D.E.A.R Drone

Group B:

Dan Biller, CPE
Ellie Lane, CPE
Austin Perkins, EE
Raymond Chenoweth, EE

UCF

Project Description

- D.E.A.R Drone is an autonomous delivery drone that will deliver a package of ½ pounds.
- The D.E.A.R drone will use sensors such, as GPS and a camera to ensure the package makes it to the destination safely.
- Our mission is to help streamline supply chains to get people their products faster

Specs

Specifications	
Payload Weight	0.5 Pounds
Battery Capacity	6000mAH
Flight time	20min(+-10 vertical)
Field of view	60 degree horizonal
Range of view	.5-10feet
Measuring Frequency	10Hz
Autopilot	PX4

Requirements

Requirement #1: Lift Off	
1.1: The drone should perform preflight calibrations	~30sec
1.2: Be able to take off autonomously within specified time	~30sec
1.3: Be able to intake GPS coordinates from Satellites	~Pre-take off

Requirement #2: Mission	
2.1: The drone should follow a flight path from takeoff to landing autonomously	NA
2.2: The drone should return to launch after delivering the package	NA
2.3: The drone will disarm motors after landing	~5sec
2.4: Drone needs to have the ability to be manually overwritten when encountering critical failure	~1sec

Requirements Continued

Requirement #3: Payload	
3.1: The drone should be able to carry a 0.5lb payload	0.5 Pounds
3.2: The drone should be able to safely drop a payload in the delivery zone	Within 10 Feet
3.3: Drones servos should be able to hold and release package on command	NA

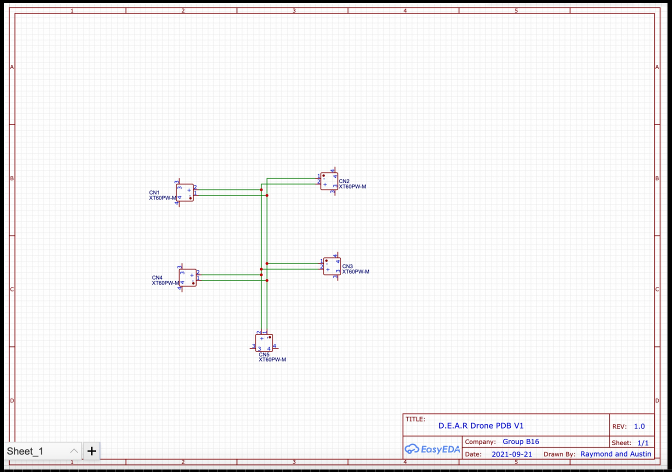
Requirement #4: Sensor I/O	
4.1: The drone should avoid obstacles in flight path using onboard sensors	NA
4.2: Intake data from sensors package onboard Navio 2	NA
4.3: Intake data from Camera and Ultra Sonic sensor	NA

Standards

Standard	Component
IPC-2221 PCB Standard	PCB design and manufacturing
NFPA Electrical Safety Standards	Safety standards for electrical components
PEP8	Python style guide
IEEE standard 802.ac/n	Wi-Fi
Batteries IEEE 485-2010	Standard for Lead Acid and LiPo Batteries
FAA part 107 UAS	Drone Flight

