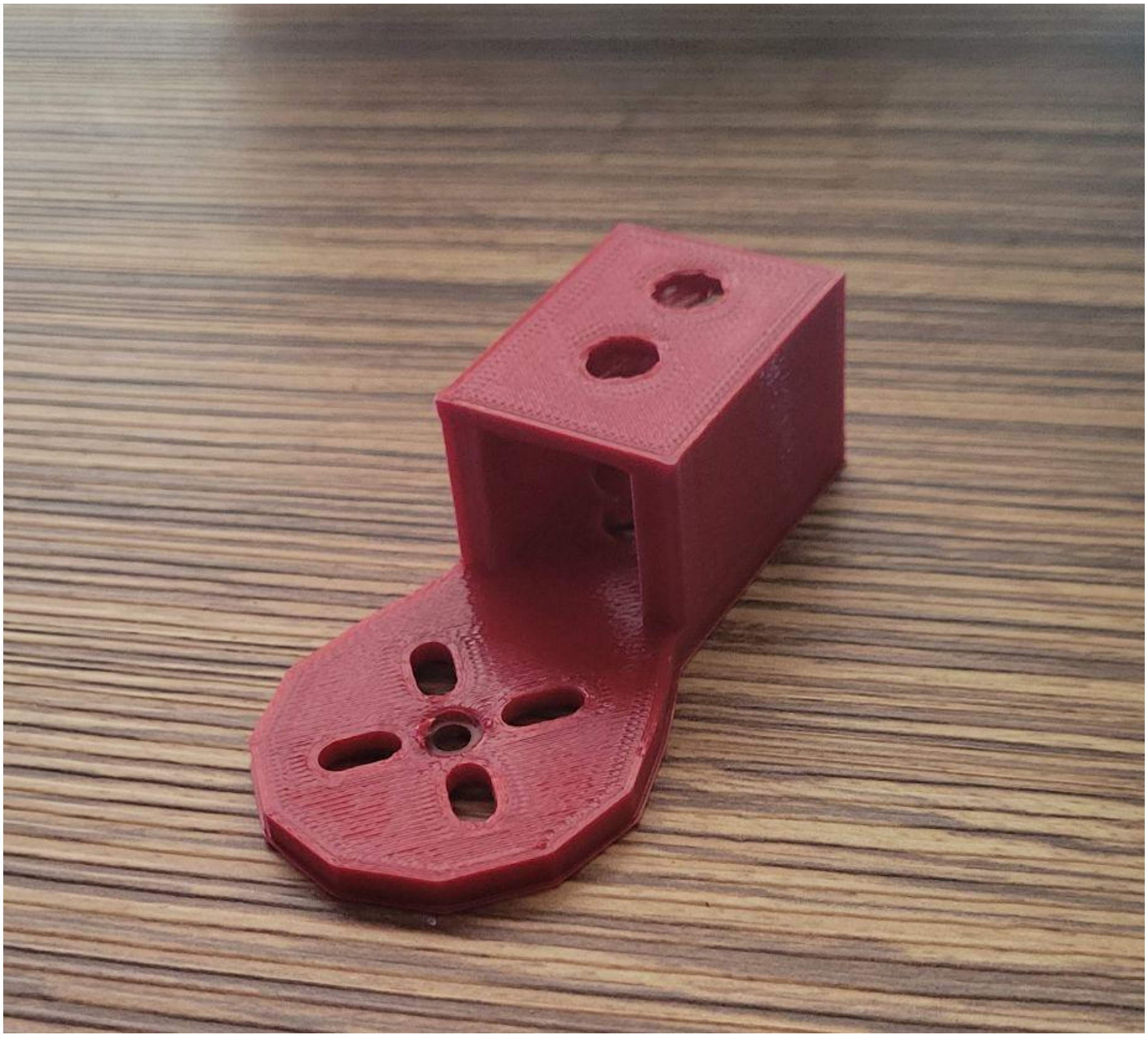


Figure

Using a sound wave form in figure we were able to find the real value for the rpm. By measuring the amount of pulses of the wave in a similar fashion to an encoder. Dividing the amount of pulses by the length of the waveform we are able to find the speed of the motor. The last set of motors we used were the 800 KV sunnysky DC brushless motors shown in figure . These motors are able to produce 2670g of thrust at max speed. Due to the amount of motor changes we needed to go through designing a motor brace to support each motor sufficiently was imperative. The motor brace went through a series of design changes to ensure that any stress that was placed on the brace would not cause it to shatter.

