MediDrone

Group 28

Final Design

Group Members:

Varadha Anandakumar (ME)
Lior Barak (AE)
Ryan Grant (EE)
Joey Hodson (EE)
Seth Horowitz (EE)
Seba Villalobos (CpE)

Project Advisors:

Dr. Samuel Richie
Dr. Lei Wei

Spring 2023

Sources of Project Motivation

Scarcity of medical deliveries and infrastructure during emergencies motivates the birth of MediDrone.

The following conditions Medidrone addresses include but are not limited to

- Spoilage of medicines during emergencies
- Delivery of medicines & goods during emergencies
- Optimization of medical deliveries by adding the domain of flight
- Lack of technological access to medical delivery drones
- High cost of commercial drones

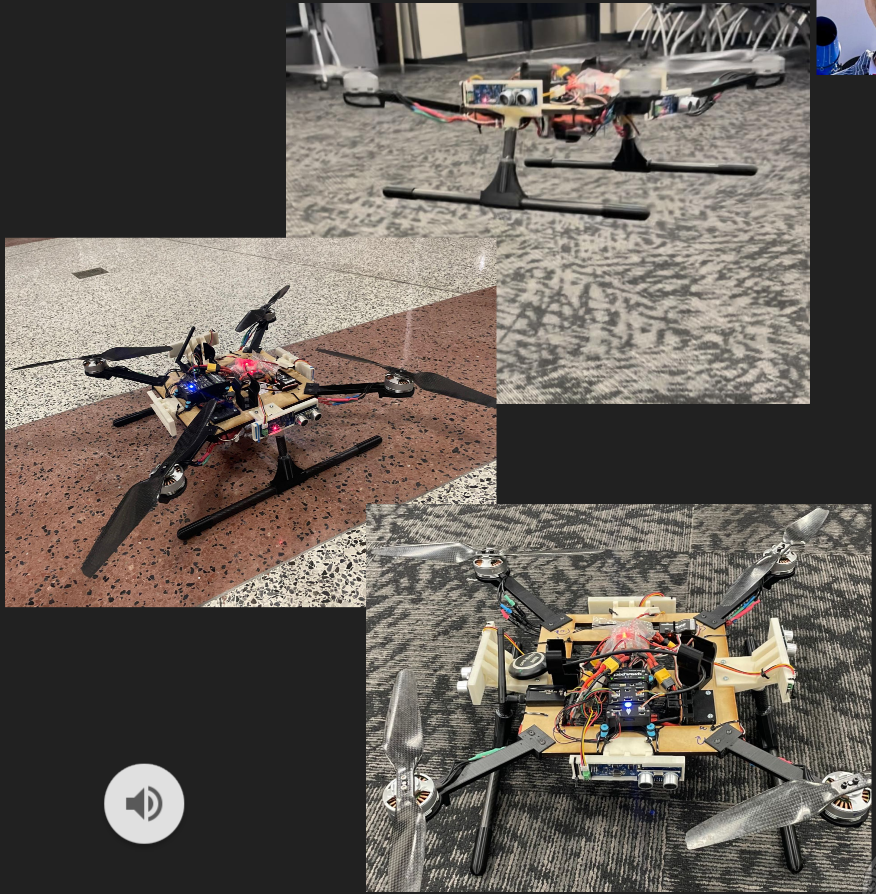
Motivation to design and build MediDrone is simple: adding flight to medical delivery will SAVE LIVES

What is MediDrone?

- MediDrone is a type of quadcopter drone designed to transport temperature-sensitive medical equipment or medication from a starting location to a designated endpoint.

- Aims to optimize medical transport to benefit both businesses and individual consumers.

- Aided partnership with Chapie to accomplish our payload



How will MediDrone be utilized?

- MediDrone will be utilized for medical supply transportation.

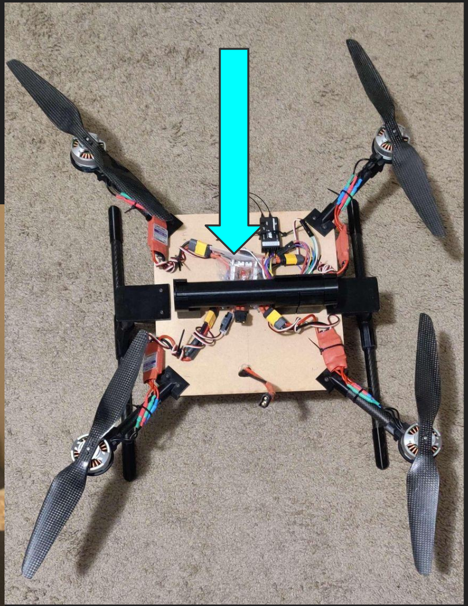
- Deliveries to time-sensitive and hard-to-reach locations, such as hospitals and disaster-stricken areas, will be most efficient.

Payload Encapsulation

To keep the medical payload safe, MediDrone utilizes Chapie technology. The Chapie is inserted into the insulated payload enclosure; inspired by tennis ball canisters and designed by the MediDrone mechanical team.

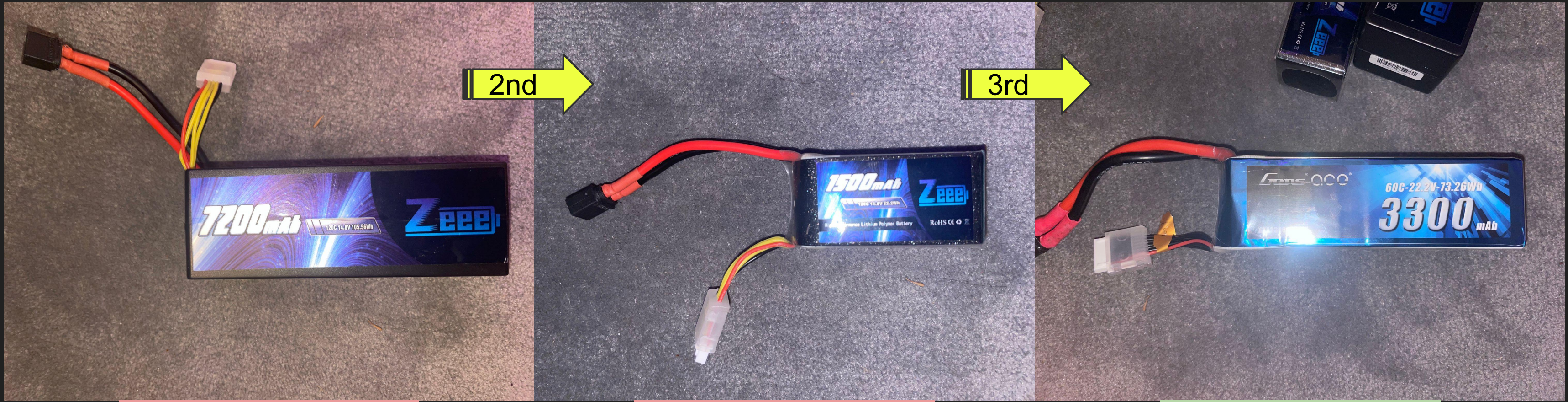
Features:

1. Easy attachment and detachment to and from MediDrone
2. Encapsulation of multiple chapie enclosures
3. Accessible and lightweight encapsulation



Power Supply Iterations

- Debugging poor flight led us down different rabbit holes from software, power distribution, current supply, weight reduction, weight distribution, and finally the root cause being the voltage supply itself.



