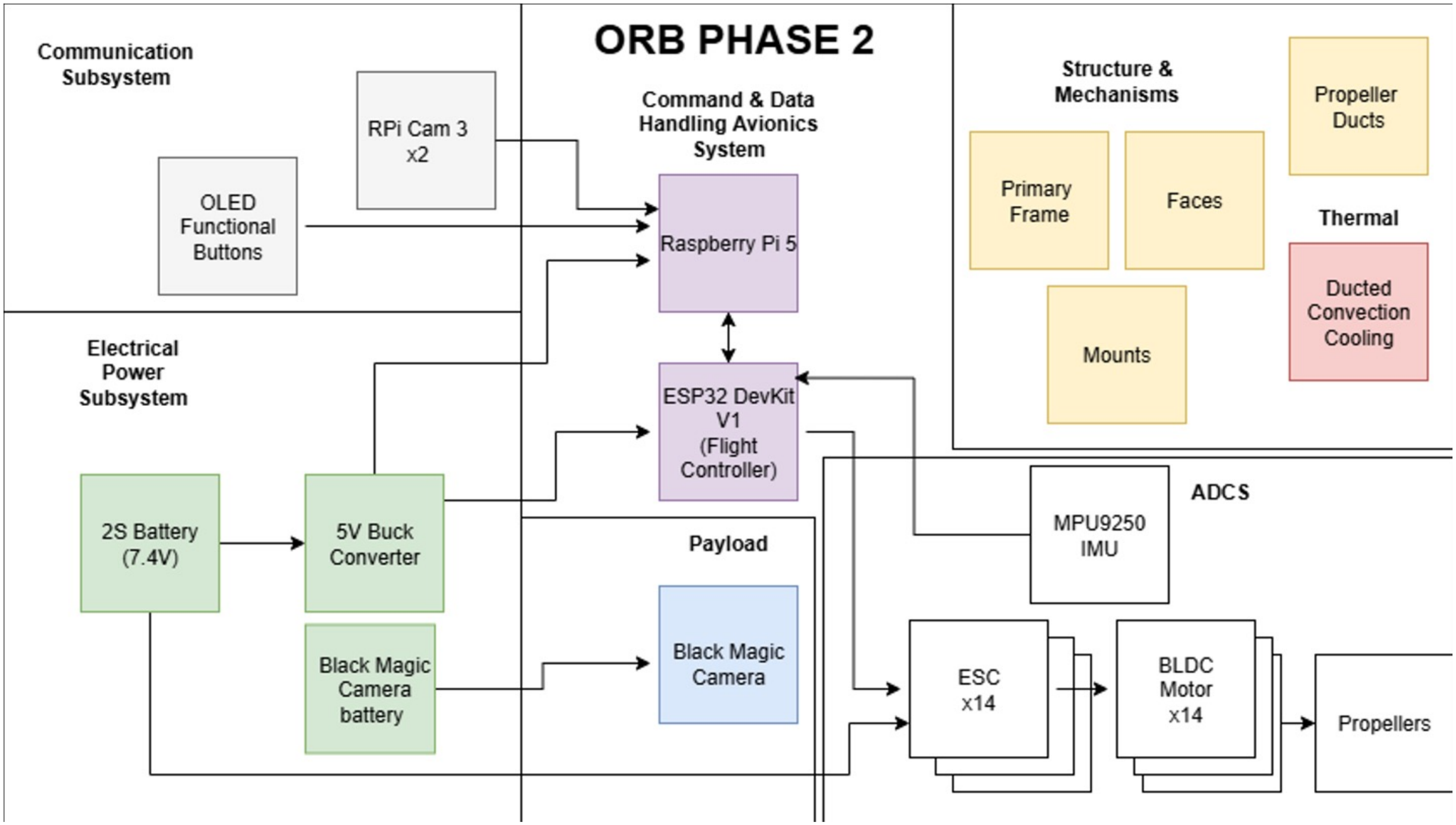
ORB Phase 2 Design

Functional Systems Block Diagram



Structures Design

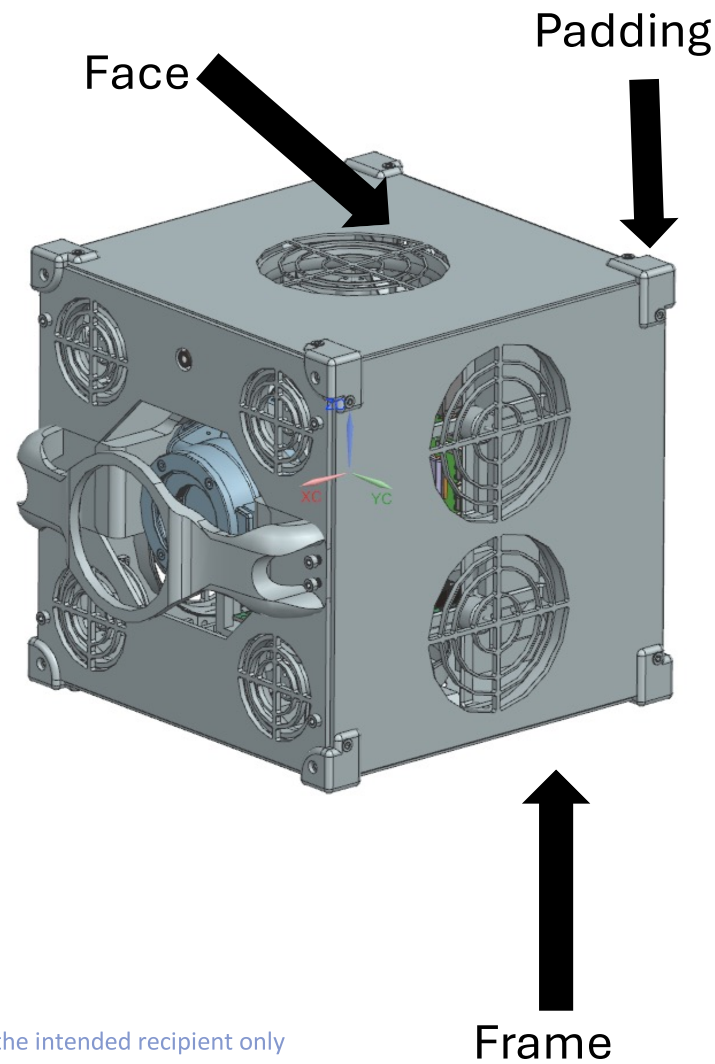
Lukas & John

Key Driving Requirements

KDR	Description	Reasoning
1 (John)	Provide secure mounting for internal components	Prevent displacement of internal components during maneuvering and standard operation vibrations
2 (Lukas)	Do no harm to people on flight	Unit must not be a hazard to those in the testing environment
3 (John)	Structural stability, avoiding jittering due to vibration or motion	High quality film requires as stable of a shot as possible
4 (Lukas)	Minimal mass & volume (size)	Low mass allows for more rapid maneuvers, small size mitigates moment of inertia

Structure Overview & Sizing

- 19x19x19 cm (18x18x18 without padding), frame is 1 cm thick.
- Openings in faces in place of propeller ducts & shielding
- Corner + edge padding to minimize damage of unit and environment in case of collision
- Cubic structure chosen for ease of manufacturing, however future iterations may be better optimized with spherical shell
- Large volume enables calculated placement of internal components to place CoM in line with camera



Frame and Face Design

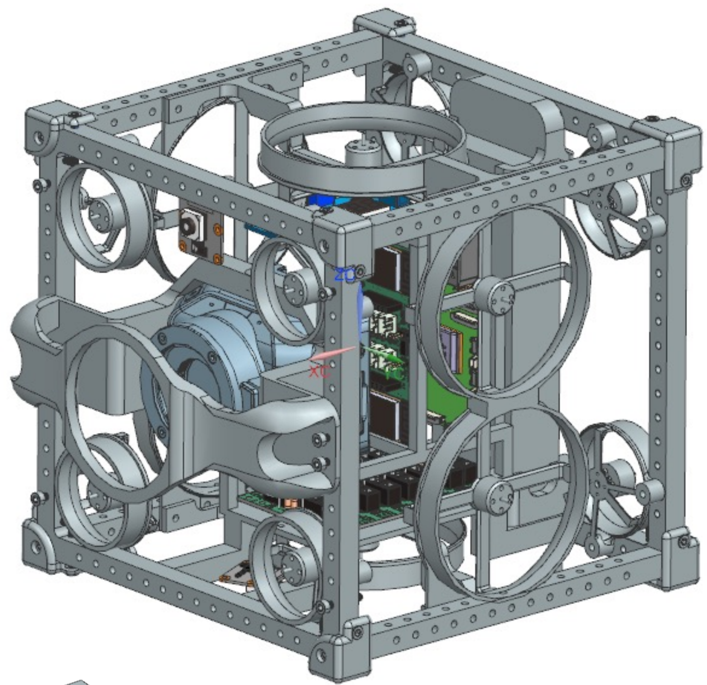
2 Frame Bars create front and back ORB structure, connected at the corners by 4 frame bars that span width.