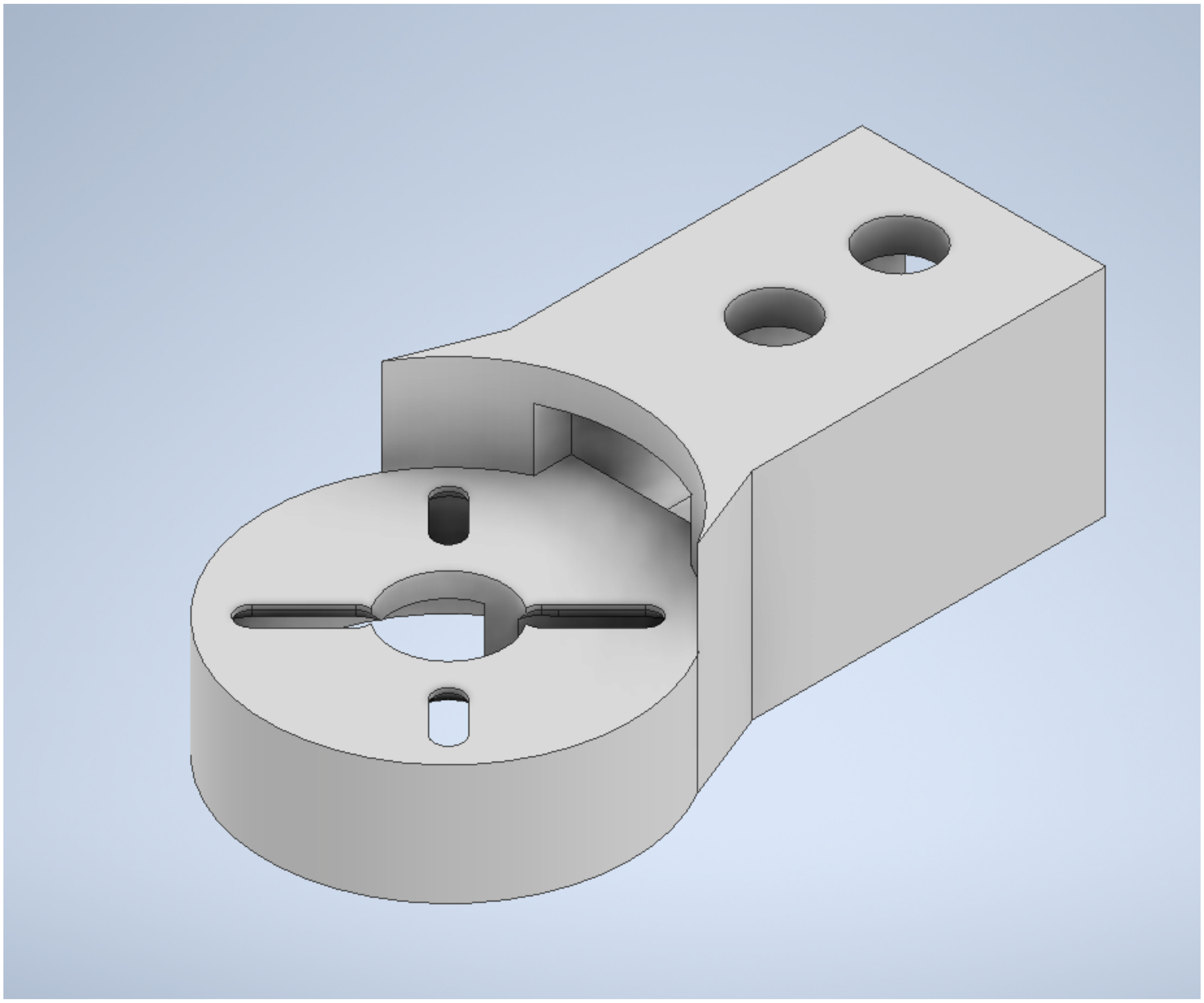
Figure

The newly designed brackets had to be made modular to fit most drone frames and motors of similar sizes. The final design for the motor bracket is shown in figure . The only material available to fabricate pieces was PLA 3D print filament limiting how strong of material that was available to use. The main issue presented by this material was if it would be able to take the stress employed by the motor rotation but still be light enough to allow lift off of the drone.The first design of this piece had a much thinner end piece that supported the motor. While testing the motor the thinner portion that holds the motors in place shattered leading to be a potential hazard for any bystanders. The piece shattered because the motor was properly secured to the brace causing it to vibrate freely putting too much stress on the system.