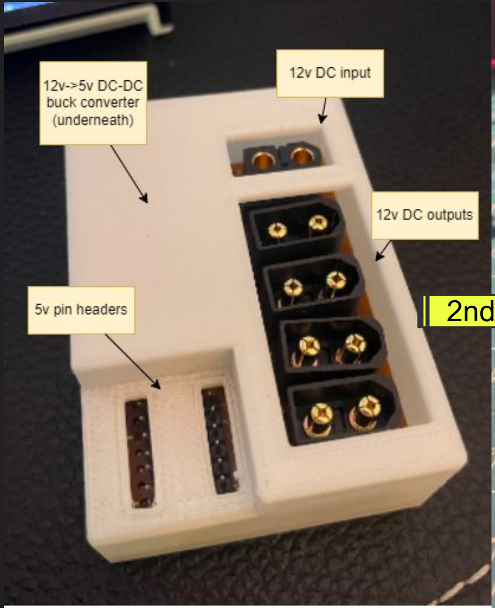
4s; 7.2Ah; 120C

4s; 1.5Ah; 120C

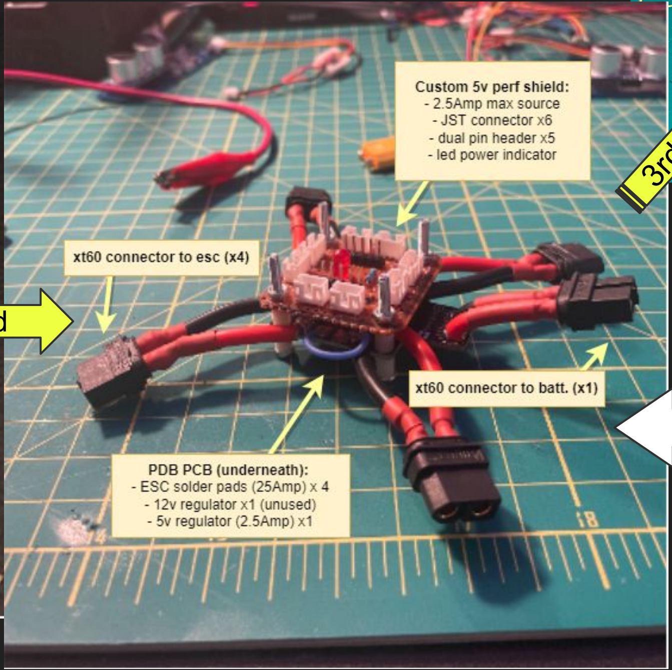
6s; 3.3Ah; 60C

Power Distribution Iterations

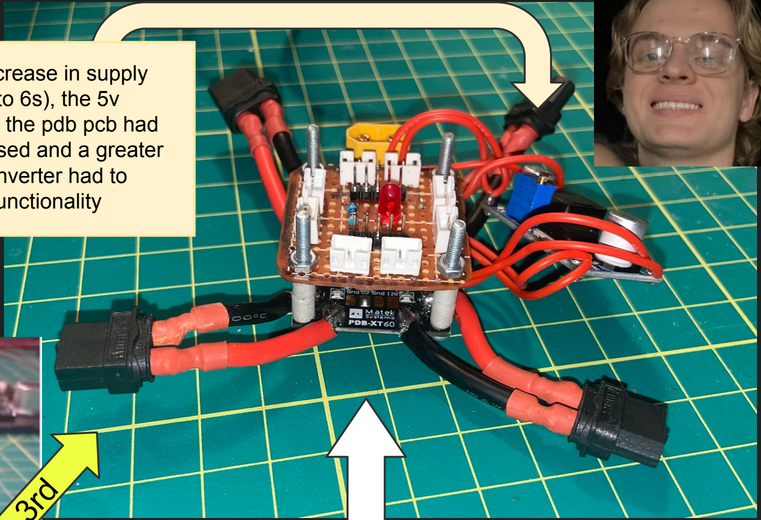
Given an increase in supply voltage (4s to 6s), the 5v regulator on the pdb pcb had to be bypassed and a greater Vin buck converter had to replace its functionality