PCB Design

- Power Distribution Board
  - Send 11V evenly to all our ECS’s



TITLE: D.E.A.R Drone PDB V1

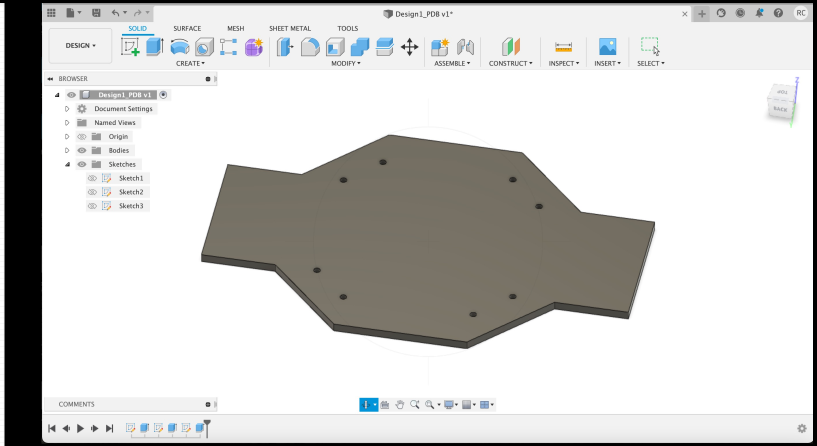
REV: 1.0

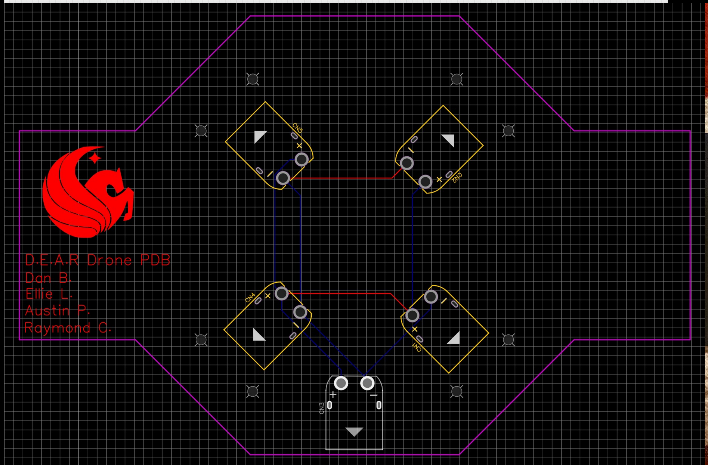
Company: Group B16

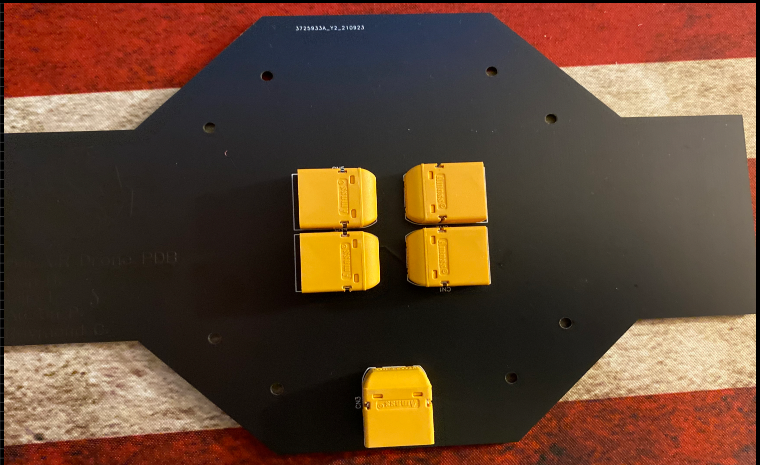
Sheet: 1/1

Date: 2021-09-21

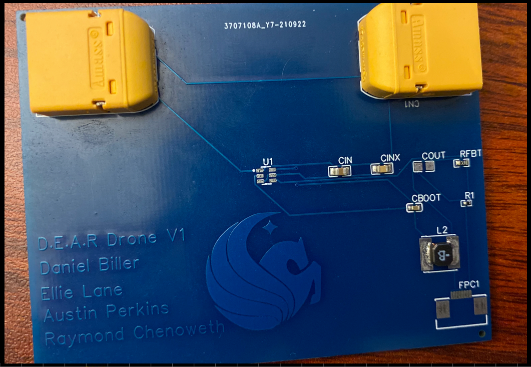
Drawn By: Raymond and Austin

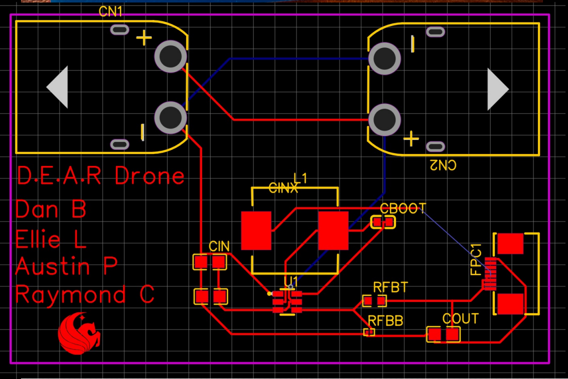




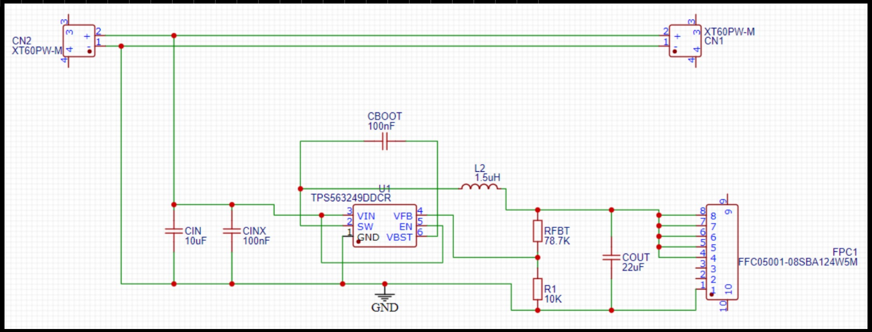


PCB Design



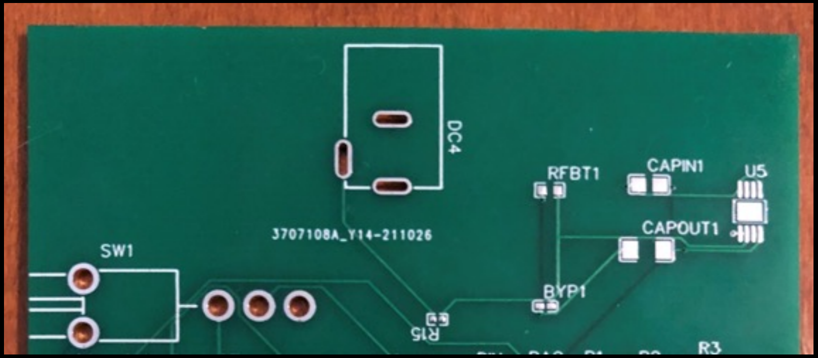


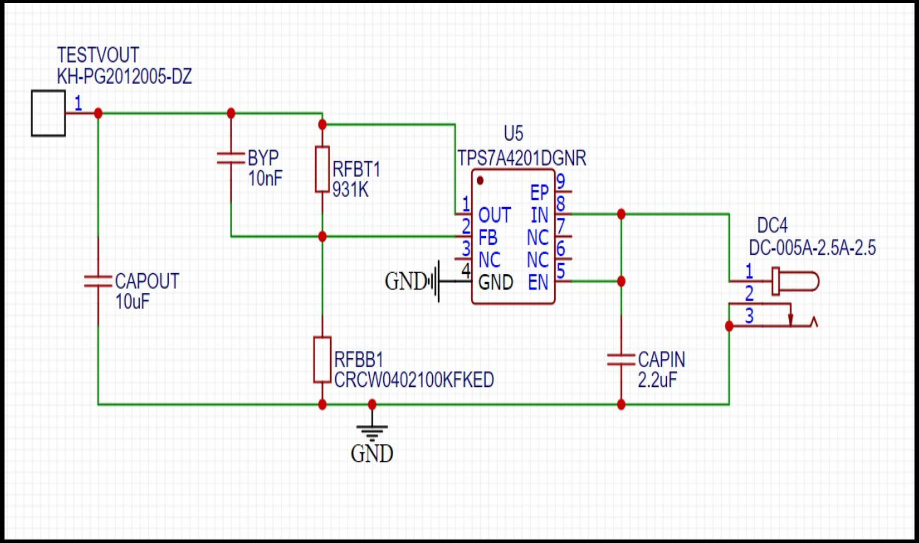
- Power Module
  - Sends 11V to our power distribution board
  - Steps down 11V to 5V and 2.5mA for the flight controller
  - XT60 to XT60 connection

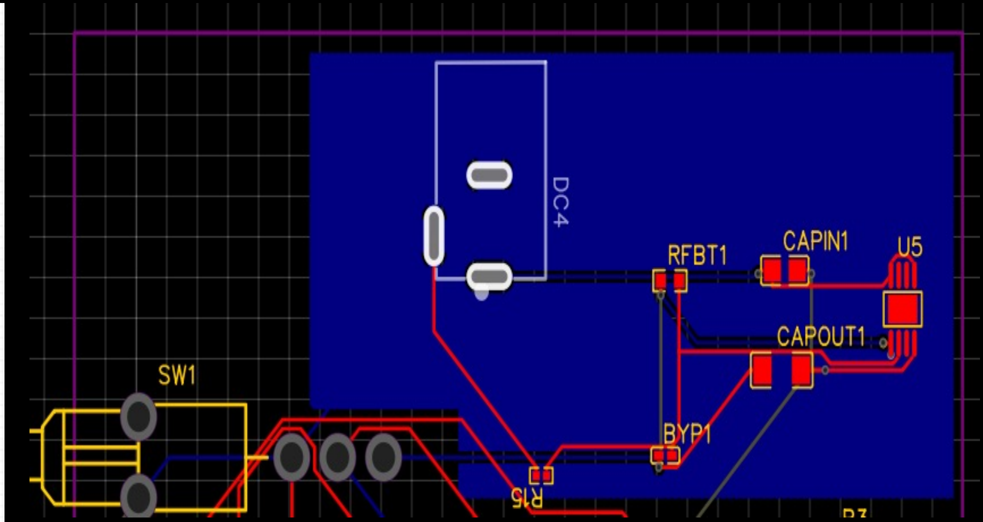


PCB Schematic [DC-DC Converter]

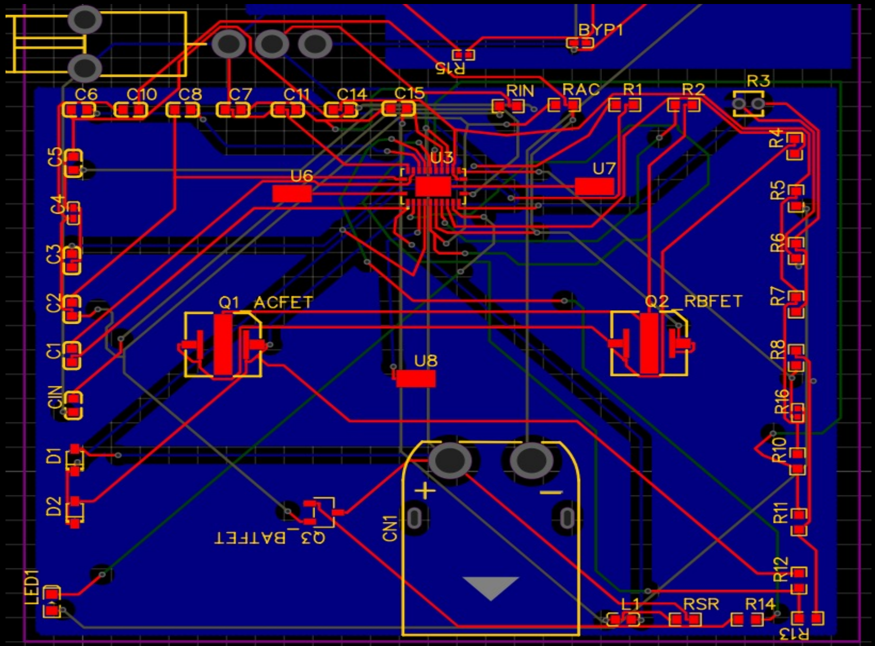
- DC-DC Converter
- Barrel Jack/ Laptop Charger
- Input DC Power 19V
- Output to Battery IC 12V at 1.5mA