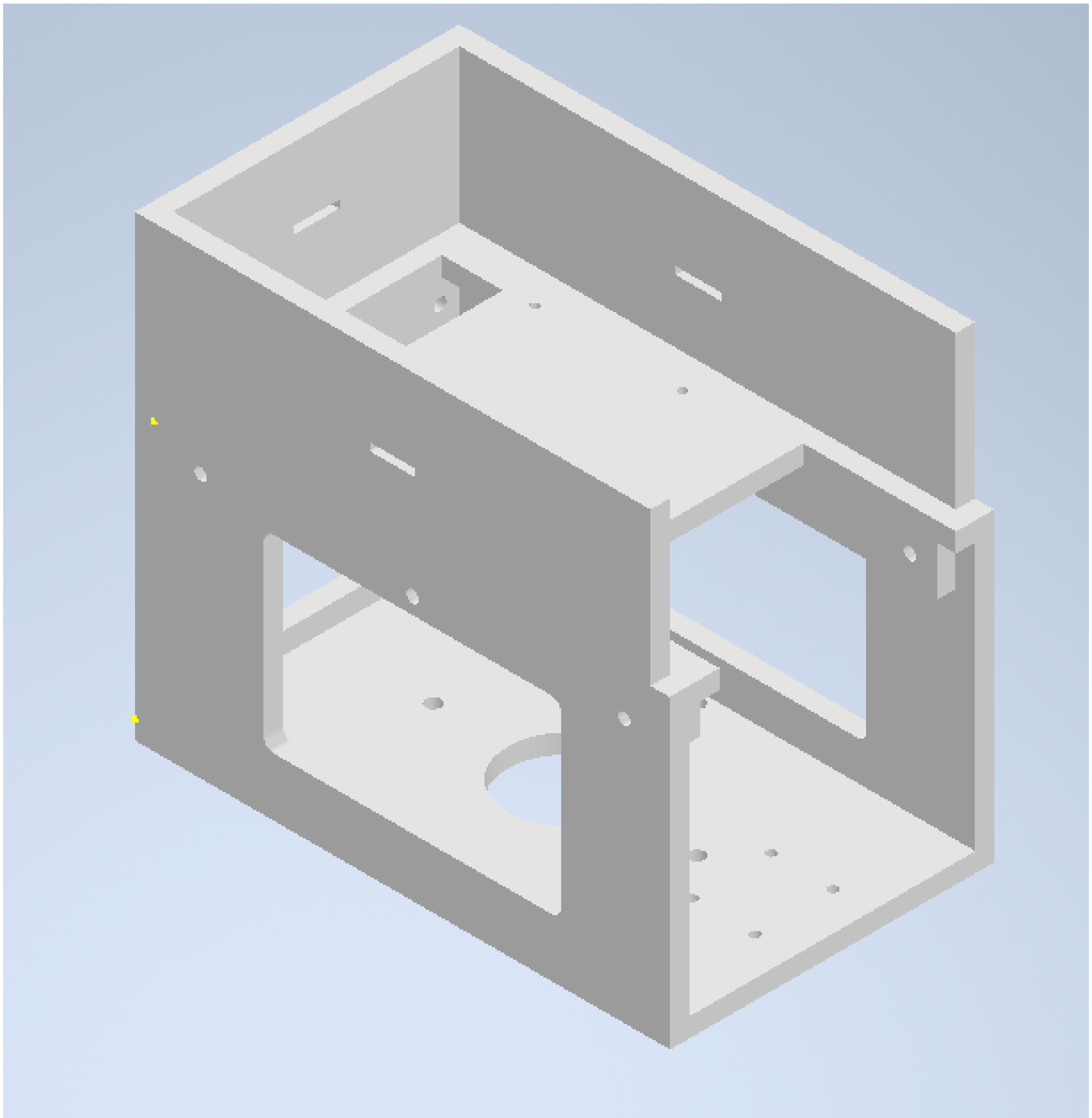
Figure

In order to remedy this problem a thicker bracket shown figure was designed to support the added mass of the new Sunnysky motors and to help stabilize the motor.



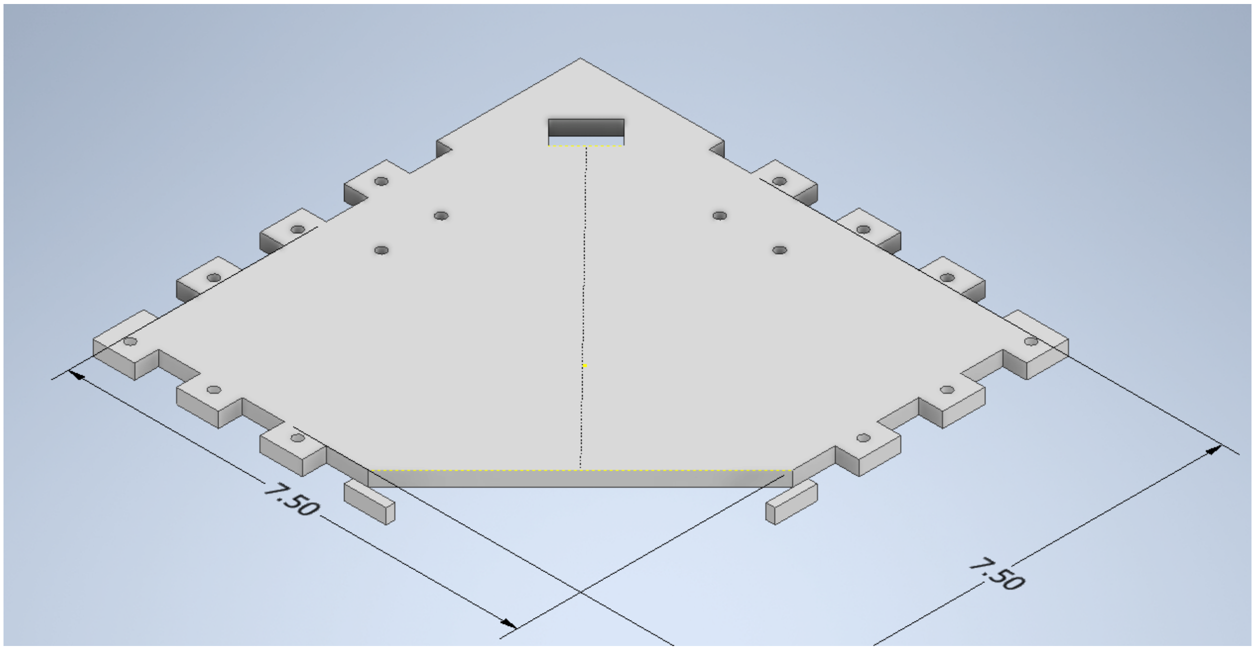
Figure

After handling the motor support issues the main cockpit was the next design challenge. In order to achieve a more aerodynamic frame a two piece cockpit was created to house all of the components shown in figure . The issues presented with this design was placement of the battery pack and the power distribution board. In this design the distribution board was exposed to the metal screws that would hold the cabin in place leading to a risk of the circuit short circuiting. When it comes to battery placement the main issue was having the battery centralized. With the weight of the battery being 544 grams having the battery off to the side would cause major shifts in the center of mass. Which would require unnecessary tuning of the flight controller.



Figure

The next cabin design was used for the final design of the drone for this project. It went back to follow the tiered structure which would allow for the most surface area to mount components while also keeping the frame relatively stable and lightweight. The plates were separated and locked in place by four 2” spacers. This allows for enough separation between the two plates, while being able to easily manipulate the components on the inside. Figure shows the final design of the drone cabin.