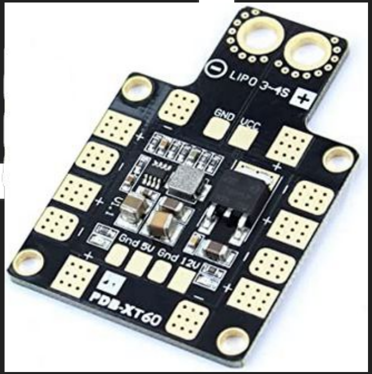


2nd

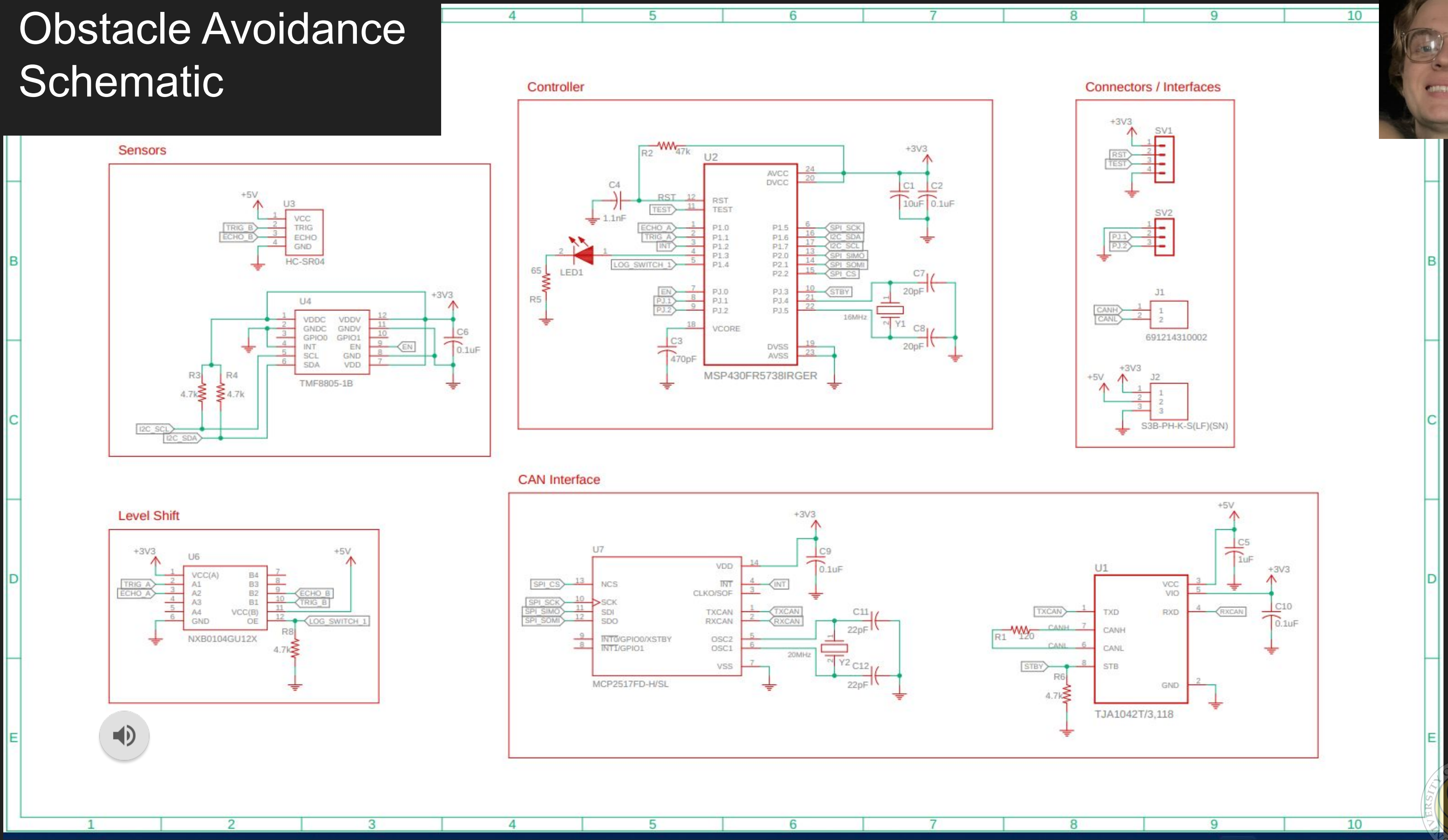


3rd

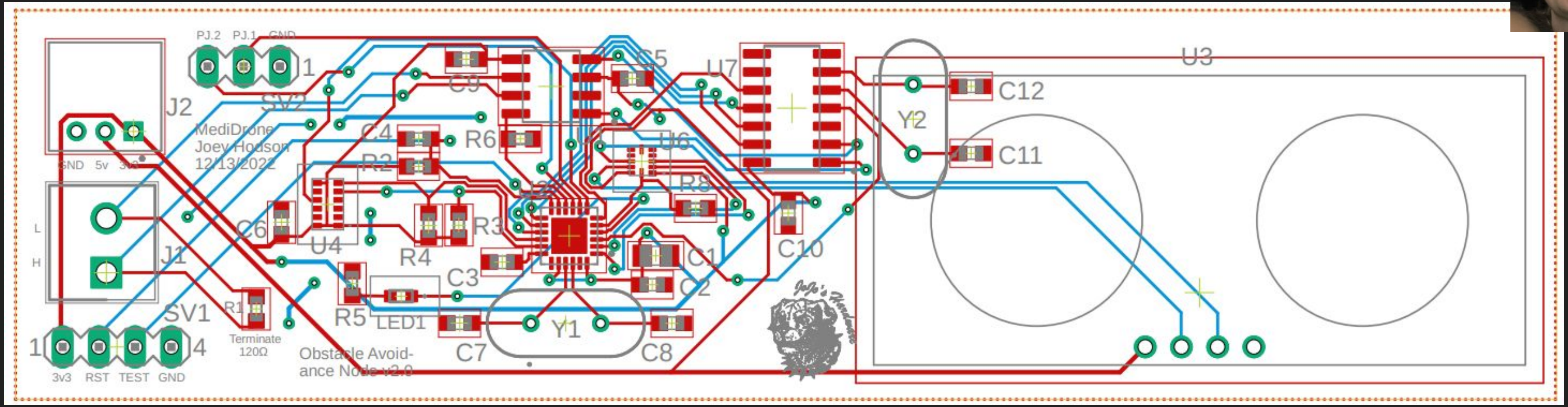


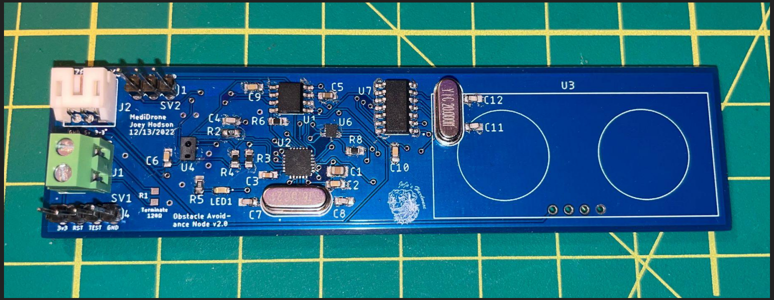


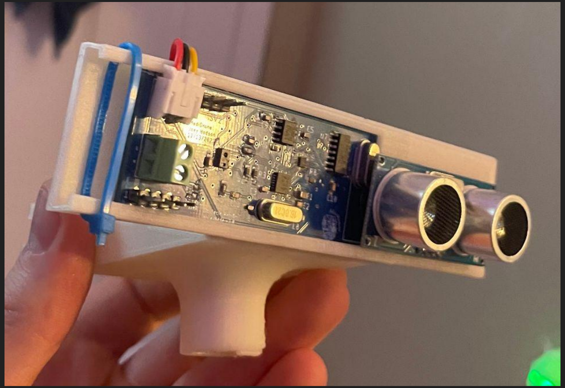
Obstacle Avoidance Schematic



Obstacle Avoidance Board Layout / Assembly



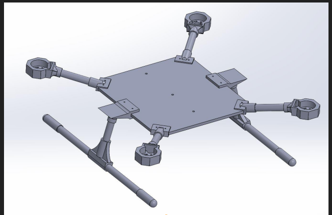




Obstacle Avoidance Short Demonstration

Design/Assembly

11/15/2022



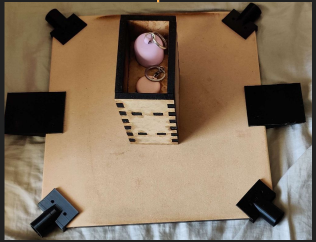
2nd drone design complete

01/25/2023

1st set of motor mounts

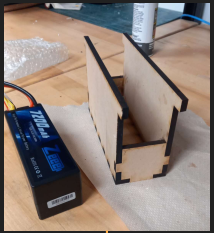
02/05/2023

1st drone build



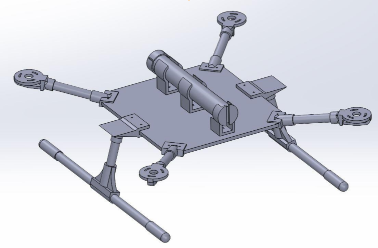
02/12/2023

1st payload container



02/16/2023

1st battery container



02/26/2023

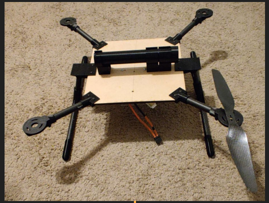
3rd drone design complete

03/04/2023

2nd payload container

03/13/2023

2nd battery container



03/15/2023

Main components attached

03/17/2023

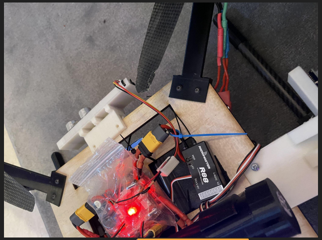
motors/props attached

2nd drone build

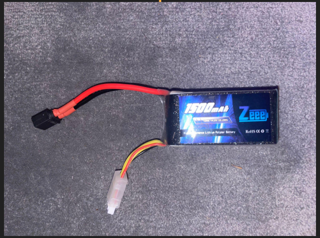
3/19/2023

Design/Assembly cont.

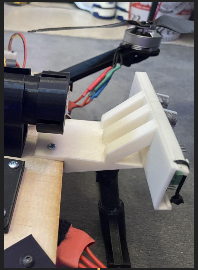
3/19/2023



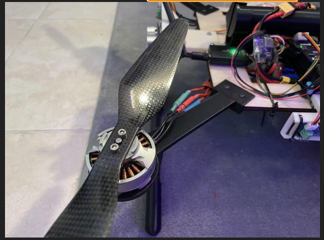
Sensor mounts on



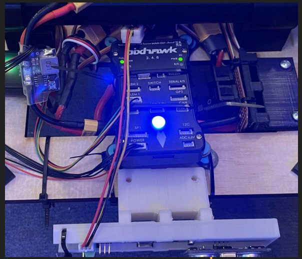