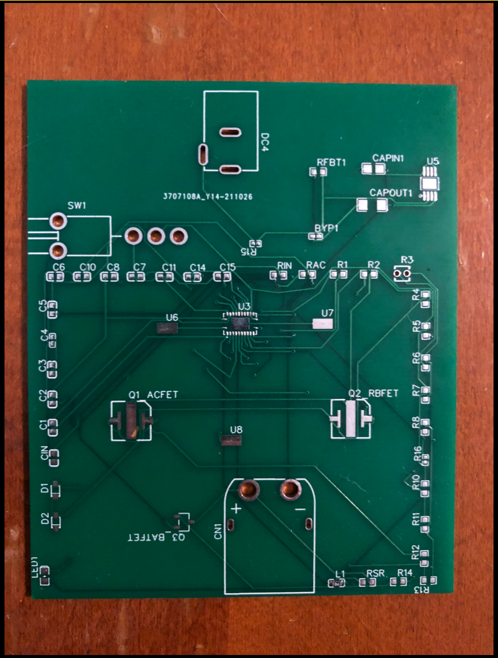






PCB Schematic [Battery Management IC]

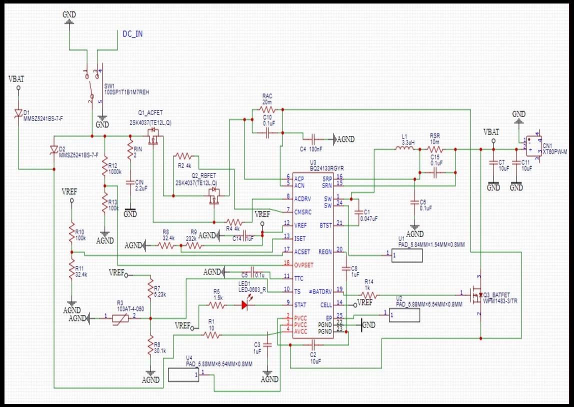




- The design of the Battery Management system was done using the "Typical Application" provided by the Datasheet

- Main Chip BQ24133RGYR and other major components were on 52 week back order so supply chain issues caused us not to have board in time.

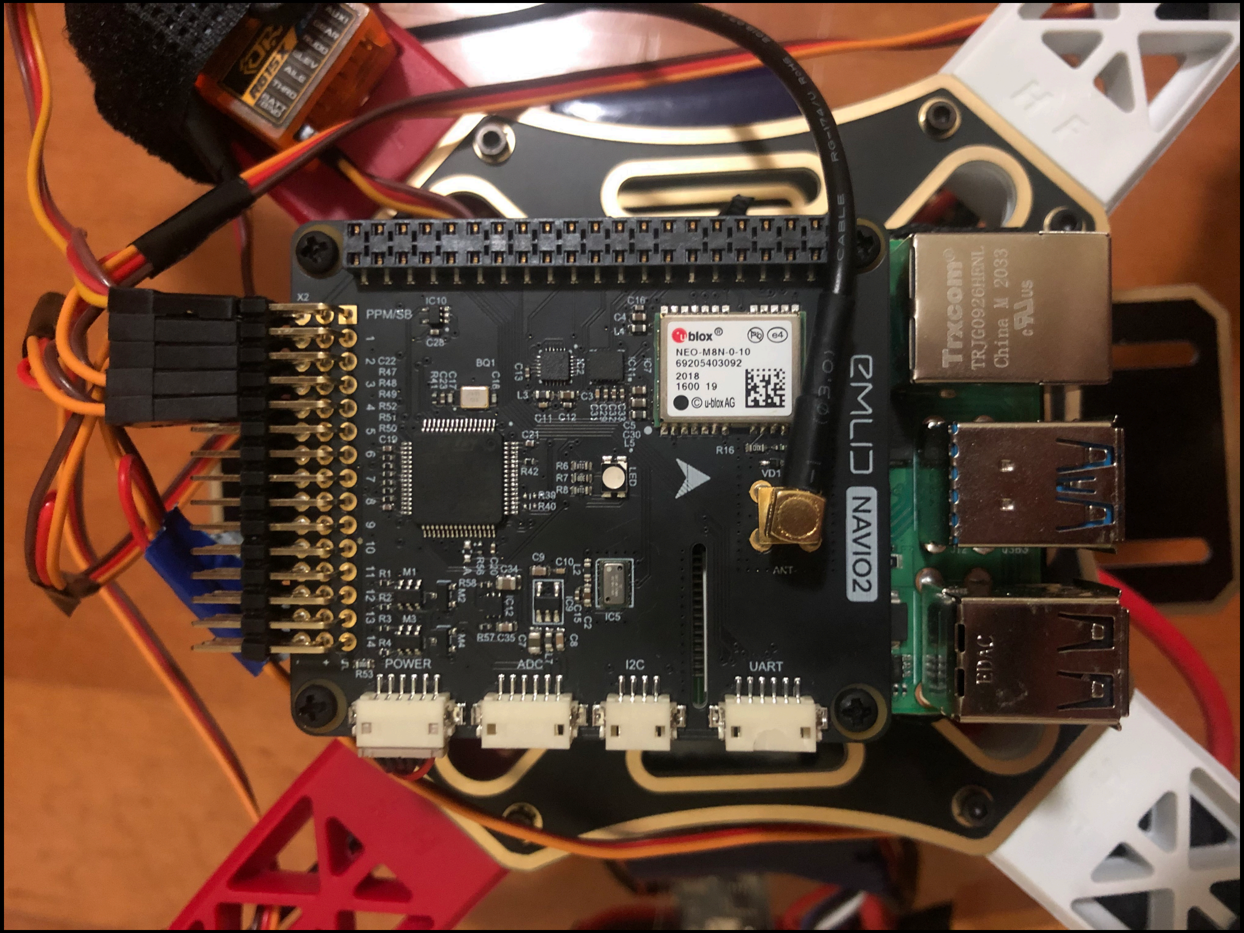
- Using this aspect of our chip, we can charge 3 Cells at one time, protect them, and manage them, it was an ideal choice.



Requirement #1: Lift Off

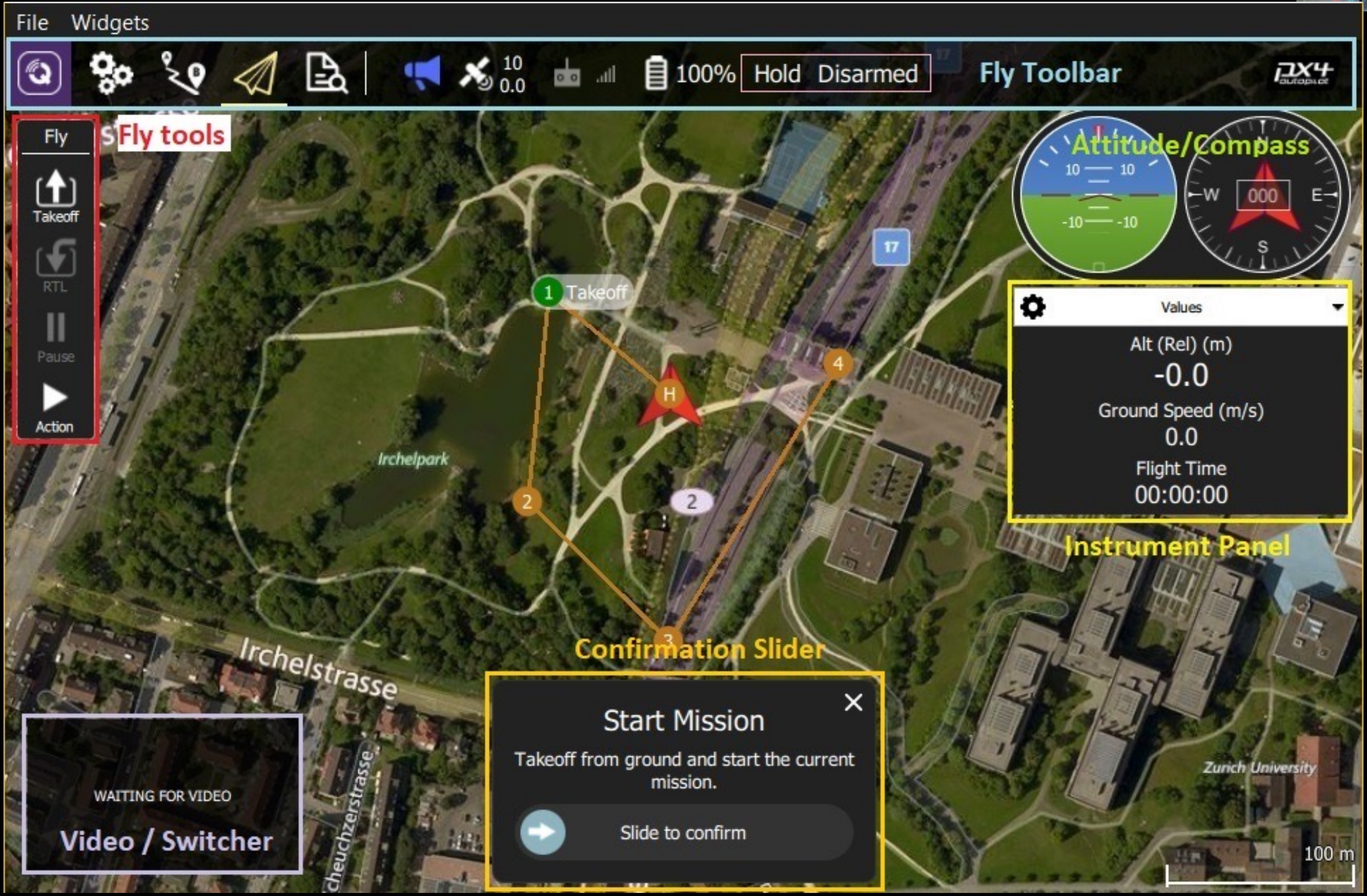
Requirement #1: Lift Off	
1.1: The drone should perform preflight calibrations	~30sec
1.2: Be able to take off autonomously within specified time	~30sec
1.3: Be able to intake GPS coordinates from Satellites	~Pre-take off