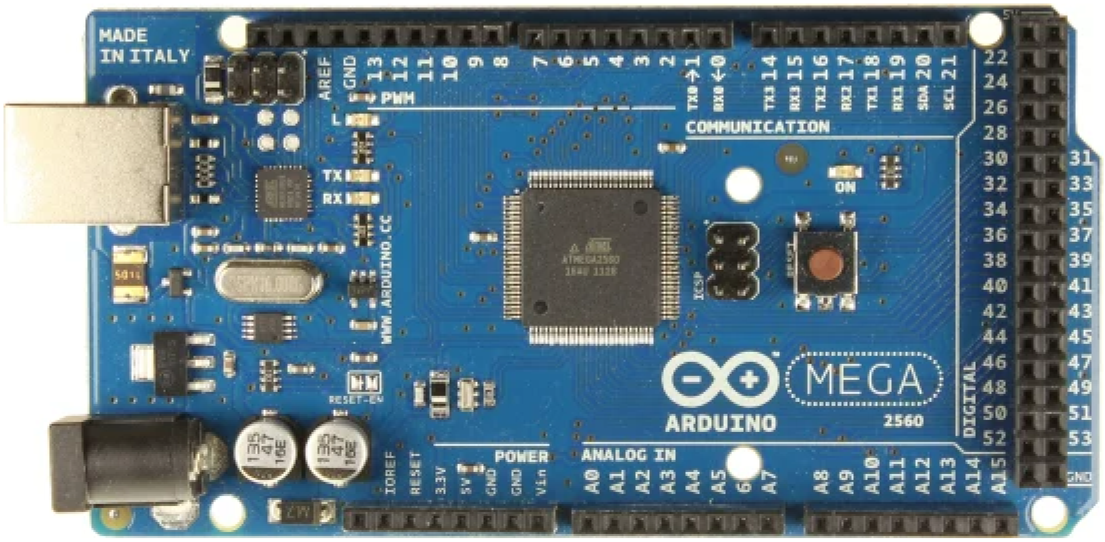


Figure

The next portion of the drone was to design a method to extinguish a fire with the first idea being a miniature fire extinguisher.

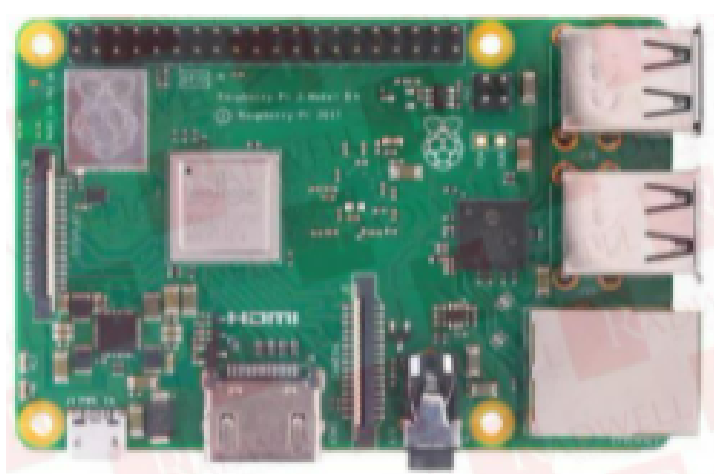
3.4 Components Description

Arduino Mega



The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. It has 54 digital input and output pins, 16 analog inputs, 4 UARTs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. We used the Arduino Mega to test a majority of our sensors and to test the flight of our drone.

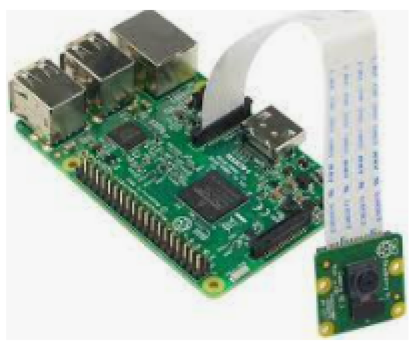
Raspberry PI 3 Model B



Figure

The Raspberry Pi 3 Model B is a small microcontroller board with 40-pin extended GPIO, 4 USB 2 ports, 4 Pole stereo output and composite video port, a Full size HDMI, CSI camera port for connecting a Raspberry Pi camera, DSI display port for connecting a Raspberry Pi touchscreen display, micro SD port for loading your operating system and storing data, and upgraded switched Micro USB power source up to 2.5A. We used the raspberry pi with the Pixhawk flight controller to control the motors. We are now powering the Arduino with the Raspberry Pi through the USB port. Together the raspberry pi and arduino will communicate serially in order to fly the drone and use all of the sensors together.

Raspberry PI Camera



Figure

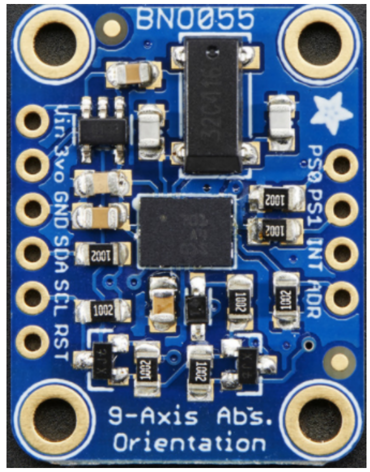
can be used to take high-definition video

Pixhawk Flight Controller

Figure

The pixhawk is an advanced flight controller that allows for both autonomous and controlled flights of copters, planes, and blimps. The brains behind the controller is a STM32F427 chip with 256 KB of SRAM and 180 MHz processing power. This chip allows for intensive calculations necessary to compute the vehicle rotational forces and speed in real time. The ST Micro L3GD20H 16 bit gyroscope, ST Micro LSM303D 14 bit accelerometer / magnetometer, Invensense MPU 6000 3-axis accelerometer/gyroscope, and MEAS MS5611 barometer are sensors on board that work in combination to get an accurate reading of the vehicles speed, orientation, and altitude. The 16 bit gyroscope is a sensor that measures orientation and angular velocity. The 14 bit accelerometer and magnetometer measures acceleration and earth's magnetic field. The barometer measures altitude based on the changes in air pressure above and below sea level. The pixhawk also has 14 on board connectors that are used for various actions and interactions between the vehicle, pixhawk, any companion computers and RC controllers. The connectors used for this project were the GPS module, I2C splitter/compass module, telemetry 1 and 2, power and switch (all can be seen in figure). Telemetry 1 and 2 both use serial communication i.e UART protocol to communicate with the Raspberry Pi and Sik Radio. The GPS and I2C splitter/compass module is used to communicate with the 3DR GPS unit on the copter.