New battery(smaller)



New landing gear/sensor mount

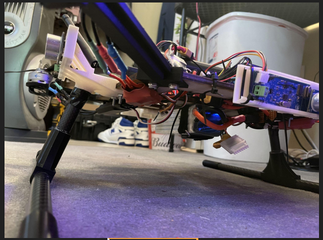


New motor mount



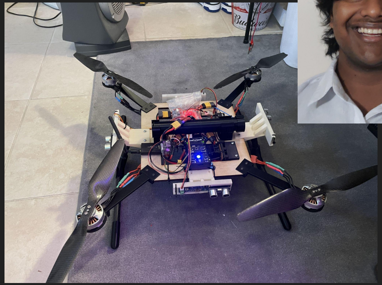
Added flight controller

New baseplate



Wire management

New battery(large)



Final drone build

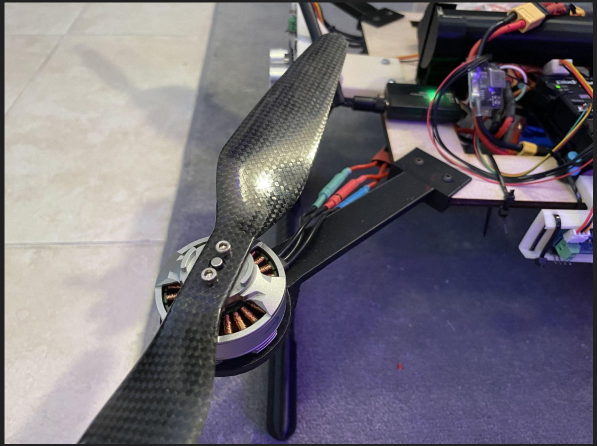
Present

Major Changes

- Motor mounts
  - The motor mount has been redesigned 4 times for various reasons
- Baseplate
  - The design was revamped for the purpose of reducing weight and creating space for electronics
- Electronic Mounts
  - All of the mounts are easily accessible due to the new baseplate
  - The mounts all use Velcro or zip ties to restrain the components but also make them easily accessible.
- Flight controller
  - Added the controller for auto pilot assistance and for better flight control during testing phase.
- Battery
  - The power source was changed for multiple reasons
- Sensors
  - The sensors have been mounted and are functional

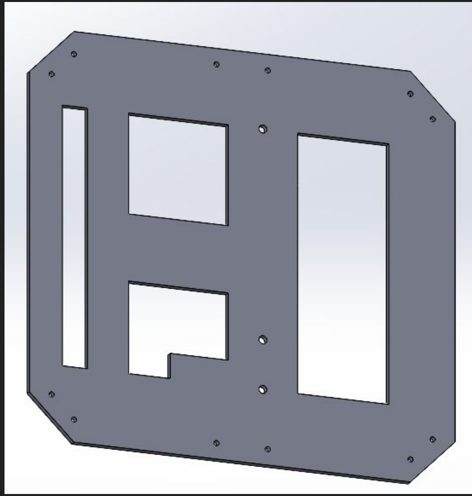
Motor Mount

The current motor mount was created to prevent the rotation of the mount. The previous design was rotating due to the strength of the motor which created the drone to spin in air or crash.