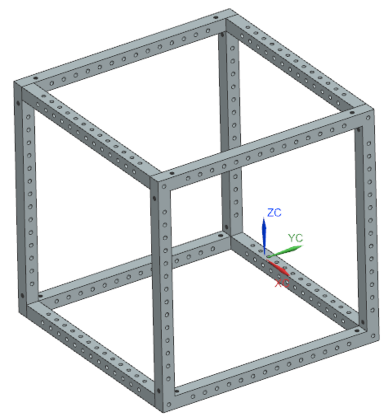
Enables straightforward assembly & manufacturing.

Chosen to reduce material costs and weight while ensuring structural integrity.

Fasteners secure Frames to each other and to the Faces, all standardized M3 bolts.

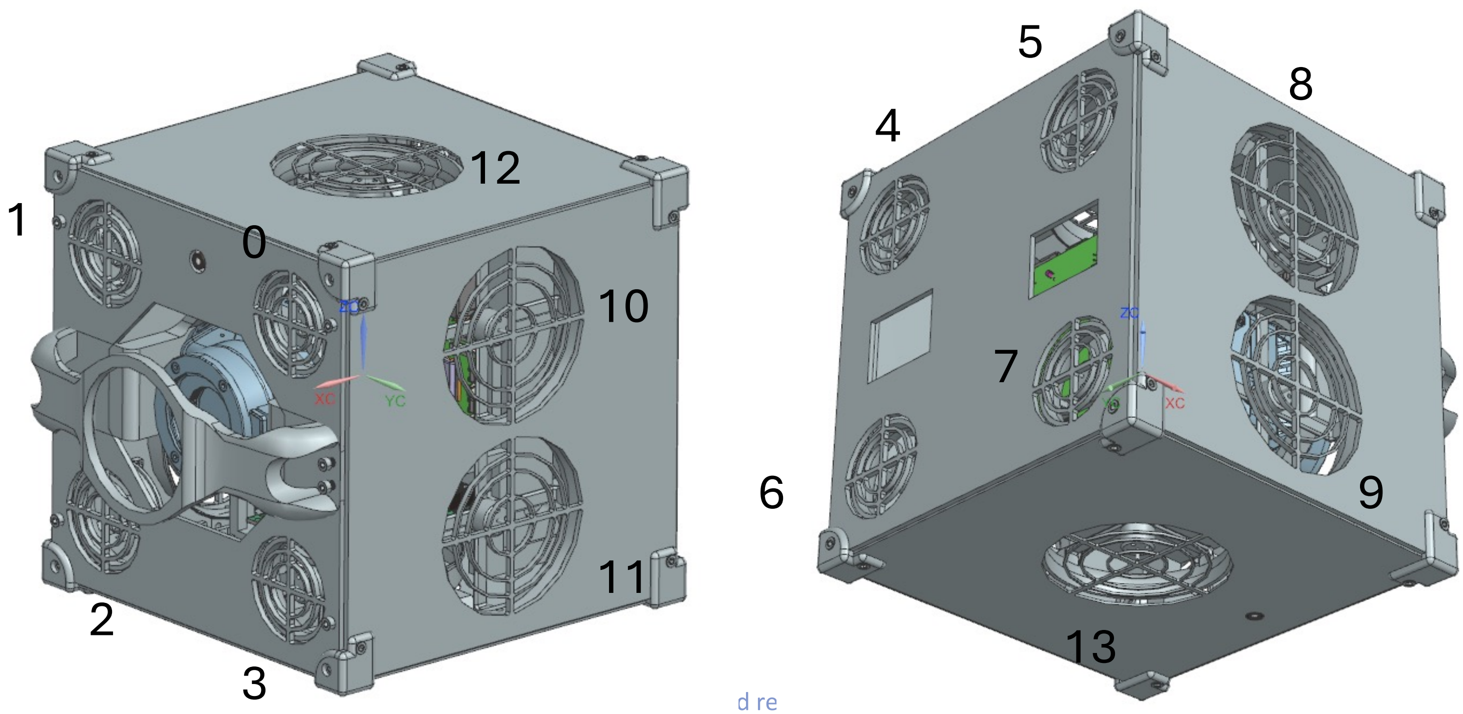
6 Faces have hole cuts for Propeller mounting, Black Magic Camera, and I/O Ports for Avionics & Data





Fan Placement Diagram

- 14 fans on flight vehicle, labeled from 0 to 13.
- This was done to simplify nomenclature and determine which fans are going to be activated on the motor controller PCB
- Key distinction: 40mm propellers are Front and Back, 76.2mm propellers are Top, Bottom, Left, and Right faces



Measured Mass Overview

Subsystem	Mass [kg]
Avionics/CDH	0.171
GNC	0.0412
Propulsion	0.4457
Structure	0.7245
Power	0.6600
Thermal	0.1200
Payload	0.5590
Total Measured Mass (CAD Assembly)	**2.7223**

Material Selection

- Usage of Ansys Granta to identify materials for Phase 2 and possibly beyond
- Material Indices set by comparison of Constraint to Objective

Function	Constraint	Objective	Index	Material
Face	Cost	Stiffness	E vs. $	Polycarbonate
Frame	Density	Strength	σ vs. ρ	Delrin (POM)
Mounts	Cost	Thermal Conductivity	K vs. $	ABS
Padding	Density	Deformable	E vs. ρ	TPU

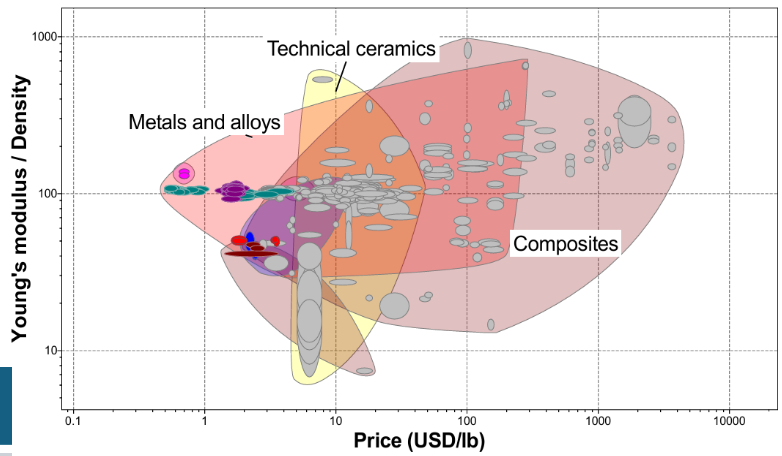
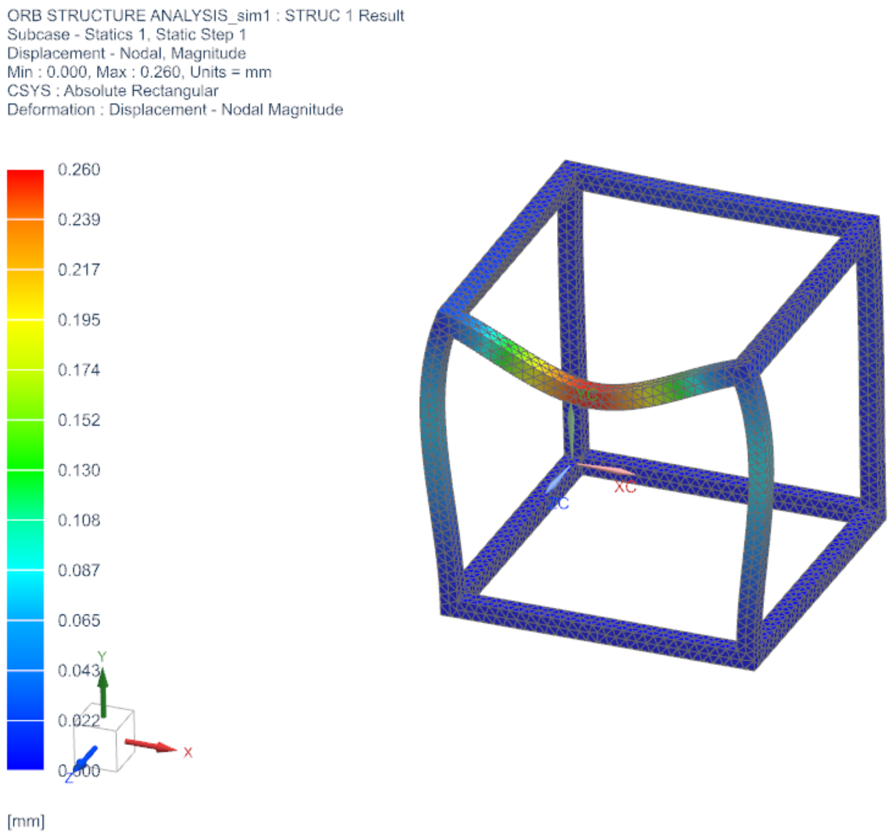


Figure 31.1: Example Material Chart indexing $ vs. E/ρ

Structural Analysis – Primary Structure

- Structure makes up 44% of total mass
- Assume maximum load case of 2Gs (based on Zero-G flight), applied as a compressive load with the ORB fixed on the bottom face
- Under load of anticipated collision at the middle of a connecting beam that causes a 1.4 m/s change (colliding and rebounding from max operating velocity) in 0.5 seconds
  - 3.13 kg x (1.4 m/s / 0.5 s) = 8.96 N