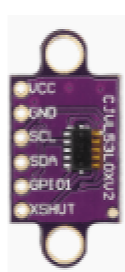
Axis Absolute Orientation Sensor (BNO055)



Figure

The BNO055, also known as an IMU, is a smart 9-DOF sensor that does the sensor fusion all on its own. An IMU combines input provided by several different sensor types in order to accurately output movement. Certain sensors like a MEMS accelerometer, magnetometer and gyroscope are used together to create a single component that is the IMU. With a high speed ARM Cortex-M0 based processor it can digest all the sensor data, abstract the sensor fusion and real time requirements away, and spit out data you can use in quaternions, Euler angles or vectors. This component did not make it onto the final product due to time constraints. We originally planned to create our own flight controller using this IMU and a barometer but realized for the sake of time we should use an actual flight controller. Though we didn't include this IMU in the final project, it was still used during tests.

Time of Flight Distance Sensors (VL53L0x)



Figure

This sensor is a carrier/breakout board for ST's VL53L0X laser-ranging sensor, which measures the range to a target object up to 2 m away. The VL53L0X uses time-of-flight measurements of infrared pulses for ranging, allowing it to give accurate results independent of the target's color and surface. Distance measurements can be read through a digital I²C interface. The board has a 2.8 V linear regulator and integrated level-shifters that allow it to work over an input voltage range of 2.6 V to 5.5 V. We will be using 3 of these sensors to detect objects and anything that could possibly get in the way of the mission for the drone. These sensors are crucial in ensuring that the task is completed successfully without damaging the drone.

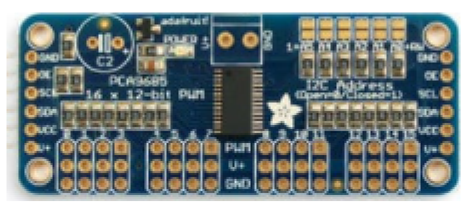
Barometer (BMP085)

The BMP085 is a basic sensor that is designed specifically for measuring barometric pressure. It also does temperature measurement on the side to help. This is another component that we used for testing purposes but did not end up on the final project. This barometer combined with the IMU would create the flight controller.

Sg90 micro servos

This servo is lightweight and small with high output power. It can rotate approximately 180 degrees (90 in each direction), and works just like the standard kind but smaller. And it is compatible with any servo code, hardware or library. We are using 3 servo motors to implement our project. The first servo will be connected to a mount for the raspberry pi camera mount; this will control the angles that the camera will be able to see in. Attached to the second servo will be a pin that when turned on will pop the fire extinguishing balloon. The last servo will be used for the kill switch. If something goes wrong with the drone we will use this servo to turn off a switch that powers the raspberry pi.

Adafruit 16-Channel Servo Driver



The Adafruit 16-Channel 12-bit PWM/Servo Driver will drive up to 16 servos over I2C with only 2 pins. The on-board PWM controller will drive all 16 channels simultaneously with no additional Arduino processing overhead. This servo driver will be used to drive 3 servos safely. Using this component was the best decision due to the fact that it is dangerous to power servos with a raspberry pi.

4s Battery 6000mAh

This 4s lipo battery has the most advanced production technology and provides a steady discharge curve and a high discharge rate to reach the top speed with the most stability and power. It can hold more power and will enhance the drone's overall ability. With this battery we can achieve a good balance between the performance and flight time. The battery will provide our motors power, which will propel the propellers. Therefore, the drone's strength will depend on how much power is available in the battery. With this battery we will be able to power all of the components on our drone.

Matek X class 12S PDB

This FCHUB-12S is designed specifically for X Quad. It supports 8~60V DC (3~12S LiPo) input. It comes with 5V 5A and 12V 4A regulators and high precision up to 440A current sensor range. There are 2 BEC 5V output (5V pads), 3 BEC 12V output (12V pads), as well as a switchable 12V output (SW12V pad). We are connecting a 4s lipo battery to this power distribution board which will power a majority of the components used on our drone. It will power our 4 electronic speed controllers, as well as the servo driver through the 5V BEC output, and the raspberry pi with the other 5V BEC output.

35A 2S-6S ESC

This electronic speed controller will regulate and control the speed of a motor. As the duty cycle or switching frequency of the transistors are adjusted, the speed of the motor is changed. We are using 4 ESCs for our 4 brushless DC motors.

3.X Machine Learning for Fire Detection

The development of the artificial intelligence field has given room to the expansion of automation. This project leverages this expansion of the field to the automation of fire detection. The desire to use machine learning (ML) algorithms in fires stems from the need to have a precise prediction method to detect fire on materials and structures. These systems should be as precise as possible and with minimum margin of error, including human error. The more data used to train the algorithms the better it gets at determining if there is a fire present in the image.

The Raspberry Pi 3B and tbe Pi Camera module were in charge of obtaining the video to conduct real-time fire detection. The main aspect of this system is the incorporation of a neural network to conduct real-time fire detection. Neural networks are inspired by the inner workings of the neurons in our brain. These algorithms are created by stacking up node layers in a sequential manner. The Neural Network includes an input layer, hidden layers, and output layers. An example of a Deep Neural Network can be seen in Figure Y.