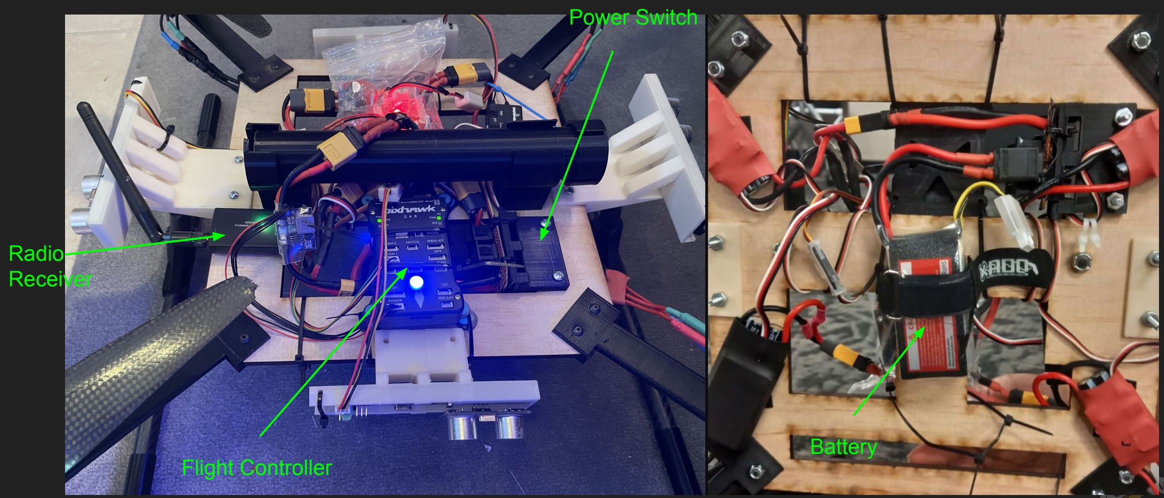
Weight Reduction

- Weight was a concern as the drone could not easily launch off the ground
- Changed the type of wood to Plywood
- Re-designed the base to remove excess material



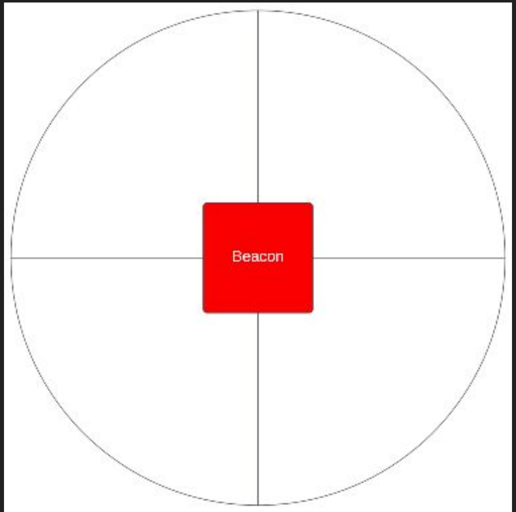
Electronic Mounts

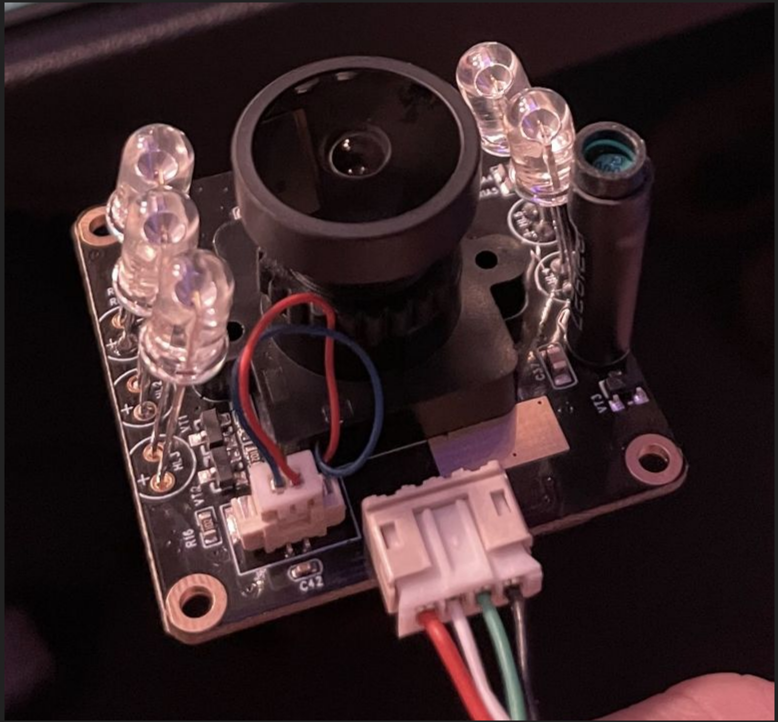
- The redesigned base made it possible to easily include the final electronic components
- Power switch is firmly attached in the front hole and is easily accessible
- Flight Controller is stabilized in the front
- Radio Receiver and wires have ample space
- New battery mount uses Velcro, negligible weight, very accessible.



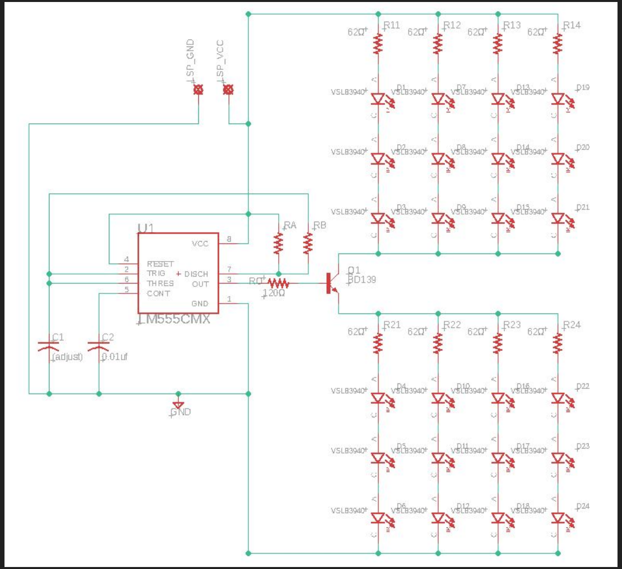
Precision Landing Using Infrared LED Beacon

- Single-Beacon System
- Ideal Optimal Distance
  - Goal (Before Testing): 10-20 meters