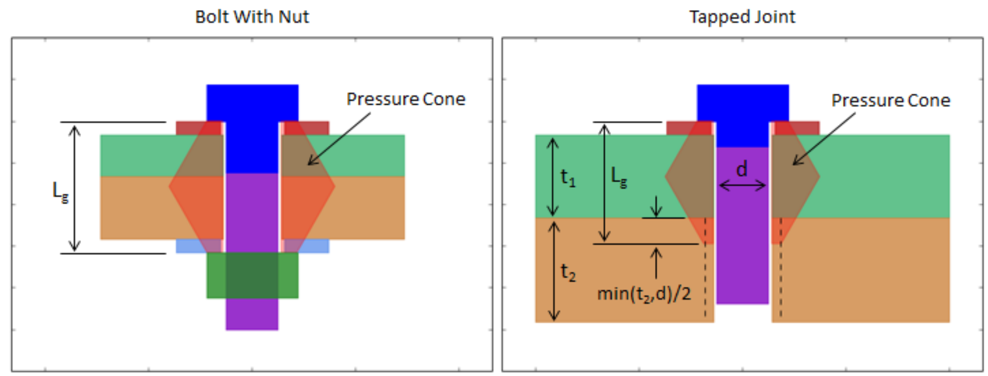
- **Verified Hand Calculations**
- Maximum Beam Deflection with Anticipated Load: 0.108mm
- Maximum Beam Deflection with Max Load: 0.238mm
- Stress with Anticipated Load: 4.84 MPa
- Stress with Max Load: 10.6 MPa
- Significantly lower than yield strength of Delrin (POM) at 65 MPa (5.13 calculated Margin of Safety)



Fasteners & Bolted Joint Analysis

- Holes in Delrin Frame are either tapped or meant to be a through hole for M3 fastener to connect a component
- Preload assumptions, should be torqued to 67% amount of yield strength of Delrin
- Varying fastener lengths throughout ORB, bolts for mounts are 12mm while bolts for the frame are 18mm
- Does not count bolts found in avionics boards or motors

Fastener Component	Count
Faces	24
Frame	8
Mounts & Camera	28
Total	60



Torque for preload = 0.15 Nm in Delrin material (conservative answer for FOS)

$$T = KFd$$

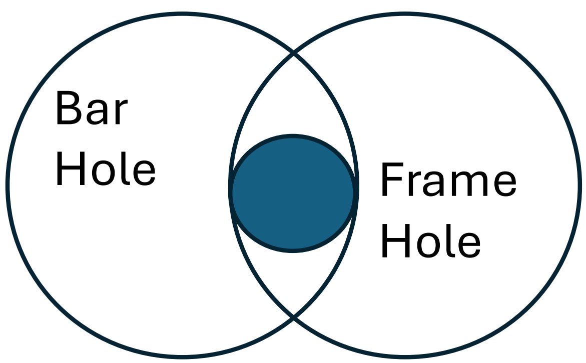
$$A_{shear} = \pi d_p L \eta$$

$$F_{strip} = \tau \cdot A_{shear}$$

$$F \leq F_{strip}$$

Tolerance Stack Up (Bar to Frame)

- Fasteners tend to be interference fits because 3D printed components are utilized, thus tapping the holes when they are mounted onto different points along flight vehicle frame.
- Assume Front and Back frame are global datum schemes determined to be 0,0
- Hole diameter set with tolerance ± 0.1mm and True Position of ± 0.25mm
- Hole Alignment is more essential to having a properly secured, reproducible ORB rather than being slightly larger than the 19x19x19cm size.



Fastener (exaggerated for visualization)

Tolerance Stack Up for Fastener Holes			
If Margin is positive, then there is clearance			
Margin	0.15		
Dia 1 (MMC)	2.9		
Dia 2 (MMC)	3.5		
TP 1	0		
TP 2	0.25		
Fastener	3	(2x)	

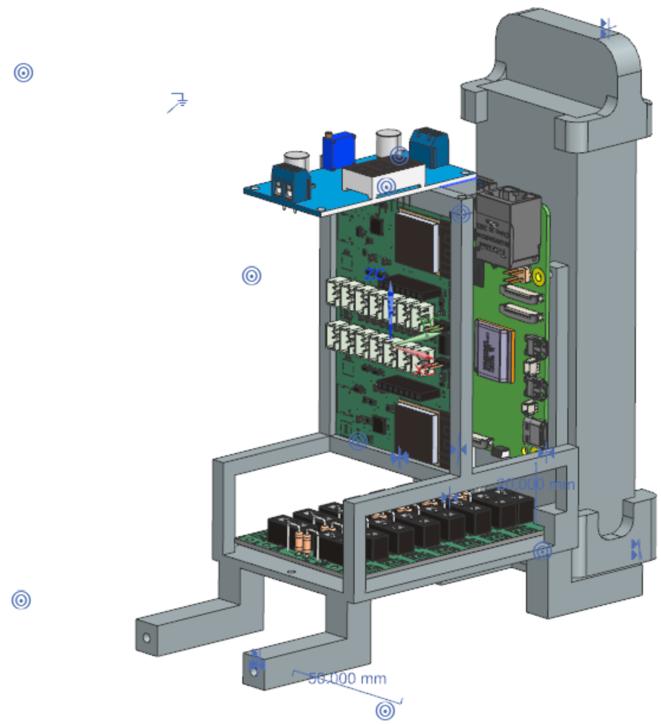
Avionics Stack - Secondary Structure

- Stack consists of: Flight Computer (RPi5), ESP Board, Power Distribution Board, Buck Converter

- Configuration maximizes space for inside ORB for cooling surface area and cable harnessing connections

- Placed in line with the X axis of the flight vehicle for greater stability, alongside Black Magic Camera and Battery

- Potential modification for thermal considerations and optimized structure



Next Steps

- ORB Phase 2 is fully assembled, no more modifications structurally for this version
- ORB Phase 2 Version 3 will have modifications for the Zero-G flight
  - Polycarbonate Faces
  - Resin Printed Propellers & Ducts
  - New Avionics Secondary Structure with better sizing for wiring and PCBs

Thermal & Modal Analyses

Lukas & John

Key Driving Requirements

KDR	Description	Reasoning
1 (John)	Include cooling fan to avionics & promote interior convection	Prevent overheating of boards and power supply
2 (John)	Reinforce mounting local to high heat components	Prevent deformation/failure of plastic mounting
3 (Lukas)	Design path(s) for heat extraction	Allow airflow through structure [vital for propulsion]
4 (Lukas)	Isolate high heat components	Avoid regional overheating by spacing out thermally concerning components
5 (Lukas & John)	Reduce noise disturbance generated by propulsion system	Avoid discomfort for staff/customers on Zero-G flight interacting with ORB

Thermal & Modal Overview

- Primary flight computer (Raspberry Pi 5) expected to reach temperatures up to 82° C w/o fan
- Primary power source (7200 mAh 7.4 Lithium-Polymer Battery) expected to reach up to 50 – 60 ° C (under investigation)
- ORB Phase 2 MVP uses Delrin and Polycarbonate with approximate thermal conductivity values of 0.33 and 0.19 respectively. These insulation properties do not account for the convection cooling done by the propellers
- Create theoretical results for natural frequency of ORB along XYZ and different applied loads.
- Compare results for [TBD] experimental vibration test

Thermal Components Temperatures

Component	Idle Temperature [°C]	Max Measured Nominal Temp [°C]	Max Operating Temperature [°C]	Has a cooling component?
Battery	30	50	70	No
Raspberry Pi	25	62*	85	Yes
Propeller & Motor	20	30	125	Yes
ESP32	30	40	85	No
Black Magic Camera	25	58	40	Yes
RPi Camera	25	50	85	No