Flight Controller



px4 autopilot

QGroundControl



Values

Alt (Rel) (m)
-0.0

Ground Speed (m/s)
0.0

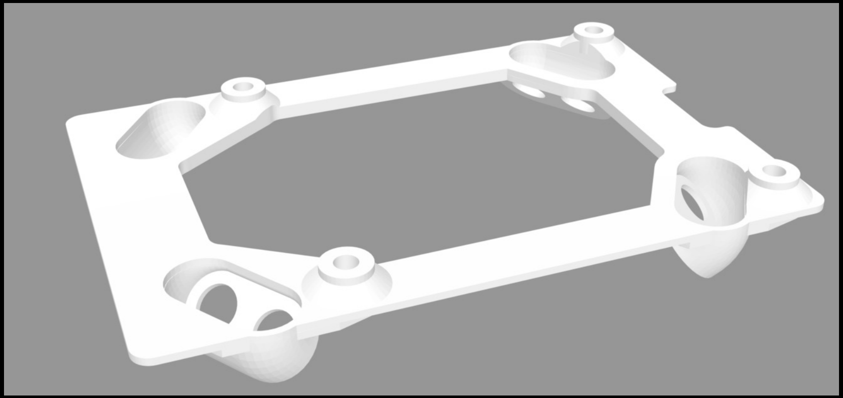
Flight Time
00:00:00

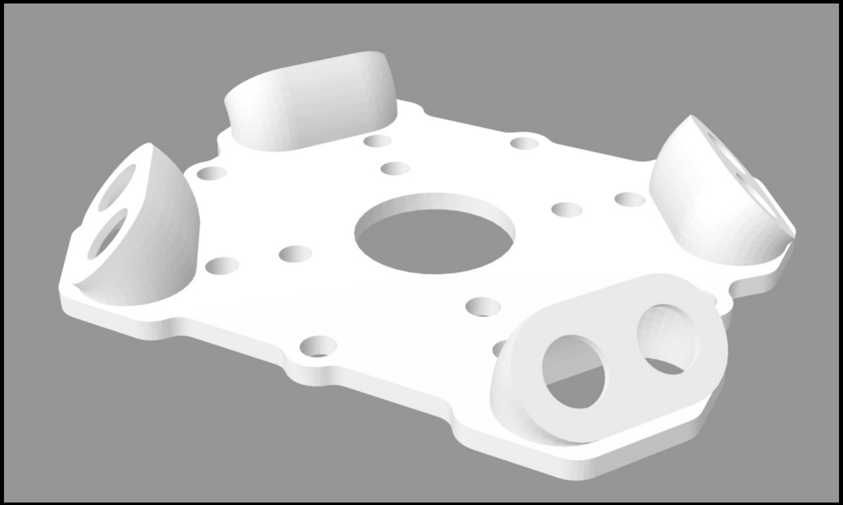
Start Mission

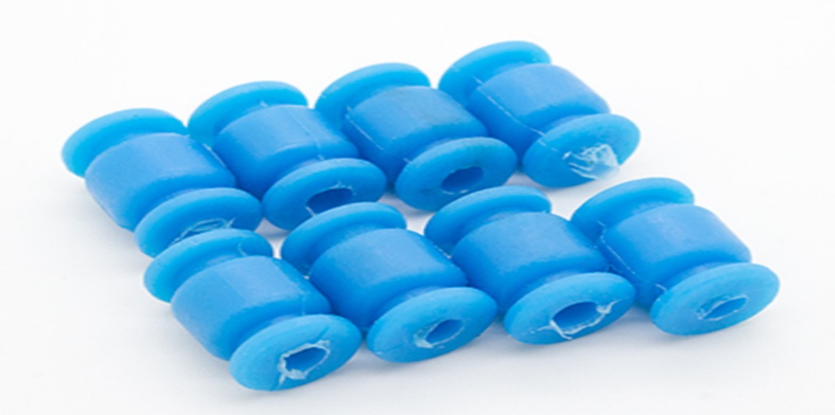
Takeoff from ground and start the current mission.