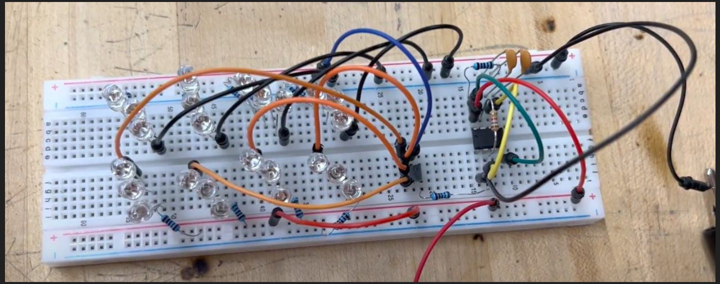


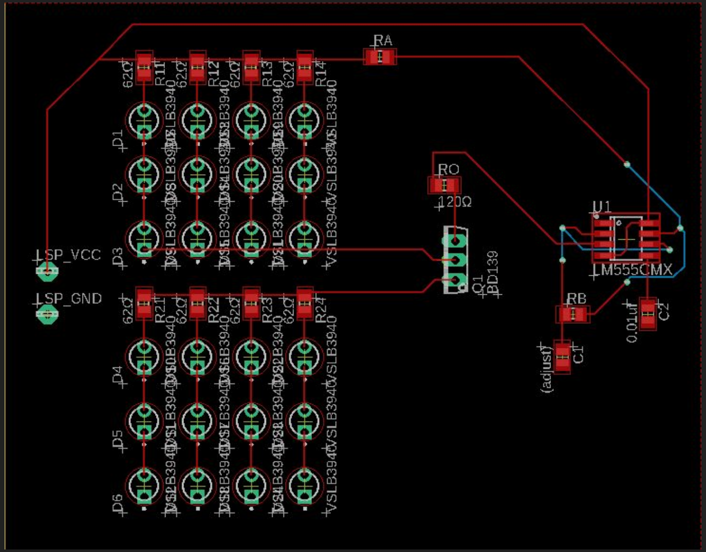
Precision Landing Beacon, Design



Schematic

Breadboard Model



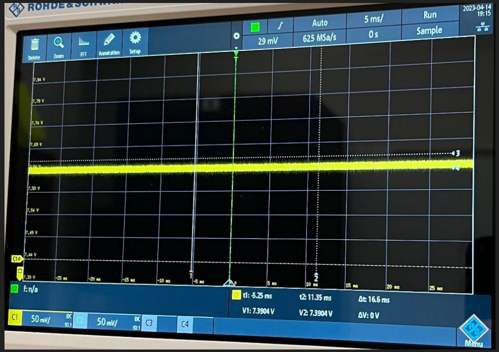


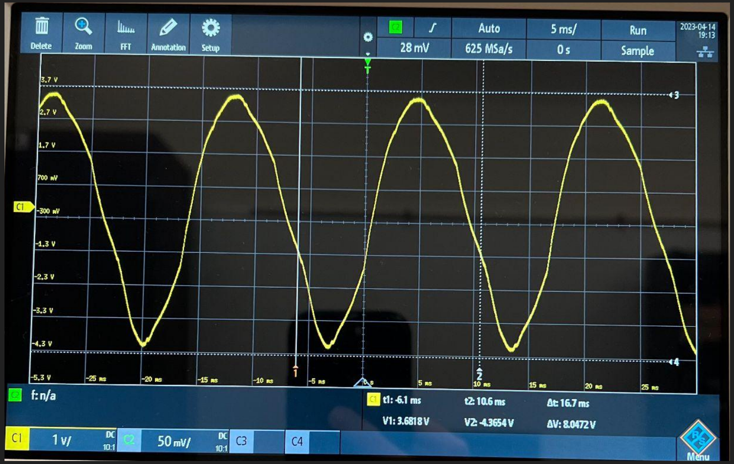
PCB Design

Precision Landing Beacon, Testing

Oscilloscope Reading
Timer Output

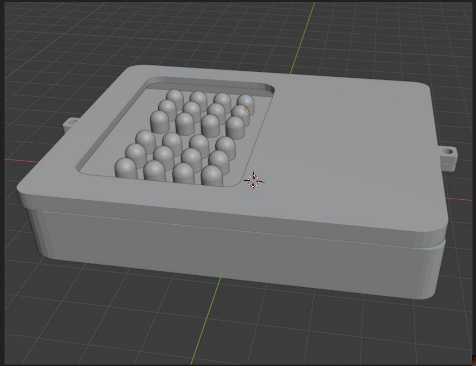
Total Voltage
(9V Battery)