*Utilizing updated cooling unit

Thermal Model Calculations

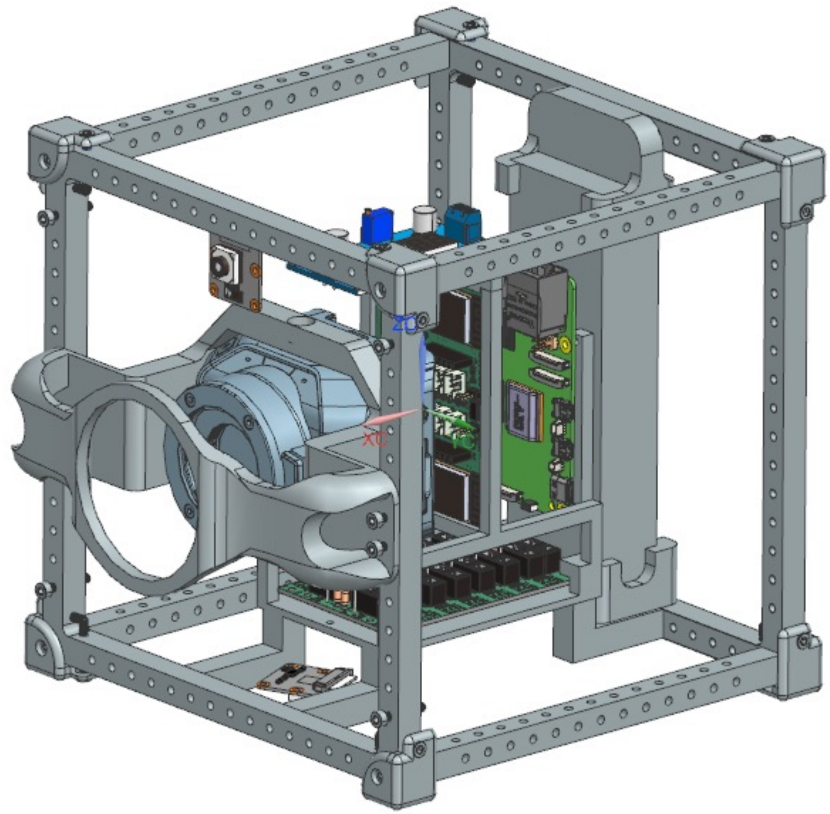
Governing Equations & Assumptions

Max Operational Temperature of Pi5 was set to 82*C

Assume that natural convection is only current method of cooling from ambient air

$$Q = hA(T_{board} - T_{air})$$

$$\Delta T_{air} \sim \frac{Q}{\rho c_p \frac{V}{t}}$$

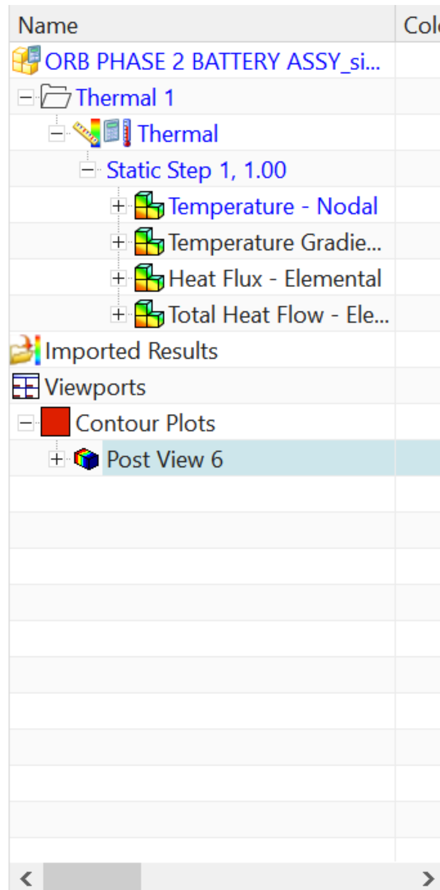


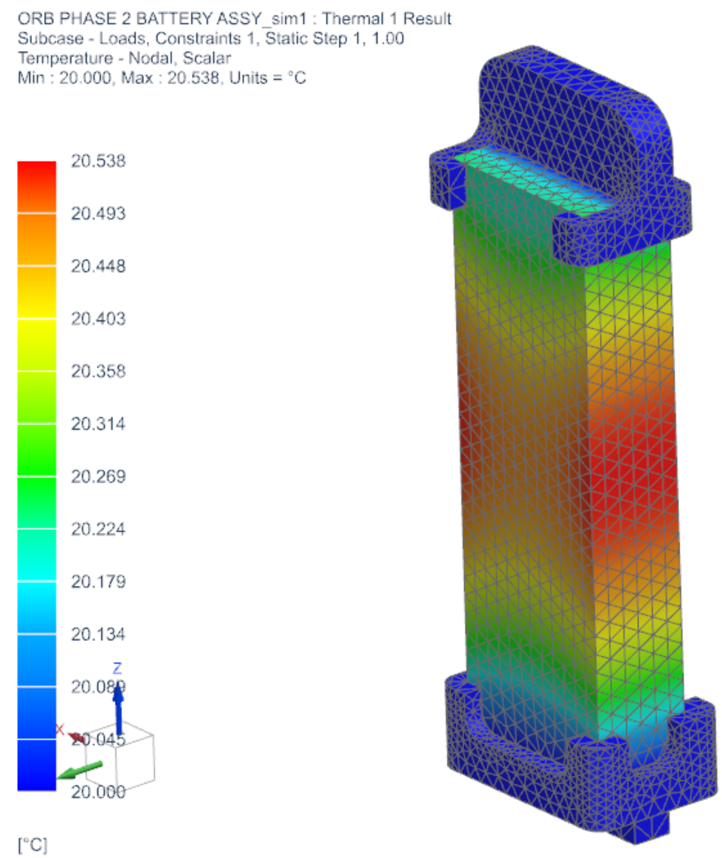
Thermo-structural – Battery

- Purpose: Full integration of thermal components that generate heat inside of the enclosed flight vehicle

- Approach: Using Sim center 3D SOL 153 Steady State heat transfer, assuming that the ORB is stationary and is set to an ambient temperature of 20*C (68F). Determine how ambient temperature is affected by max battery temperature

- Results:
  - Resultant Ambient temperature: 60.538*C
  - Compare with empirical test data for all components and determine main sources of heat.

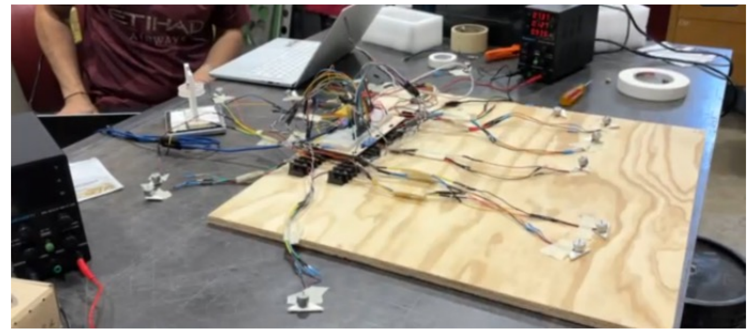


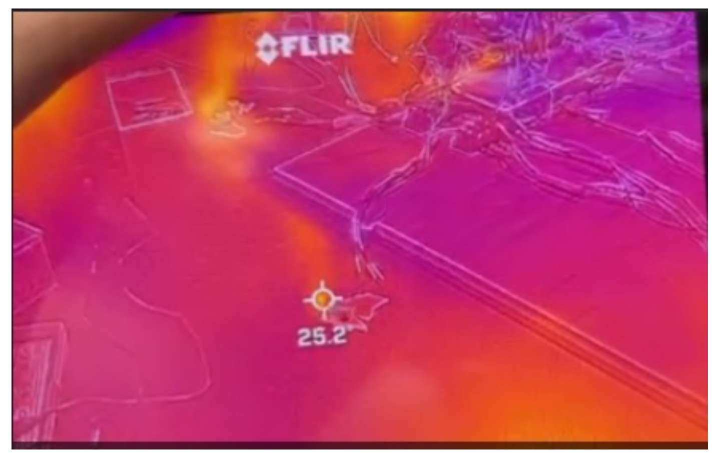


Testing (Motors + Avionics)

- Testing with previous motors pulsing from 0 – 100% throttle showed minimal thermal strain on the system
  - FLIR showed temperatures confined to under 35 °C
  - 14 motors, power distribution bus boards, 18 and 28 AWG wires, 2 ESP32 boards, 1-2 10 A power supplies

- Initial tests with Raspberry Pi and cameras showed up to 83°C from the Pi's internal monitor before shutdown
  - CPU determined to be the primary heat source, lowering concern for transfer to other hardware but still needing to be addressed

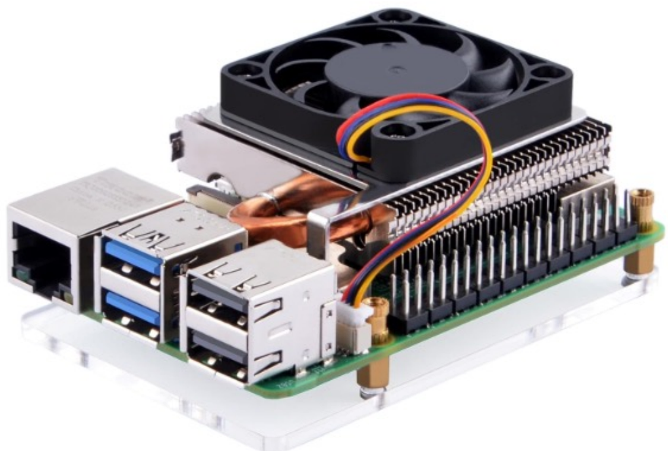




 |

Raspberry Pi Cooling Unit

- Aluminum heatsink
- Copper heat pipe
- Direct attachment to CPU
- Fan activates once Pi-measured temperature exceeds 60°C by default
  - Reprogramed to run constantly
- With constant fan, Pi-measured temperature maintains sub-62°C, thermocouple readings placed on the heatsink capped out at 33°C