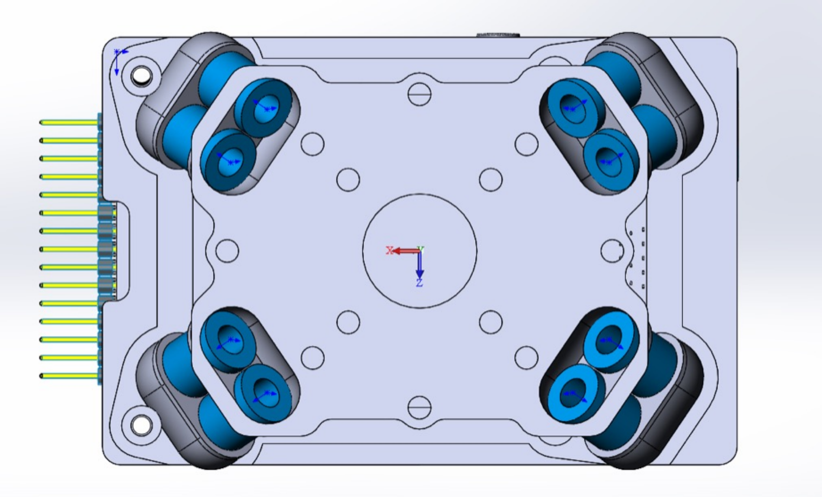
Slide to confirm

Vibration Dampening Mount

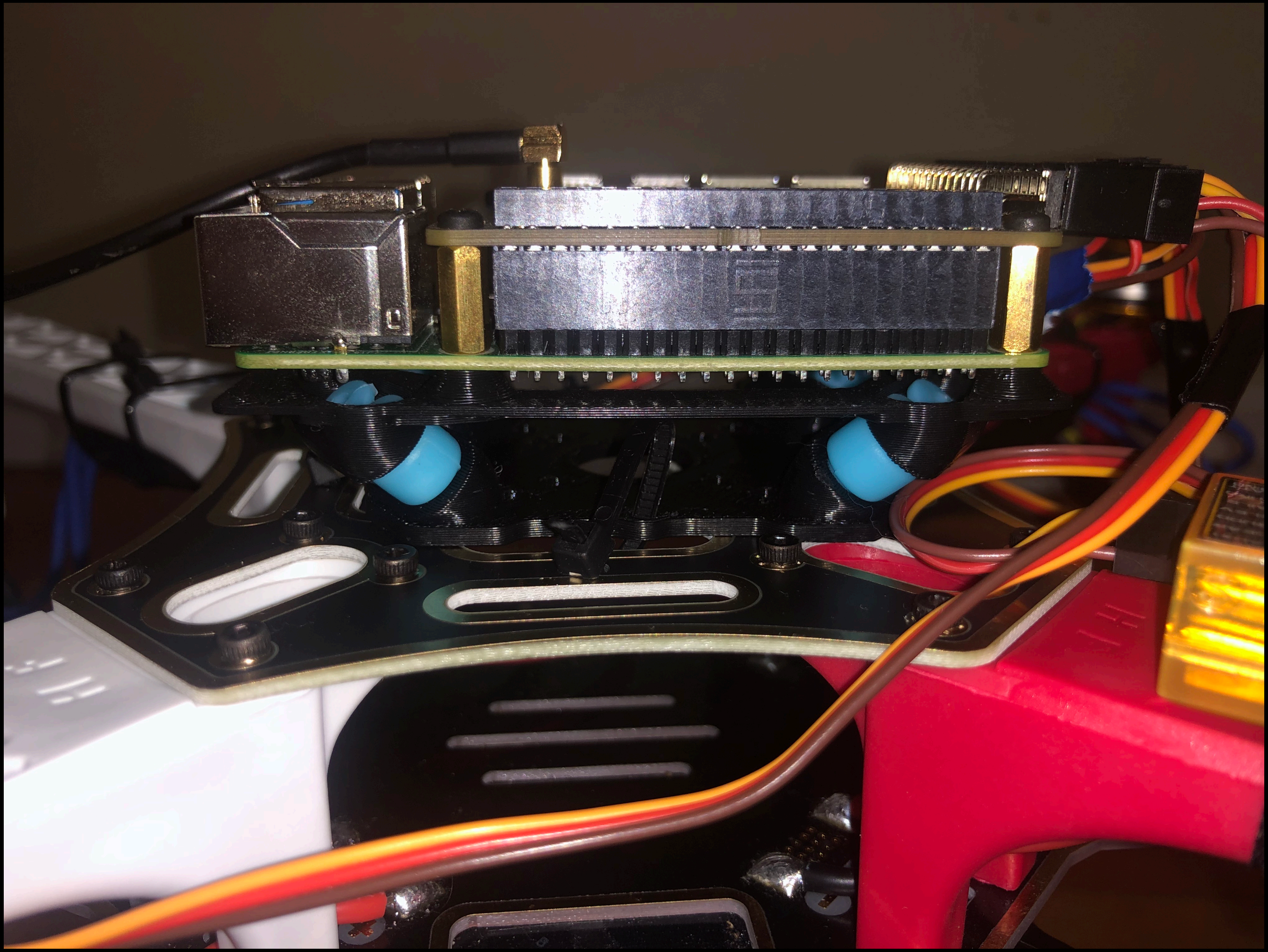








Vibration Dampening Mount



Without Vibration Mount

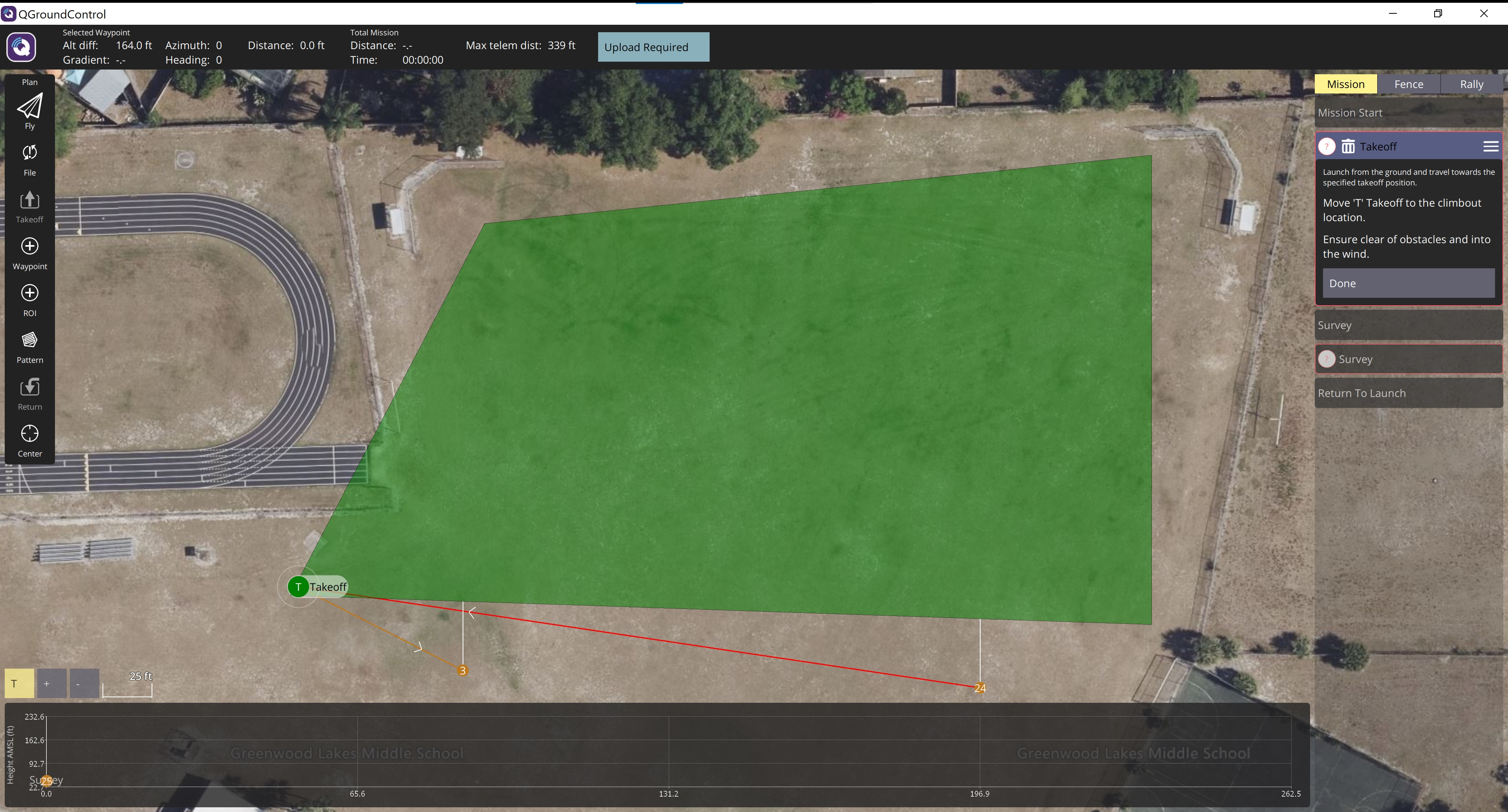
With Vibration Mount

Calibrations Check

Requirement #2: Mission

Requirement #2: Mission	
2.1: The drone should follow a flight path from takeoff to landing autonomously	NA
2.2: The drone should return to launch after delivering the package	NA
2.3: The drone will disarm motors after landing	~5sec
2.4: Drone must be able to be manually overridden when encountering critical failure	~1sec

QGroundControl



QGroundControl

Selected Waypoint

Alt diff: 164.0 ft
Gradient: -.-
Azimuth: 0
Heading: 0
Distance: 0.0 ft

Total Mission

Distance: -.-
Time: 00:00:00

Max telem dist: 339 ft

Upload Required

Mission Start

Takeoff

Launch from the ground and travel towards the specified takeoff position.

Move 'T' Takeoff to the climbout location.

Ensure clear of obstacles and into the wind.