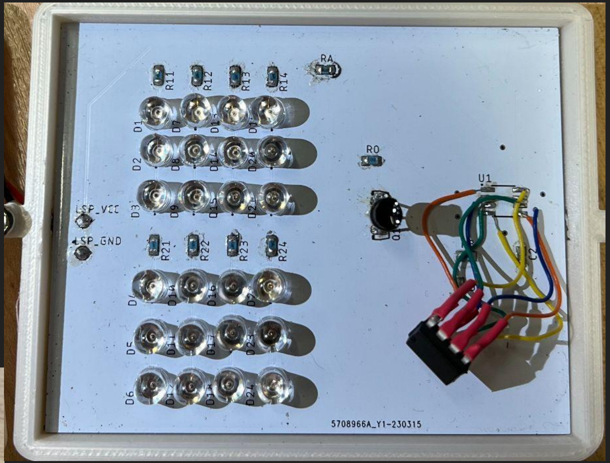
Oscilloscope Reading
Across LED Column

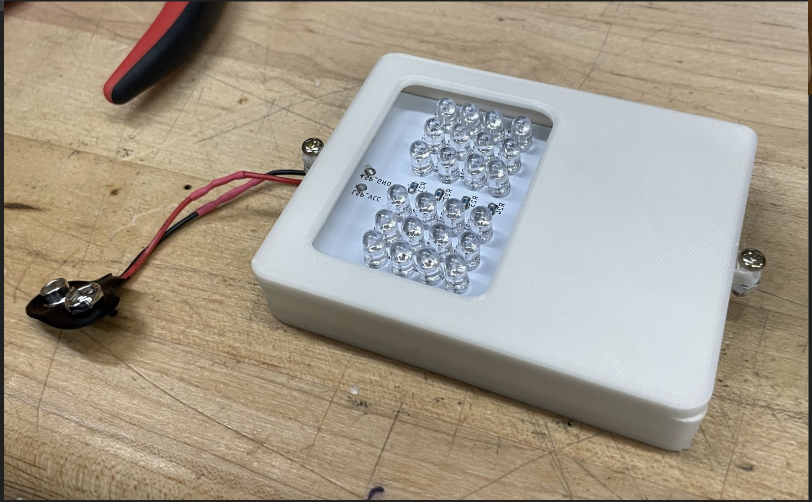
Precision Landing Beacon, Assembly



Enclosure
CAD Model

Beacon
PCB





PCB in
Enclosure

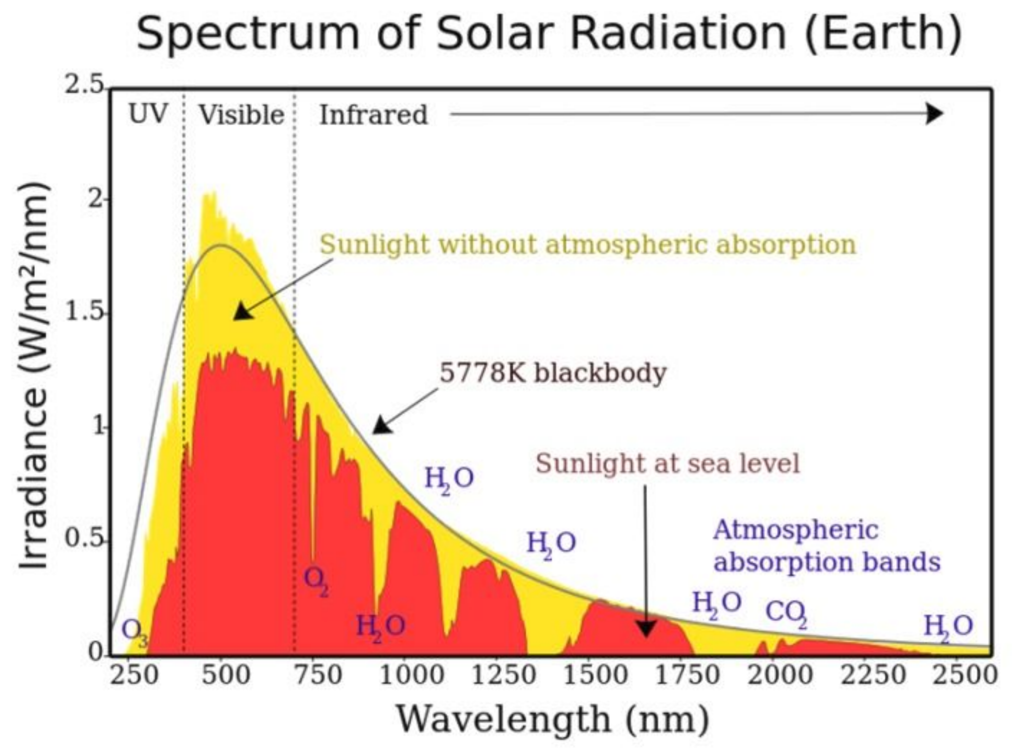
Detecting IR Beacon Outside

Sunlight has a significant infrared component

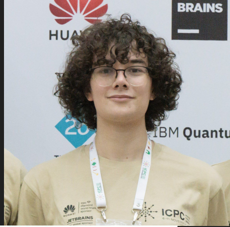
Finding the brightest IR point could be reflected sunlight, instead of the IR beacon

Solved by using a fixed frequency on the beacon

Want to avoid synchronization



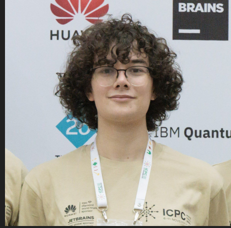
Precision Landing, Detection Algorithm Theory



- Toggle the IR beacon on and off at a preset frequency, T = period
- Find positions in the image that toggle at the same frequency as the beacon
  - Maintain a buffer/window of frames from the last T/2 seconds
  - Find differences between two images
- Positions that match more than some threshold are averaged together
- Requires stable position, but the camera can shake
- Frequency trade off:
  - High frequency: fast location detection, but higher error
    - Bottlenecked by framerate
  - Low frequency: slow location detection, but lower error

```
procedure ProcessFrame(f, frames, states, counts, threshhold, T)
    Enqueue(frames, f)
    while Time(Front(frames)) + T/2 < Time(f) do
        Dequeue(frames)
    end while
    f′ ← Front(frames)                          ▷ Frame with opposite beacon state
    positions ← ∅
    for each position (x, y) do
        if f[x, y] ≠ f′[x, y] then
            s ← 1
        else
            s ← 0
        end if
        counts[x, y] ← counts[x, y] + s − Front(states[x, y])
        Dequeue(states[x, y])                   ▷ Update buffer of previous states
        Enqueue(states[x, y], s)
        if counts[x, y] ≥ threshhold then
            Insert(positions, (x, y))
        end if
    end for
    return Average(positions)
```

Precision Landing, Short Detection Demo



1. Find pixels that are flashing at the correct frequency
   a. XOR current frame with the buffered frame T/2 sec. ago
2. Add to the weight of those pixels
   a. Pixel weight decays over time
3. Take the average of high weight pixels to get a location estimate
4. Look at the last K=5 location estimates, see how close they are
   a. small circle = small error = high confidence