Subsystem Thermal Report

Subsystem	Independently Operational?	Threat to Other Subsystems?	Threatened by Other Subsystems?
Avionics	Yes	Potentially	No
Power	Yes	No	Potentially
Propulsion	Yes	No	No
Structures	Yes	No	No

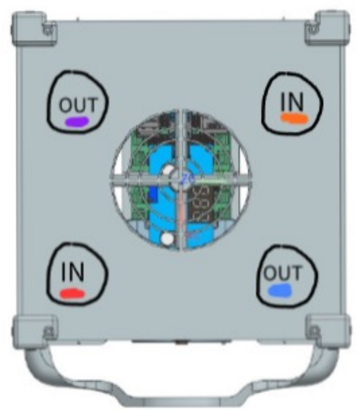
Thermal Next Steps

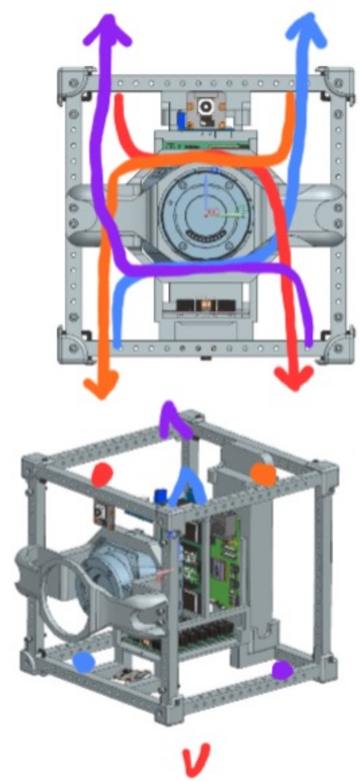
- Test fully integrated system to observe how avionics heat is transferred
- A quick simulation of all current avionics needs to be done in Simcenter
- If heat proves to be a concern, especially for the battery, a more robust cooling system may be required

Exploring options in the event active cooling is required for avionics

Multi-Fan Thermal Cooling Concept

- 4 small cooling fans in ducts
  - Two stacked on the upper portion of avionics stack/power supply
  - Two stacked on the lower portion of avionics stack/power supply
- Counter rotating, blowing oppositely to eliminate forces/torques
- Ducted with inlets and exhausts on top and bottom faces





Modal Assumptions & Calculations

- Assumptions: A free-flyer is unconstrained, so the analysis has no applied constraints =for the initial test, then it is considered fixed along its propellers since that is where force is being induced. For any secondary structures, the constraints are applied where they are fastened onto the primary structure

- Basic Hand Calculations based on knowns from previous mechanical properties to determine natural frequency
  - Bending Beam Modes:
  - Kirchhoff–Love classical

$$f_n = \frac{\beta^2{}_{mode}}{2\pi L^2}\sqrt{\frac{EI}{\rho A}}$$

$$f_n = \frac{\pi^2}{2\pi}\sqrt{\frac{D}{\rho h}}[\,(\frac{m}{a})^2 + (\frac{n}{b})^2]$$

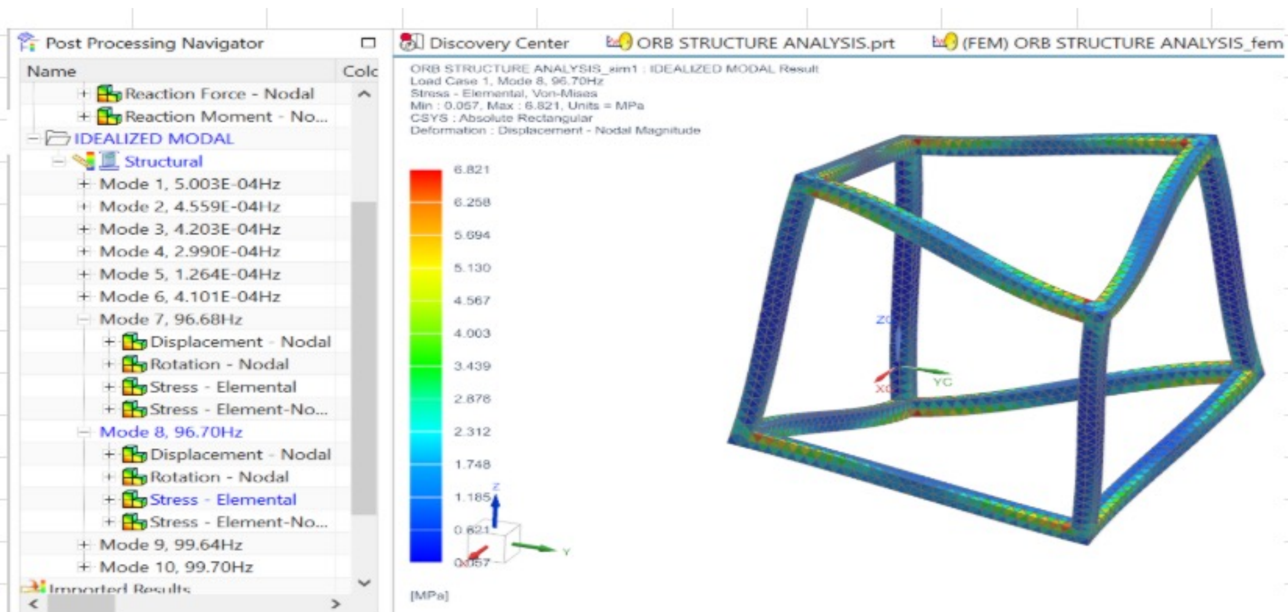
- Avionics Stack treated as vertical cantilever, might reinforce but vibration created by propeller needs to be studied
- Battery Mount is simply supported load beam mounted along vertical back face plate of the ORB
- Modes are generated due to inherent mass and stiffness of structure, no loads are applied
- **Next Steps for all mechanical analyses: Validate performance of flight vehicle at SERC Lab Test (or any testing campaign in this situation)**

Modal Analysis – Primary Structure

- Purpose: Understand characteristics of flight vehicle to identify natural frequencies along Mode 1 and Mode 2 . These values are useful for further vibrational & dynamic analysis when subject to different load cases

- Approach: Using Simcenter 3D SOL 103 Real Eigenvalues. Idealized body is set to Delrin (POM) material and unconstrained 6 DOF.

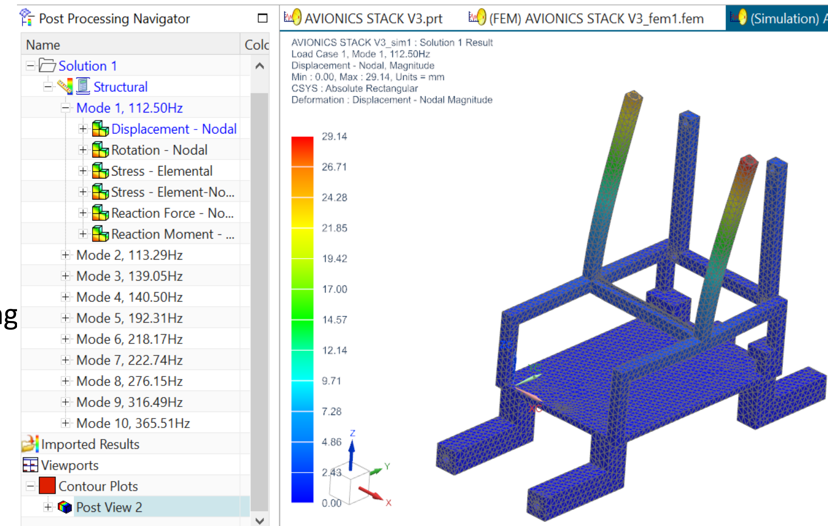
- Results:
  - Mode 1: 96.68 Hz
  - Mode 2: 96.70 Hz
  - Structure shows natural frequency to induce bending stress

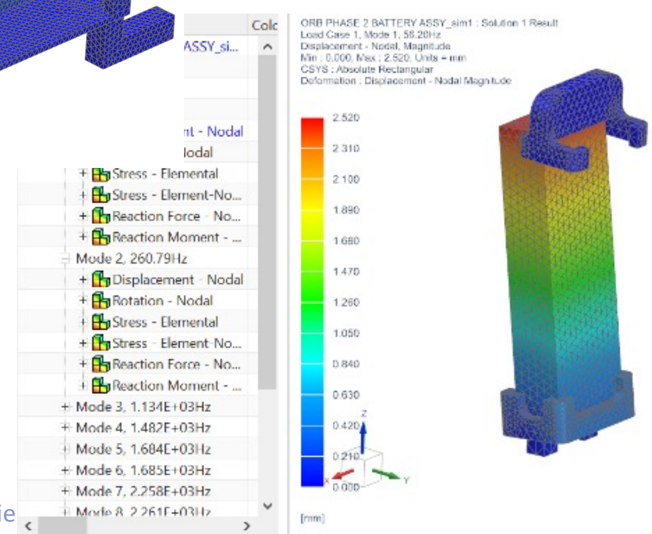


Modal Analysis – Secondary Structure (Avionics)