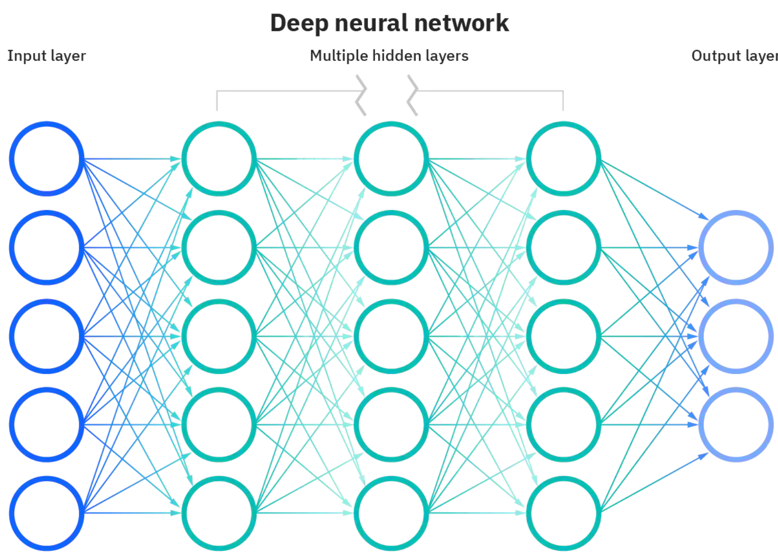


Figure Y. Deep Neural Network (IBM, 2020)

The neural network built for this project is a fairly simple one with an input layer to 64x64 RGB images. The next layers consist of a 2D Convolutional Layer with a Rectified Linear Unit (ReLU) activation function, an Averega Pooling layer, and a Dropout Layer. This block of layers is repeated three times before connecting to a Flatten Layer. After the Flatten Layer we incorporated a Dense layer with a ReLU activation function. Lastly, there are two Dense Layers, the first one with another ReLU activation layer and the second with the Softmax activation layer. Lastly the network will contain the output layer. Figure X showcases the network's architecture.

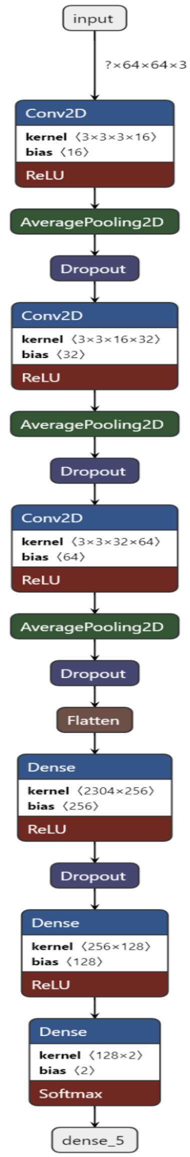


Figure X. Fire-Detection Neural Network

The network was build on a Windows 10 computer, however the project requires the transition to the Raspberry Pi 3B operating system. An obstacle encountered in this project was this required transition. The network was built in a Python environment with the help of the Tensorflow, Numpy, and Keras library. Tensorflow is currently not supported in the Raspberry Pi 3B due to its heavy size. To overcome this issue, we implemented Tensorflow Lite which is lighter than the normal Tensorflow. Additionally, a virtual environment for the building of all the libraries needed was created on the Raspberry Pi. The team made sure to obtain all possible dependencies needed to use the Tensorflow Lite library. Tensorflow Lite is designed to compute network inferences on devices with low computational power. The Tensorflow library is capable of translating the trained model into a Tensorflow Lite version. The Tensorflow Lite version of

3.3 Engineering Standards

When becoming an engineer of any kind, there are standards that all must abide by to maintain a high level of quality throughout every product. When these standards are followed, not only will they provide a great product for the consumer but they will also keep the consumer safe when the product is handled properly. There are many engineering standards that span every discipline but the ones used for this project are coding, wiring, and communication.

This project involved using multiple different wires to run power from the battery to all the different components on board. When selecting wires, we made sure to pick ones that were able to carry the circuit's voltage and current level safely. We wanted to make sure none of the wires would fry or cause a short while the drone was in the air. This would not only harm the drone but could also harm one of us, or a bystander, if it were to fall out of the air.. We also made sure to keep the wires short, compact, and color coded for easy maintenance in case a problem were to occur.

The brains behind this project were the Pixhawk and Raspberry Pi, making it extremely reliant on code. Coding standards require that code be heavily commented, simple and easy to understand. The code should also follow naming conventions for variables and functions, functions should be kept short, and the code should be efficient. Coding standards were carefully followed because not only was there alot of code to write but the whole group contributed to it. This kept the code uniform, understandable and allowed any member of the team to debug anything if necessary.

Communication standards refer to proper use of different communication protocols when building a network of devices that transmit data between one and another. When creating and programming our network we chose to use protocols that would allow for fast transmitting speeds that would also keep our data safe. We used both inter and intra protocols. Inter protocols are used when two devices like a computer and microcontroller want to communicate, intra protocols are used when two devices want to communicate within a circuit. We used UART, a subcategory of inter protocols, to allow the pixhawk and Raspberry Pi to share data through serial communication and used I2C, a subcategory of intra protocols, to allow the pixhawk and GPS unit to communicate with each other.

Professional Components:

Sustainability plays a huge role in the design of a fire fighting drone. It is important that the drone is able to follow all of these standards to ensure that production leaves a positive footprint on the world. The fire fighting drone will have an environmental impact , with its material use and the ability to reduce, reuse, and recycle. The energy use plays a big role in sustainability with either how much it utilizes and how it is disposed of. Societal impact would be how it interacts with people around them, such as safety issues or health issues. There is also an economic impact with the money saved or money spent.