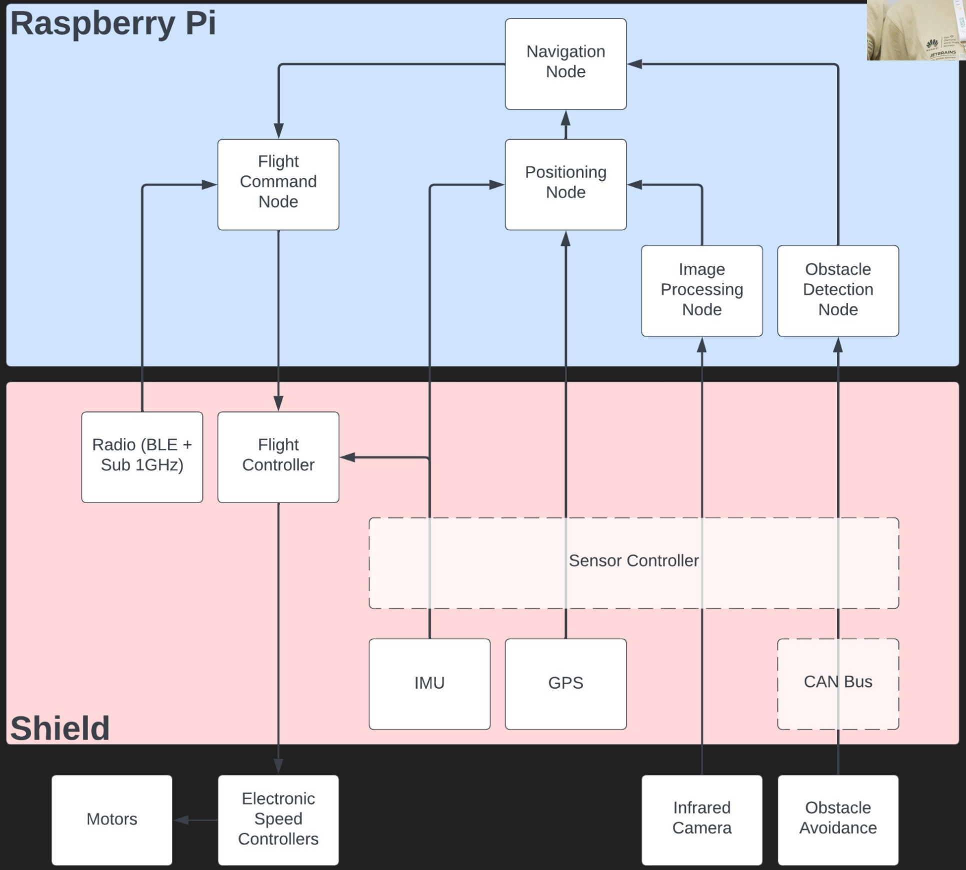
- Error comes from the edges of static light
- Tough test environment: highly reflective surfaces
  - Landing near a lake
  - Pieces of plastic outside

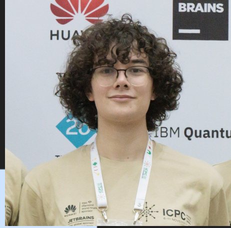
Blue brightness = pixel weight

Green = passes weight threshold

Software Block Diagram

- Uses a Raspberry Pi running ROS as a companion computer
  - Python with C bindings (numpy) has been fast enough
  - Scale up with C++/Rust + NVidia Jetson
- Handles high level mapping and computer vision
- External boards handle lower level control
  - Flight controller: stabilization and command execution
  - Obstacle avoidance: manages ultrasonic sensors





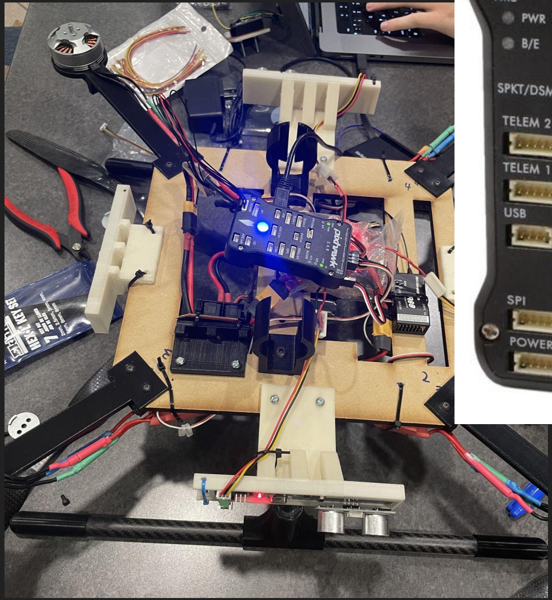
Flight Controller

Pixhawk PX4 Flight Controller

Autotuning PID controller

Interfaces directly with ROS via MAVLink library

Supports Sub-1GHz Telemetry Radio, for QGroundControl



Short Flight Demonstration

Administration Content

Drone Pros/Cons

Pros:

- Increased delivery speed
- More efficient than ground transport
- Reduced human error
- Completely electric
- Cheaper

Cons:

- Short range
- Limited payload capacity
- Single delivery per trip

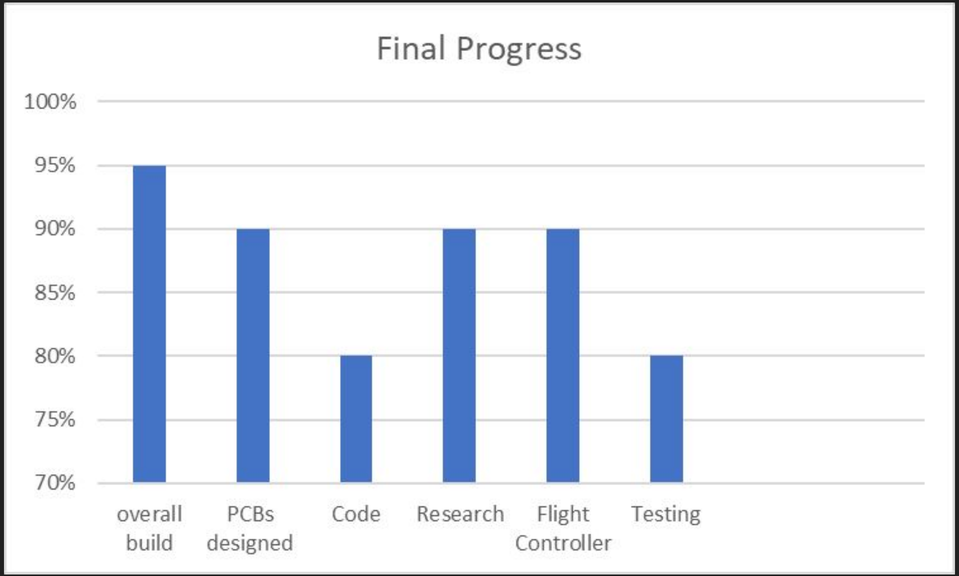
Budget

We were able to use some components from prior Senior Design projects so our actual total is $251.29

Part	# of parts	Price per part	Total price
Motor	4	$65.00	$260.00
Propeller	4	$5.22	$20.88
ESC	4	$28.11	$112.44
Carbon Fiber Rod(4”)	4	$5.95	$23.80
Battery	2	$63.75	$127.50
Plywood(1’ x 2’)	1	$8.00	$8.00
3D-printer Filament(PLA - 1KG)	1	$21.00	$21.00
			$574.61

Final Progress

- Overall Build: 95%
- PCBs designed:90%
- Code:80%
- Research:90%
- Flight Controller:90%
- Testing: 80%



Future Adjustments / Advancements

- Full CAN communication between electronics
- Rebuild the landing gear to be more robust
- Downsize the drone and make it more compact
- Beacon detection algorithm improvements
    - Use areas instead of pixels
    - Optical flow
    - Pulse modulation with more information

Questions & Comments