- Approach: Using Simcenter 3D SOL 103 Real Eigenvalues. Assumes that base of avionics stack is constrained at mounting points, thus shaking the beams to cause cantilevered motions.
- For the battery mount, the top and bottom 3D printed brackets are the fixed constraints, simplifying the analysis to a fixed beam

- Results:
  - Avionics Mount
    - Mode 1: 112.50 Hz
    - Mode 2: 113.29 Hz
  - Battery Mount
    - Mode 1: 56.20 Hz
    - Mode 2: 260.79 Hz



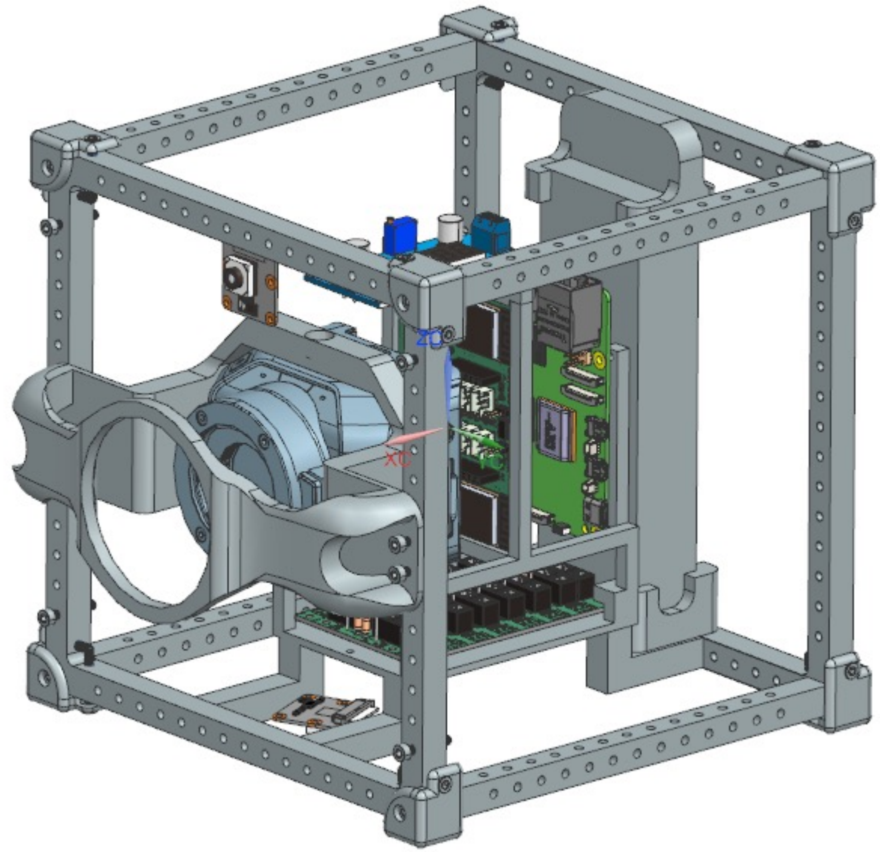


Modal Analysis – Full ORB Phase 2 (Attempt)

- Attempted to run simulation on entire ORB similar to Thermal analysis, however analysis kept failing. Issue might come from PCBs despite having simple geometry it might be too much to process entire stack

- Will try creating "dummy CAD" model approach to get a result out of idealized geometry of the PCBs.

- Conclusive study needs to be done on entire stack for comparison during a workmanship vibration test



Avionics & CDH Subsystem & GNC

Andrew, Simon, Gavin, Walid

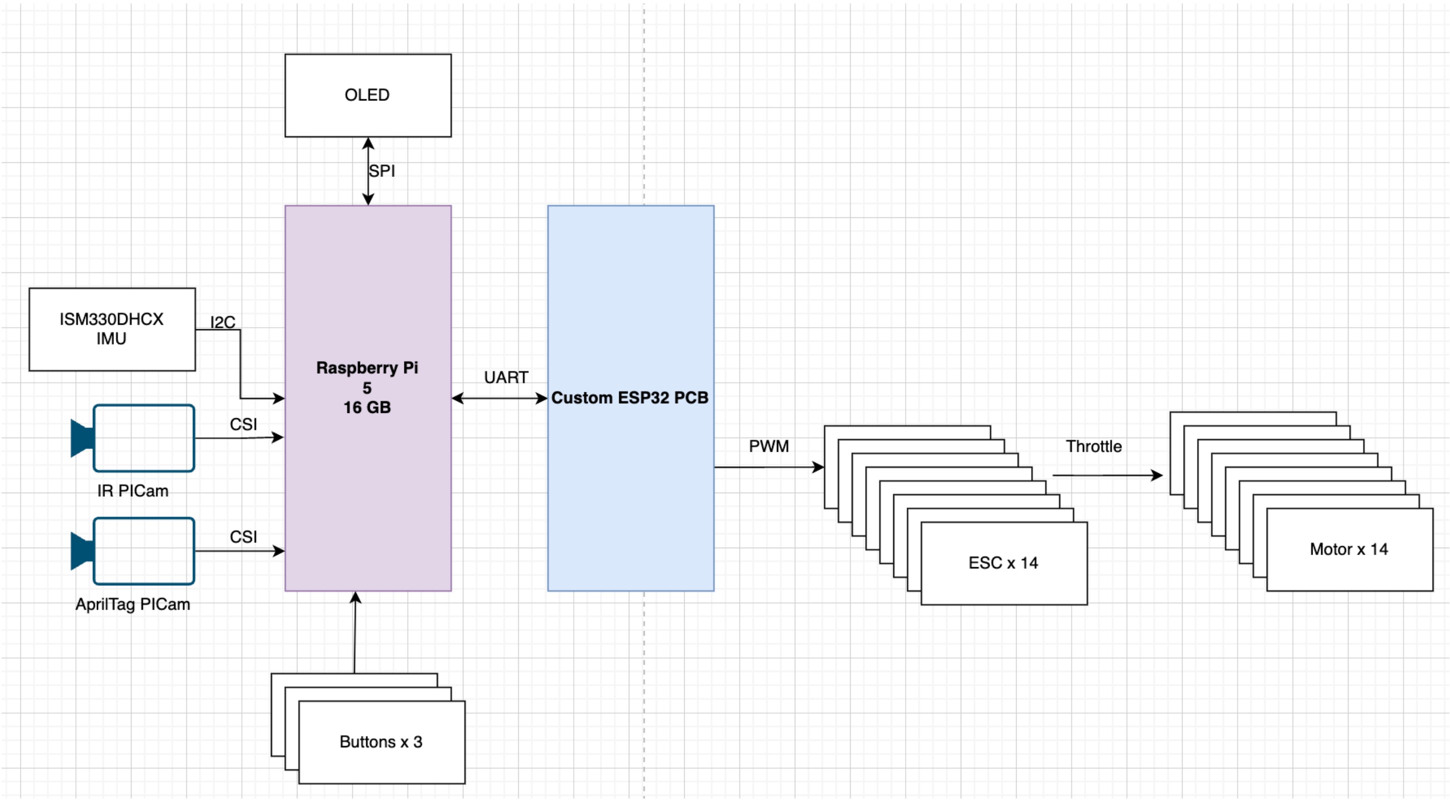
Key Driving Requirements

KDR	Description	Reasoning
1 (Gavin)	All Axis Response shall have 5° Steady State Error with 1°/s drift rate	Provide acceptable pointing accuracy for film-making
2 (Walid)	Reaction time should be 200ms or less	Providing smooth and constant reactions
3 (Walid)	Have redundancies for all major and minor control system and avionics	For safety and worst case scenario

Avionics Overview

Sub-Systems	Hardware	Notes
Compute	Raspberry Pi 5 (16GB)	Runs tracking algorithm (OpenCV) + camera processing and sending commands
Motor control	Custom ESP32 Dev PCB	Receives commands from RPi, Close Loop Control, and controls ESCs
Sensing	2* Pi Cam v3 (CSI), IMU (xt), OLED and buttons	Tracks IR bracelet and AprilTag, provides orientation, OLED and buttons tell the program what to run
Communication	UART I2C SPI	RPi ↔ ESP32 command pipeline IMU --> OLED RPi -->RPi Data flow

Avionics Overview



[Full System Wiring Diagram]

Communication Protocol

- We have four main communication protocols running simultaneously at any time

UART, I2C, SPI and CSI

UART is used between the RPI and the 2 esp32 chips, we run a bus every 100ms that sends data back and forth