Environmental Impact

Most of the issues presented that would impact the environment would be the material use and its disposal. There are a myriad of materials being used on the drone. Some of the more common materials used on the drone are different types of metals. The metals are zinc, copper, fiberglass, and ferrite. Other materials used are PLA filament, wood, plastics, rubber, and latex. The copper is found in the soldered components and pcb's. Zinc is mostly used in the frame of the drone and in order to process the material, it does have negative effects. The processing impacts water, soil, and crops and extended durations of it negatively affects humans. Copper is another material with a similar footprint to zinc. Copper is relatively easy to recycle with little negative impact on the environment. If the project was to continue and turn into a final sellable product, we would move into Carbon Fiber REinforced Composite. Carbon fiber is a strong and lightweight material that is easily recyclable through methods such as chemical, mechanical and thermal.

Energy use

In order to power the UAV. 4s LIPO battery will be used as the main power source. A LIPO battery is a lithium polymer battery used for its high current capacity and watt hour. A lipo battery is a reharble source of energy which allows for multiple uses before needing to be disposed of. The main issue with this type of battery is the difficulty of disposal. If they are not properly disposed of they will take 100 years to degrade. For future projects we would opt to use solar powered batteries. Ths

Pollution Issues:

The pollution issues that will stem from the drone will be whatever debris that might be left over after a mission. Currently after a mission is complete there will be balloon debris left over. The latex in the balloon could take up to four years to degrade, which can harm the local wildlife. During the degrading process the debris could be blown into the wildlife in the surrounding area or end up in some body of water. No matter where it ends up, it won't go into a trash can. This is also due to the fact that we want to involve as few people as possible. The lithium polymer batteries also add to the pollution issues. These

Conclusion:

If none of the code of ethics were violated, the innocent men that were killed in the missile explosion as well as the people involved in the Challenger space rocket catastrophe would have lived to see another day. These cases show the importance of abiding by these codes and the reasons why they are put in place. In each case, there was pressure to complete the given task to meet a certain deadline in time and this led to the managers being careless and negligent. Whether it is pressure or not, nothing should lead to negligence. As an engineer, you must follow the code of ethics and ensure that the safety and wellbeing of the workers and the public are being considered first and foremost. More problems will be caused by not following these codes and at the end of the day, the deadlines won't be reached, and companies could possibly be faced with lawsuits instead. It is imperative that enough data is collected, and simulations run to ensure that a task will be completed successfully and safely. It is more important to delay development if there is not enough data to prove that the procedure can be done safely. And if an issue is brought up, it should not be ignored. Lastly, even though not everyone is held to the same standard as engineers, as the engineer you must stress that the code of ethics is to be followed and support one another in their findings and knowledge. By following these ethics codes, this is the only way engineers can positively affect the quality of life around the world.

Project Implementation

Results:

Discussion

Problems that arose: motors, battery, flight controller, limited storage of the given microcontroller board, neural network

At the beginning of our project we decided to use a drone given to us by Dr. Rios that was used in a previous project. We figured that this would be the best plan of action due to the fact that the drone was recycled. The problem that came with this drone was that the motors were not good enough to carry out the mission successfully. During the first month of testing we used these motors and we didn't make much progress with flight. We thought that the propellers weren't put on correctly and so we switched them multiple times. After switching them, we tried testing again but the drone still had trouble taking off and we believe that it was a weight issue that contributed to this. As the design process went on, we calculated the weight of the components used. Some of the heaviest items used are the 4s battery and the "grenade" itself. We got more motors hoping to solve this problem but those motors were from another older and recycled drone and unfortunately for us one of the motors didn't work. To avoid another issue like this we eventually made the decision to purchase brand new motors. After researching and calculating we found some motors that met our specifications and could lift our drone. Though this was a small success we couldn't celebrate just yet. Our drone was able to get lift but the flight wasn't stable. No matter what we did, one side of the drone would get more lift than the other and if given enough power, this would cause the drone to completely flip over. Some items and many propellers were damaged beyond repair during these tests. This quickly turned into a dangerous environment. Because testing was so critical, we had two different sized propellers on the drone at one point. We tried adjusting the speeds of the motors that were lagging as well as slowing down the time it took for the motors to reach full speed. No matter what we tried, we couldn't get the drone to a level hover. We came to the conclusion that we needed some type of flight controller which leads into our next problem.

We had several problems when designing and developing our flight controller. We initially wanted to build our own flight controller with the BNO055 and Raspberry Pi at the heart of it. We chose this IMU device because it offers fast data transfer speeds with its built-in I2C bus. We quickly realized the BNO055 was unable to communicate with the Raspberry Pi reliably due to the I2C interface using clock stretching. Clock stretching, i.e. pulling the SCL line down low for an extended time, slows down communication and allows the slave device to do what it needs to until it is ready to proceed. The ARM-processor, Broadcomm BCM2835, on the Pi cannot handle this feature because of a manufacturing issue that was far too complicated for us to understand. After many hours of researching, we found two solutions that worked. We could either use the device's built-in UART bus or slow down the rate at which the I2C bus reads from 100kbps to 1000bps. We decided against both of these because we wanted the maximum transfer speeds the IMU had to offer. We then decided to switch microcontrollers and use the Arduino Uno. The Uno supports clock stretching and was able to reliably read the date from the IMU. After a few tests with the uno, making sure the readings we received from the IMU were accurate, we decided to implement the code necessary to control the motors and see if we could get the drone to hover. We then ran into another issue, storage. The uno's 32Kb of programmable storage was not enough to store the script necessary to make the drone hover. This was a major issue because we still had to implement the code for the barometric sensor and other components outside the flight controller that will also be used by the drone. This was easily fixed by using an Arduino Mega instead. The mega's 256Kb of programmable storage was more than enough and the added bonus of 4Kb of EEPROM allowed us to store calibration data for each sensor. After about a month of problem solving we finally had a working prototype of our flight controller.