Done

Survey

Survey

Return To Launch

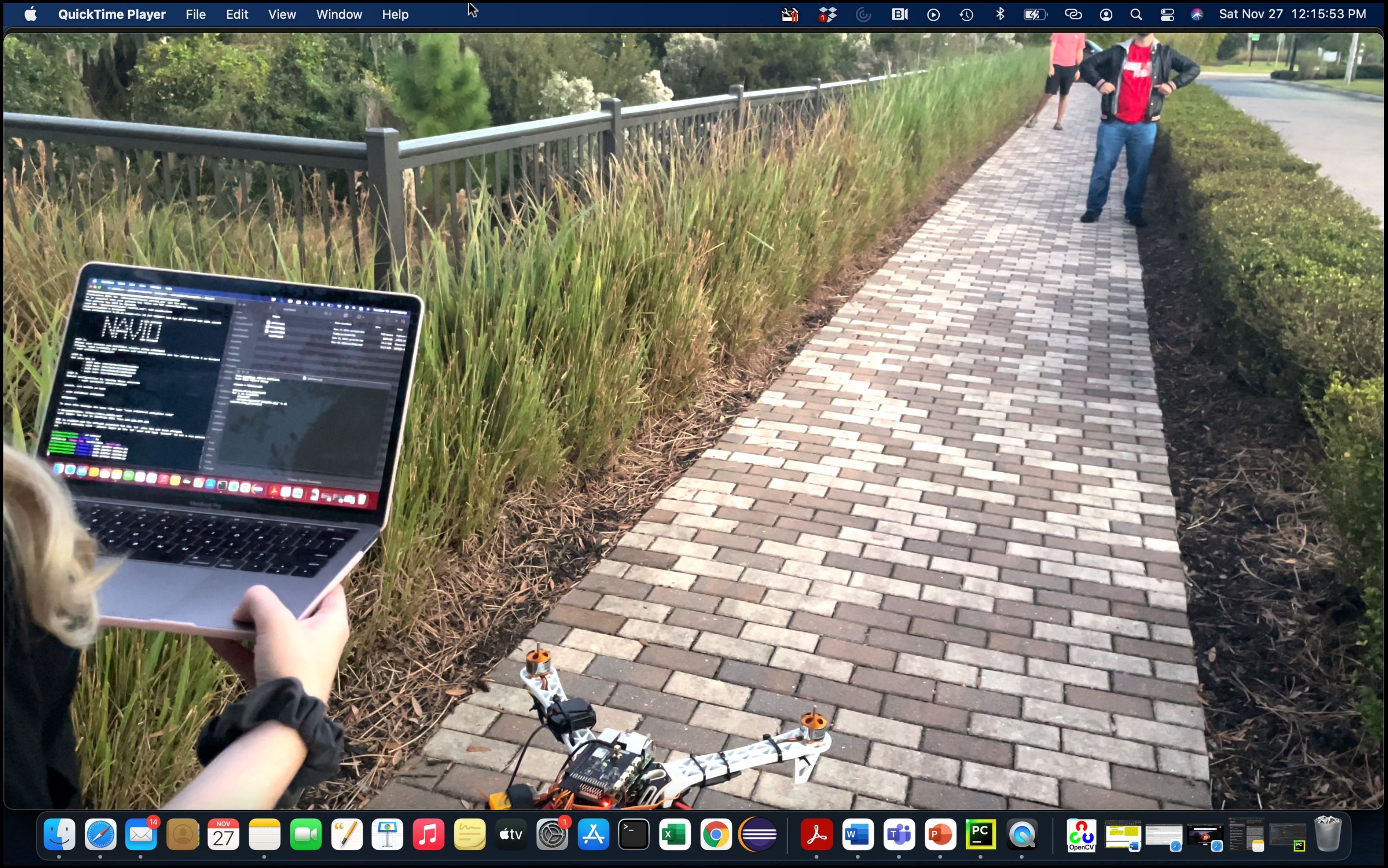
Crash #1

Video of Override

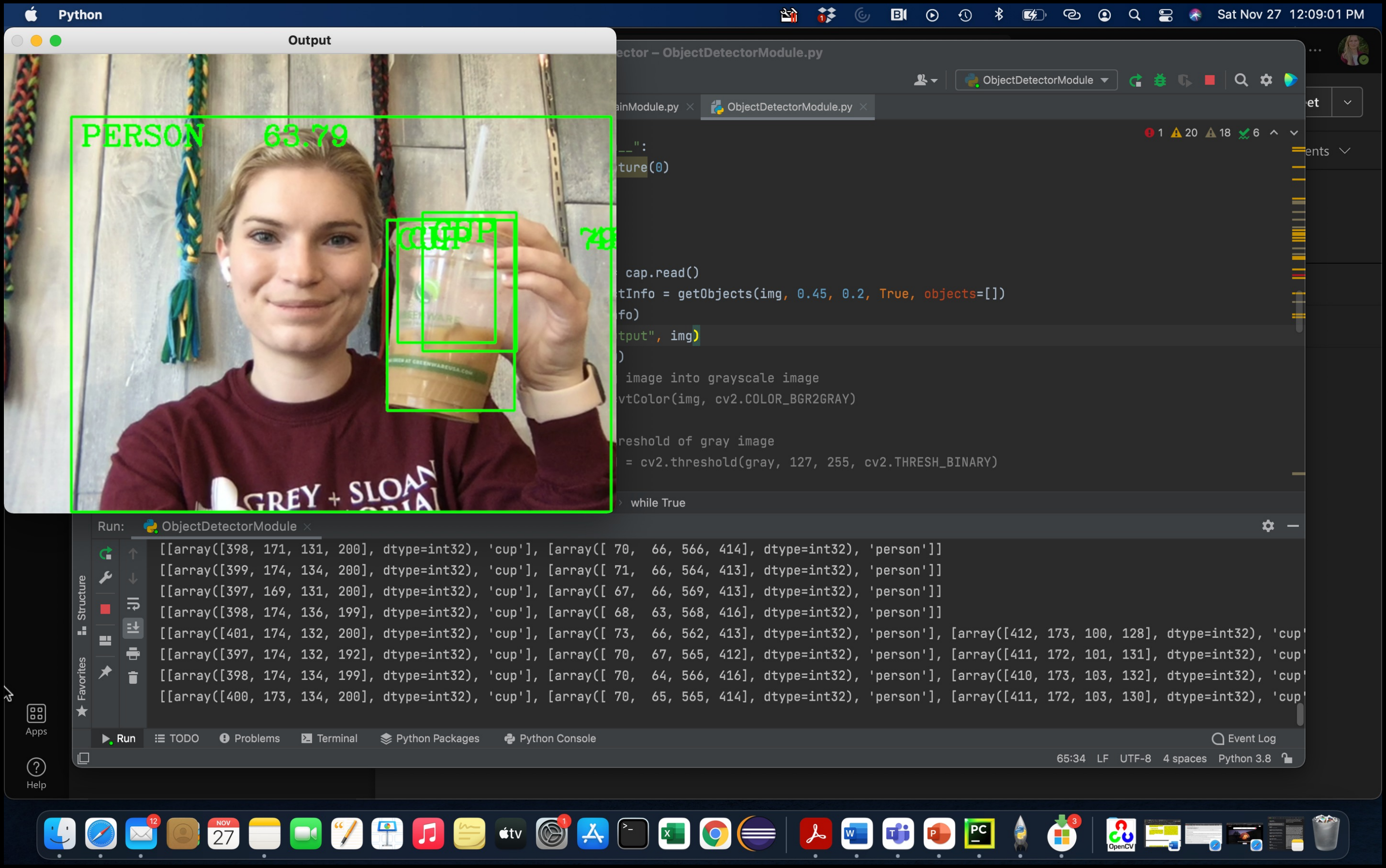
Flight Demo

Landing Demo

Camera


QuickTime Player
File
Edit
View
Window
Help
Sat Nov 27 12:15:53 PM
NAVIO

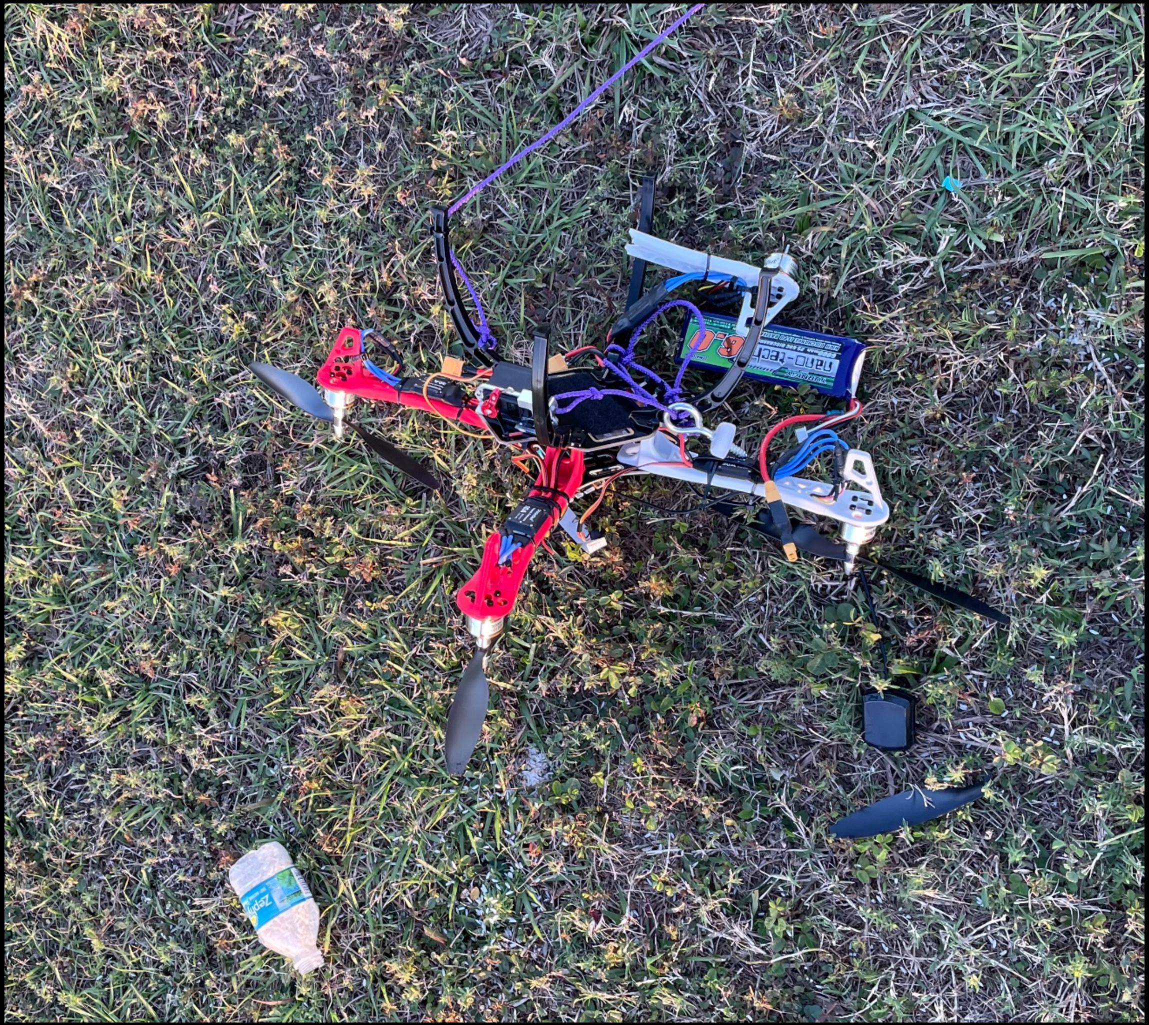
Recognizing Landing Zone - OpenCV



```
__":
ture(0)



 cap.read()
tInfo = getObjects(img, 0.45, 0.2, True, objects=[])
fo)
tput", img)
)
 image into grayscale image
vtColor(img, cv2.COLOR_BGR2GRAY)

reshold of gray image
 = cv2.threshold(gray, 127, 255, cv2.THRESH_BINARY)

while True
```

```
[[array([398, 171, 131, 200], dtype=int32), 'cup'], [array([ 70,  66, 566, 414], dtype=int32), 'person']]
[[array([399, 174, 134, 200], dtype=int32), 'cup'], [array([ 71,  66, 564, 413], dtype=int32), 'person']]
[[array([397, 169, 131, 200], dtype=int32), 'cup'], [array([ 67,  66, 569, 413], dtype=int32), 'person']]
[[array([398, 174, 136, 199], dtype=int32), 'cup'], [array([ 68,  63, 568, 416], dtype=int32), 'person']]
[[array([401, 174, 132, 200], dtype=int32), 'cup'], [array([ 73,  66, 562, 413], dtype=int32), 'person'], [array([412, 173, 100, 128], dtype=int32), 'cup'
[[array([397, 174, 132, 192], dtype=int32), 'cup'], [array([ 70,  67, 565, 412], dtype=int32), 'person'], [array([411, 172, 101, 131], dtype=int32), 'cup'
[[array([398, 174, 134, 199], dtype=int32), 'cup'], [array([ 70,  64, 566, 416], dtype=int32), 'person'], [array([410, 173, 103, 132], dtype=int32), 'cup'
[[array([400, 173, 134, 200], dtype=int32), 'cup'], [array([ 70,  65, 565, 414], dtype=int32), 'person'], [array([411, 172, 103, 130], dtype=int32), 'cup'
```