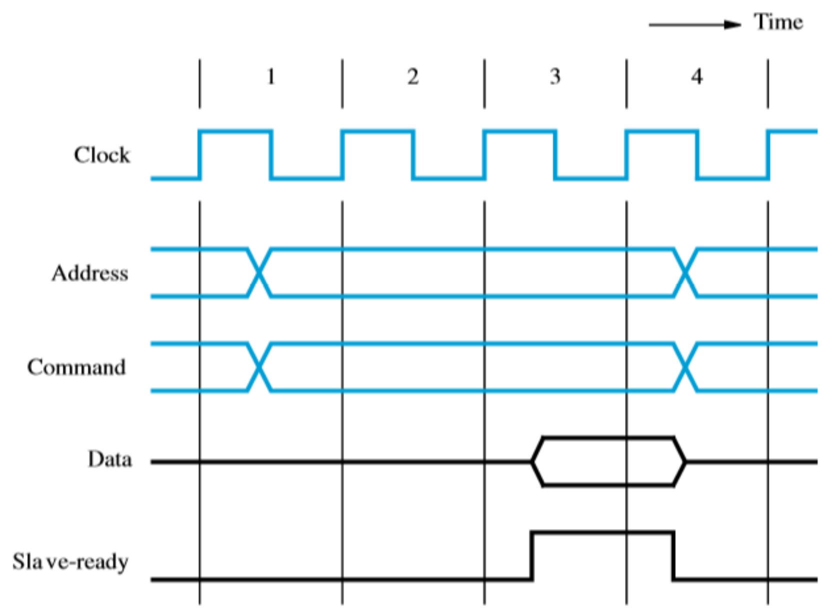
I2C and SPI work on a polling basis so whenever we have new information they send it directly to the RPI or vise versa

CSI works on a 10ms timer taking a picture every 10ms and sending it the RPi for processing

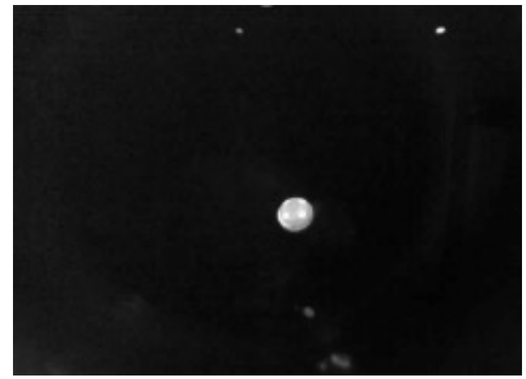
Now clearly this gives us 5 different clocks to work with 6 including the RPi, to work around this we (with help from ed) came up with a bus system that shuttles the information every 100ms, we will see in the a later slide a more clear example

Bus System

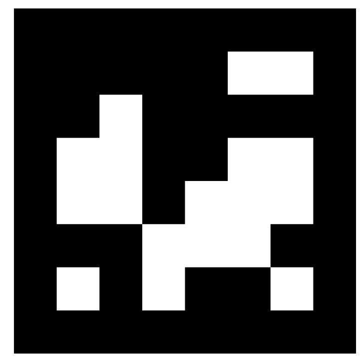
- On average each full cycle from reading the information to execution is running on around 0.1s or 100 ms
- This number can be adjusted, the fastest that we can run the program is at 35 ms however that caused some stall/old info to be sent which we didn't want so to combat this we created a bus system that would steal the latest info every 100ms



Camera-based tracking system

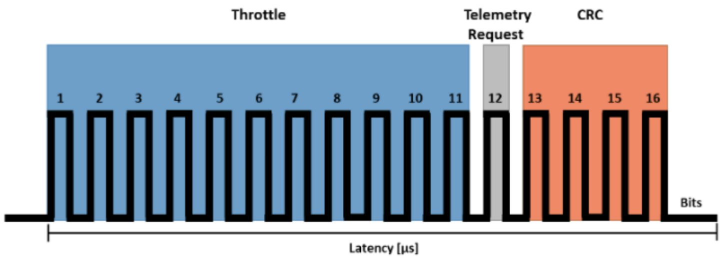


- Visual tracking and stabilization is a core part of this design and we use two different methods for tracking and stability reasons
- The first picture is what our NOIR camera sees when fitted with 940nm filter and its used to track the IR fitted object/human so the camera can stay following the object
- The second picture is the Apriltag we are using to help our ORB know its relative position, home base/docking and most importantly used for a more accurate reading than IMU. This is the primary system used for mode1 and mode 3 and even mode 4 if we reach it, only if the cam cannot see the tag do we resort back to the IMU



Custom ESP32 PCB

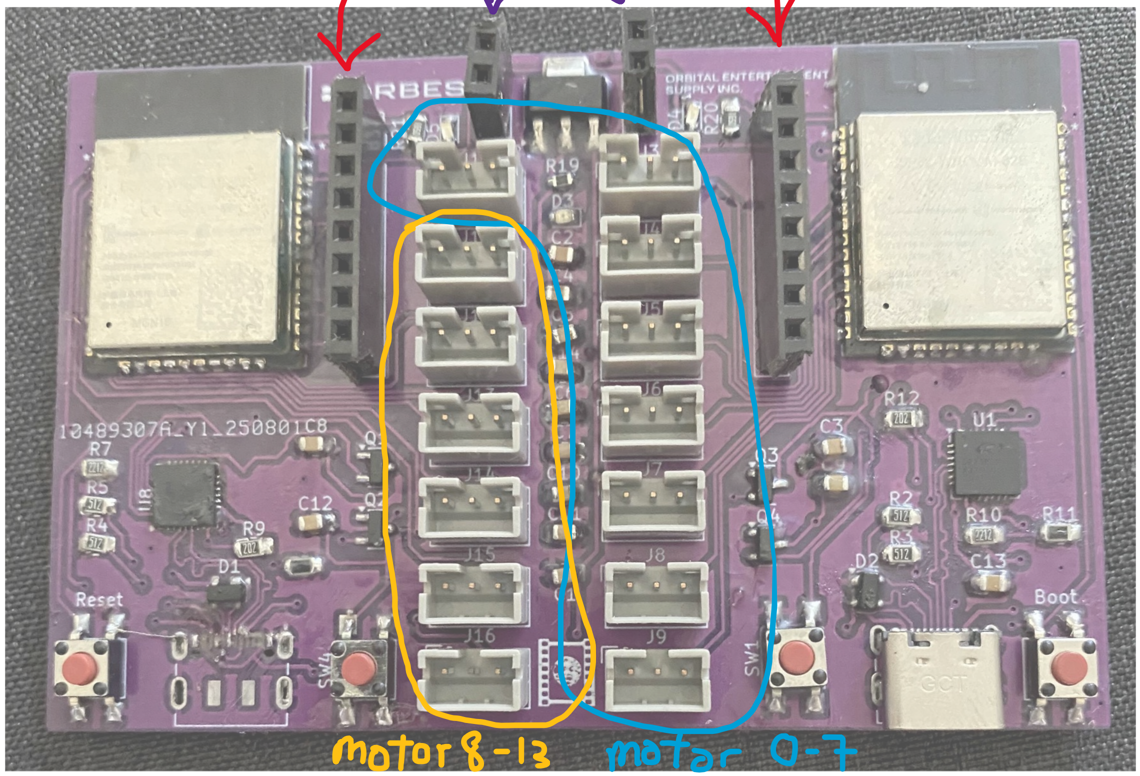
1. Receives a string from RPi5:
    - List of motor throttle commands
2. Parses the message
3. Sends the corresponding values to each motor as DShot signals
    - Generated using the ESP32's RMT peripherals.