After a few more weeks of testing, the data we were getting from the IMU started to become inaccurate. The roll, pitch and yaw values being produced were obviously faulty and we assumed this to be a software error. After tearing through the code and confirming that each calculation for the independent rotational forces were accurate, we decided to test the independent sensors on the IMU to see if there was a hardware issue causing this. We quickly realized that this was the case. The accelerometer and gyroscope on board were producing inaccurate data while the magnetometer was fine. We decided to buy a new IMU because we still had a lot of work to do on the drone and fixing the independent sensors would take too much time. We wanted to get the same IMU but it was out-of-stock. After a few days of research, we decided to get the MPU9250. It was much cheaper than the BNO055 but still offered the same 9-axis motion tracking. This change added a whole new set of problems that were unavoidable. The first being that swapping the IMUs involved us having to rewrite all our code from scratch. The MPU9250 has very few libraries built for it and the ones that are available have a huge learning curve to be able to use them. After taking a week to learn the libraries, we began testing the IMU to make sure each sensor produced accurate results and the fusion algorithm produced accurate rotational forces. Once we were able to confirm the readings were accurate we began to rebuild our flight controller with the MPU9250 at the heart of it. After programming the IMU to make the drone hover, we then attempted to add the barometric sensor allowing us to control the altitude at which the drone will fly. We wired up the barometric sensor to run on the same I2C bus as the IMU and tried to run a simple script to get the altitude of the drone while also reading the rotational force vectors, but we kept getting errors saying the mega cannot read the values from the barometric sensor. After some research, we realized this error came from the fact that the IMU acts as a master when another sensor is added to the bus. This issue could be easily remedied by putting the IMU in passive mode, as stated in the MPU9250 datasheet, allowing the sensor to bypass the IMU and talk directly with the microcontroller. We started to work on implementing this solution but were lucky enough to receive a pixhawk from Dr. Haddad, a pre-built flight controller that we would use on the final product of our drone. The issues encountered with both iterations of the flight controller taught us a lot of valuable lessons that we hope to implement in the future design changes of our drone.

Through the numerous tests we performed, one of the components we needed most was a reliable battery. We started our testing off by using a variable power supply. This did not last long because the voltage kept dropping and this was causing the voltage to not be distributed to all parts of the drone. This was before we acquired the power distribution board. Our next course of action was to use lipo batteries that we found in the lab since this is what the power source would be in the end; we didn't have many options to choose from. One of the batteries had a substantial amount of swelling that we immediately stopped using it. The other battery we use for the next 2-3 months of testing. The problem with this battery was that when we were on the brink of discovery, it would die on us thus ending our testing for the day. The charge didn't last long enough and we weren't always in the vicinity of a charger due to the fact that we were working on this many times outside of lab time. Because of our low budget we were reluctant to buy anything brand new. But in our frustration we ordered a new 4s battery and this helped testing to run much smoother.

Due to the problems listed above, the success of our project was continuously delayed.

Multiple aspects of our project have drastically changed since the start of FiFi's journey. We changed our minds about whether we wanted to use a Raspberry Pi or an Arduino Uno/Mega or both at least 3 times. We fried one of our raspberry pi's and this forced us to use an arduino instead. This set our progress back by a lot and put our programmers into overdrive to get code working for the Arduino. The good thing that came from this was that we have code for each sensor and we can make the drone fly with both the arduino and raspberry pi. This process gave everyone a chance to improve their programming skills with the different languages. After running into issues with the flight controller we were making with the arduino, we decided to switch back to using the Raspberry Pi 3B. We acquired the Pixhawk flight controller as well and began testing with that. After encountering problems with the Pixhawk we decided not to use it anymore and went back to our flight controller that we made with the Arduino Mega. We are now using both the Arduino Mega and Raspberry Pi 3B. They both have their advantages over the other but it's better to get the best of both rather than settle.

Conclusion-

References-

1. FEMA. (2019). (rep.). *Fire in the United States 2008-2017* (20th ed., pp. 1–98).
2. Covington, T. (2021, June 2). *House fire statistics and facts (data from 2020) - the zebra*. The Zebra. Retrieved February 5, 2022, from https://www.thezebra.com/resources/research/house-fire-statistics/
3. D. W. Casbeer, R. W. Beard, T. W. McLain, Sai-Ming Li and R. K. Mehra, "Forest fire monitoring with multiple small UAVs," *Proceedings of the 2005, American Control Conference, 2005.*, 2005, pp. 3530-3535 vol. 5, doi: 10.1109/ACC.2005.1470520.
4. M. G. Bocanegra and R. J. Haddad, "Convolutional Neural Network-Based Disaster Assessment Using Unmanned Aerial Vehicles," *SoutheastCon 2021*, 2021, pp. 1-6, doi: 10.1109/SoutheastCon45413.2021.9401928.
5. A. Imdoukh, A. Shaker, A. Al-Toukhy, D. Kablaoui and M. El-Abd, "Semi-autonomous indoor firefighting UAV," *2017 18th International Conference on Advanced Robotics (ICAR)*, 2017, pp. 310-315, doi: 10.1109/ICAR.2017.8023625.
6. (fire class types) https://www.nfpa.org/News-and-Research/Publications-and-media/Blogs-Landing-Page/NFPA-Today/Blog-Posts/2021/07/16/Fire-Extinguisher-Types
7.

Appendice-

Code and design diagrams