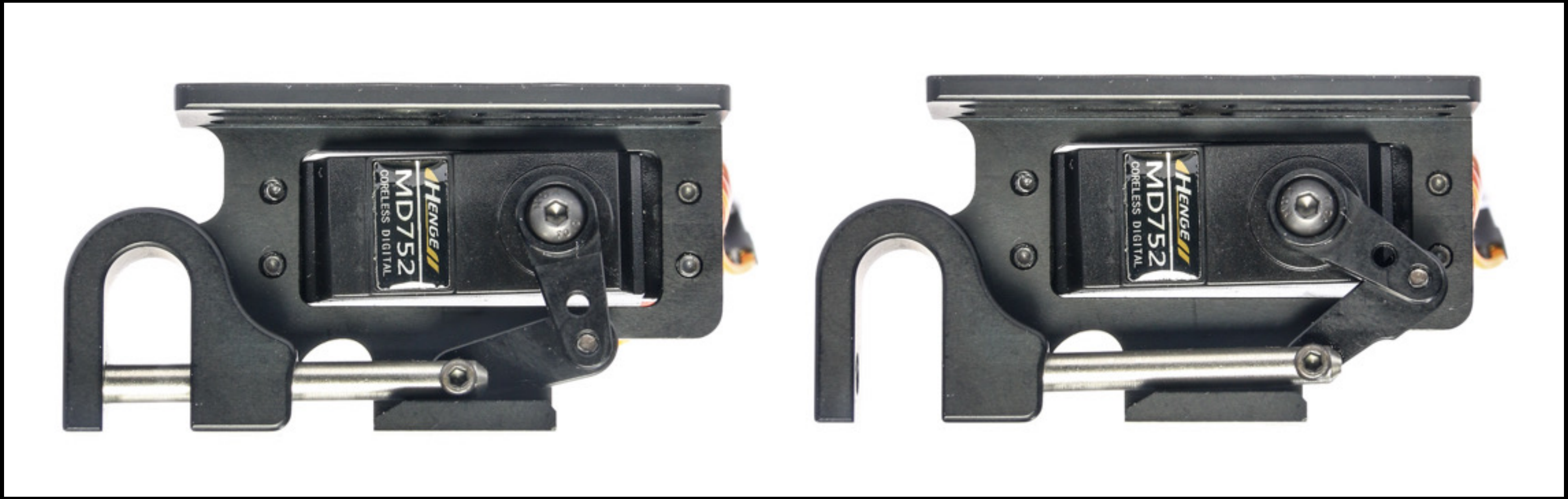
Crash #2



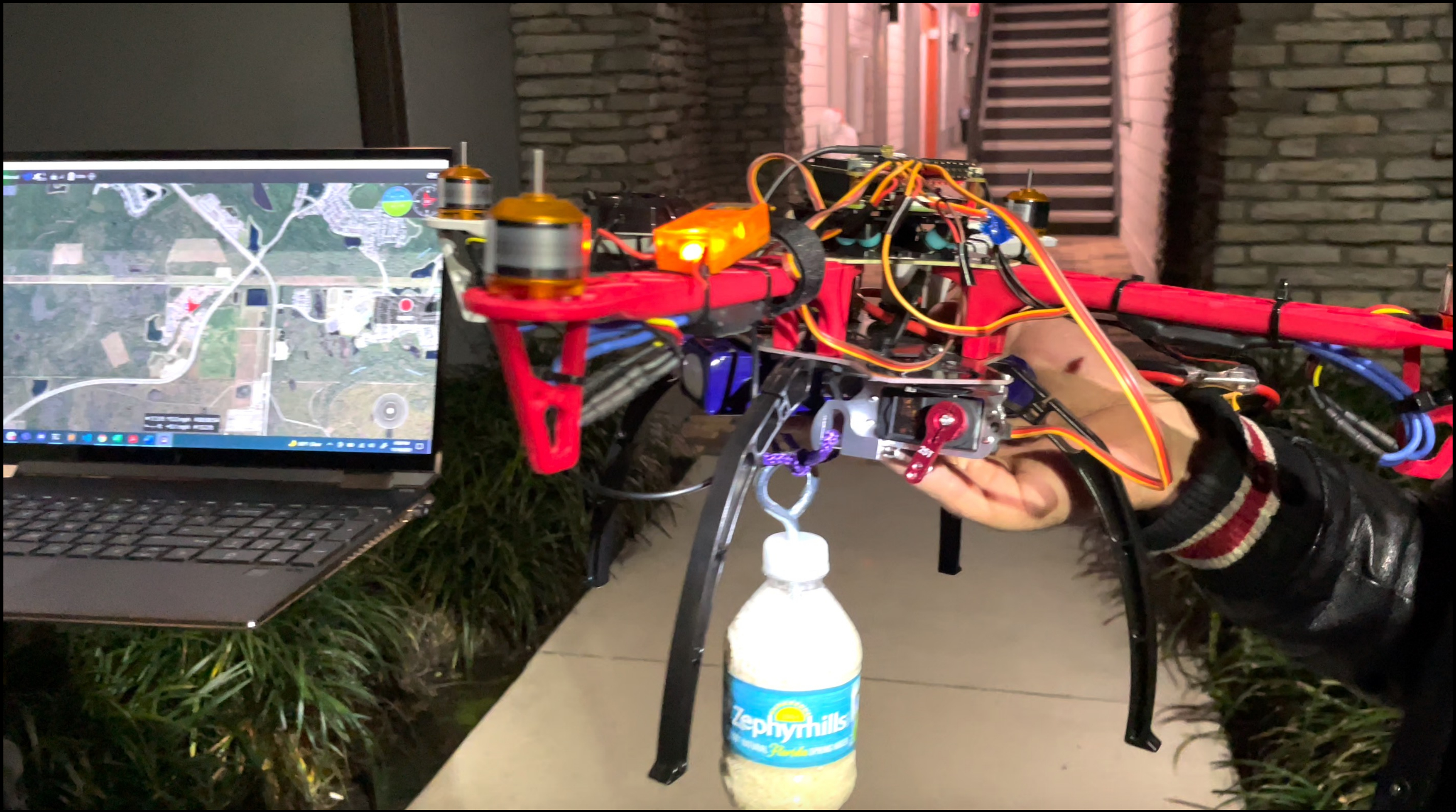
Requirement #3: Payload

Requirement #3: Payload	
3.1: The drone should be able to carry a 0.5lb payload	0.5 Pounds
3.2: The drone should be able to safely drop a payload in the delivery zone	Within 10 Feet
3.3: Drones servos should be able to hold and release package on command	NA

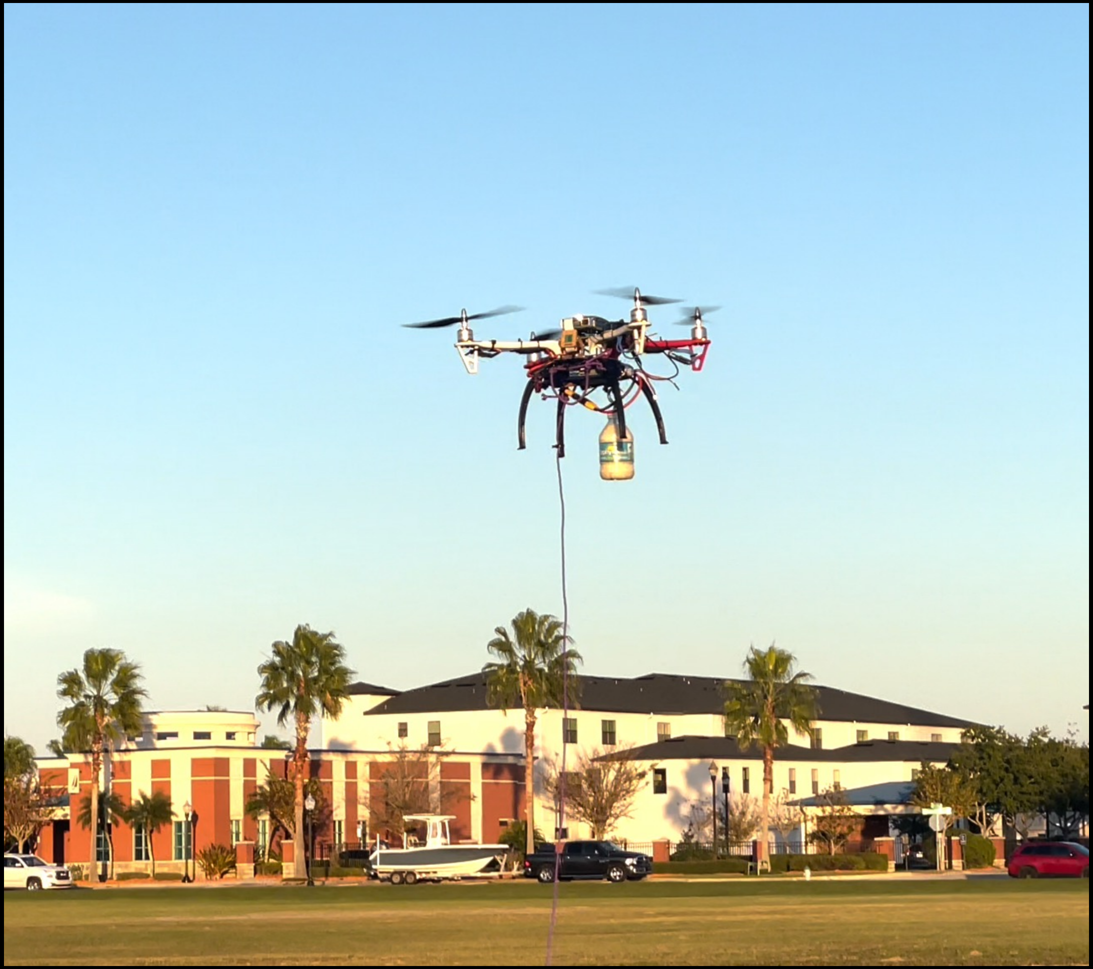
Package Release Mechanism



Package Release Mechanism



½ Pound Payload



Challenges

- Hardware
  - Components Available
  - Manufactures
  - Shipping Time
  - Rebuilding Drone
- Software
  - Setting up development environment
  - Communication between components and and code
- Overall Challenge
  - Self-Funded
  - Crashes
  - Weather

Lessons Learned

- Hardware
  - Fusion 360
  - EasyEDA/JLCPCB
  - Battery Protection
- Software
  - Flight Firmware/Software
  - QGroundControl
  - Computer Vision
- Overall Lessons
  - Attention to detail
  - Collaboration
  - Adapting

Thank you!

Q&A