*ESP32: limited 8 RMT peripherals on each chip

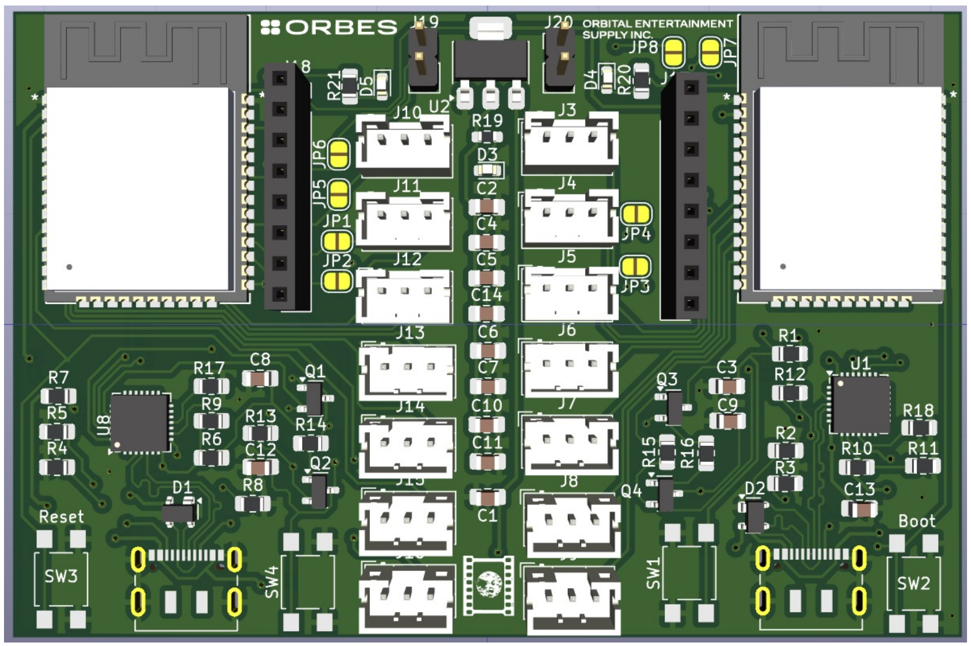
Optional I2C slots

Optional UART slots

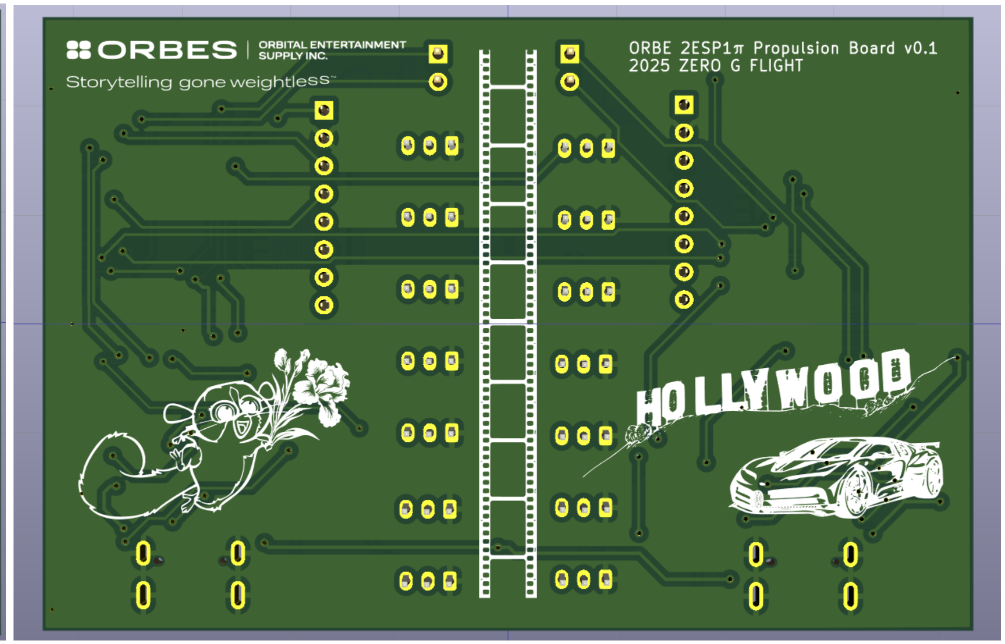


14 motors connect via JST connectors

Avionics: Motor Control PCB

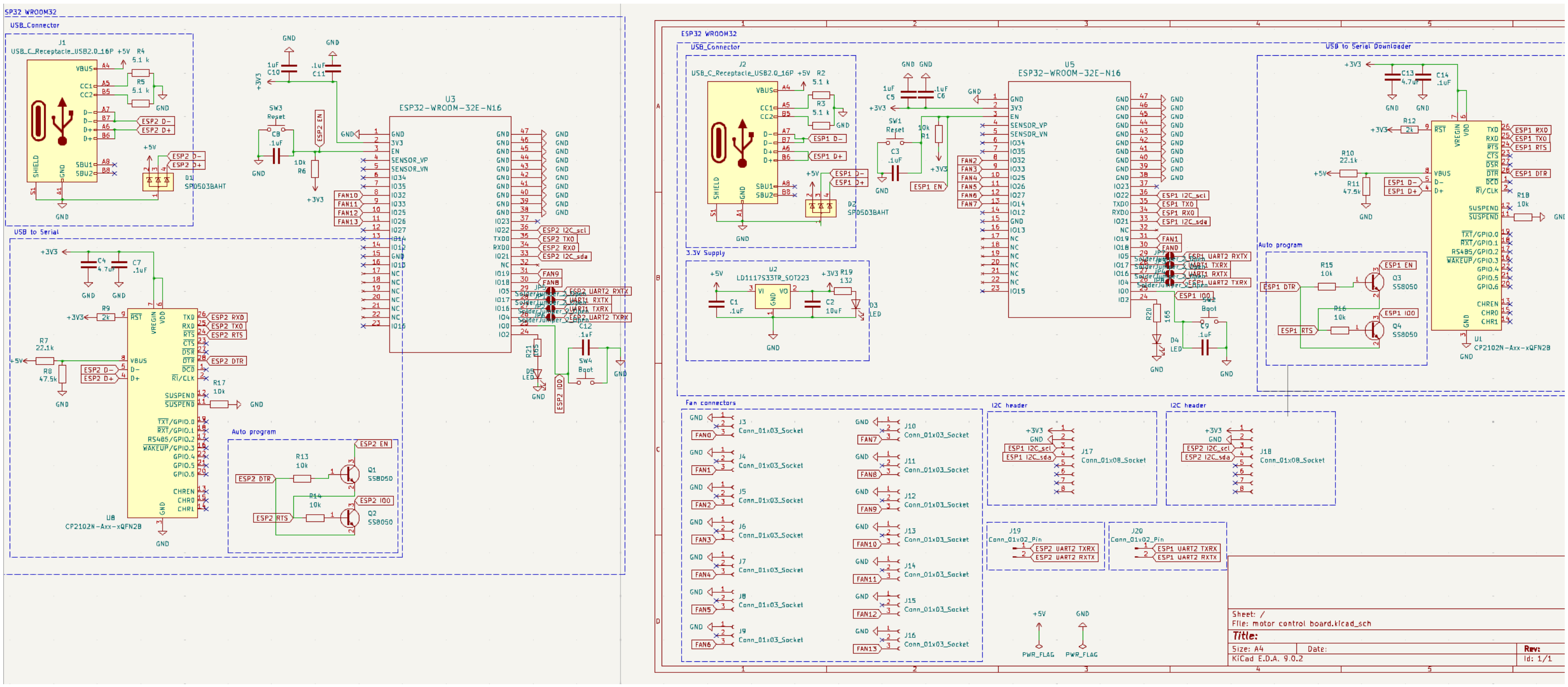


Front



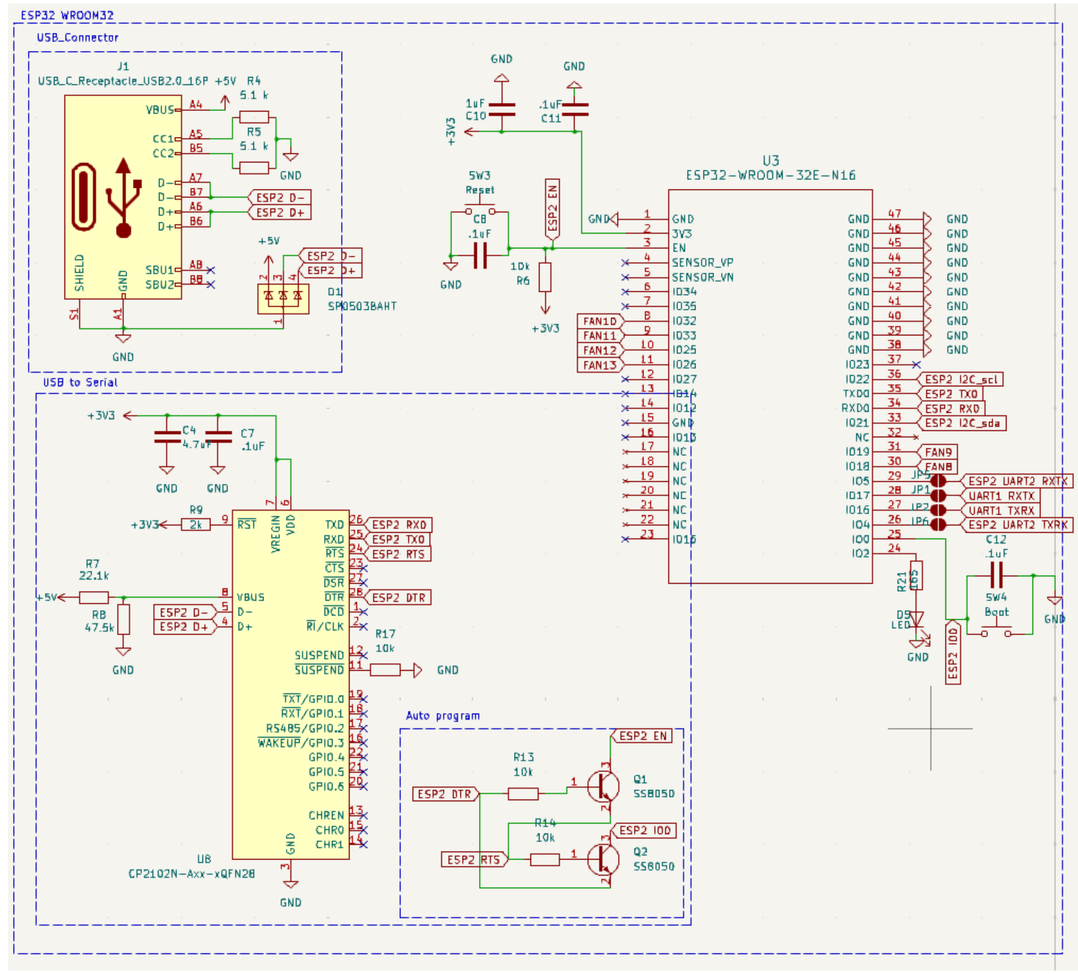
Back

Avionics: Full Schematic



Avionics: a closer look, left side ESP32

- USB to Serial chip
- 6 motors (#8~13)

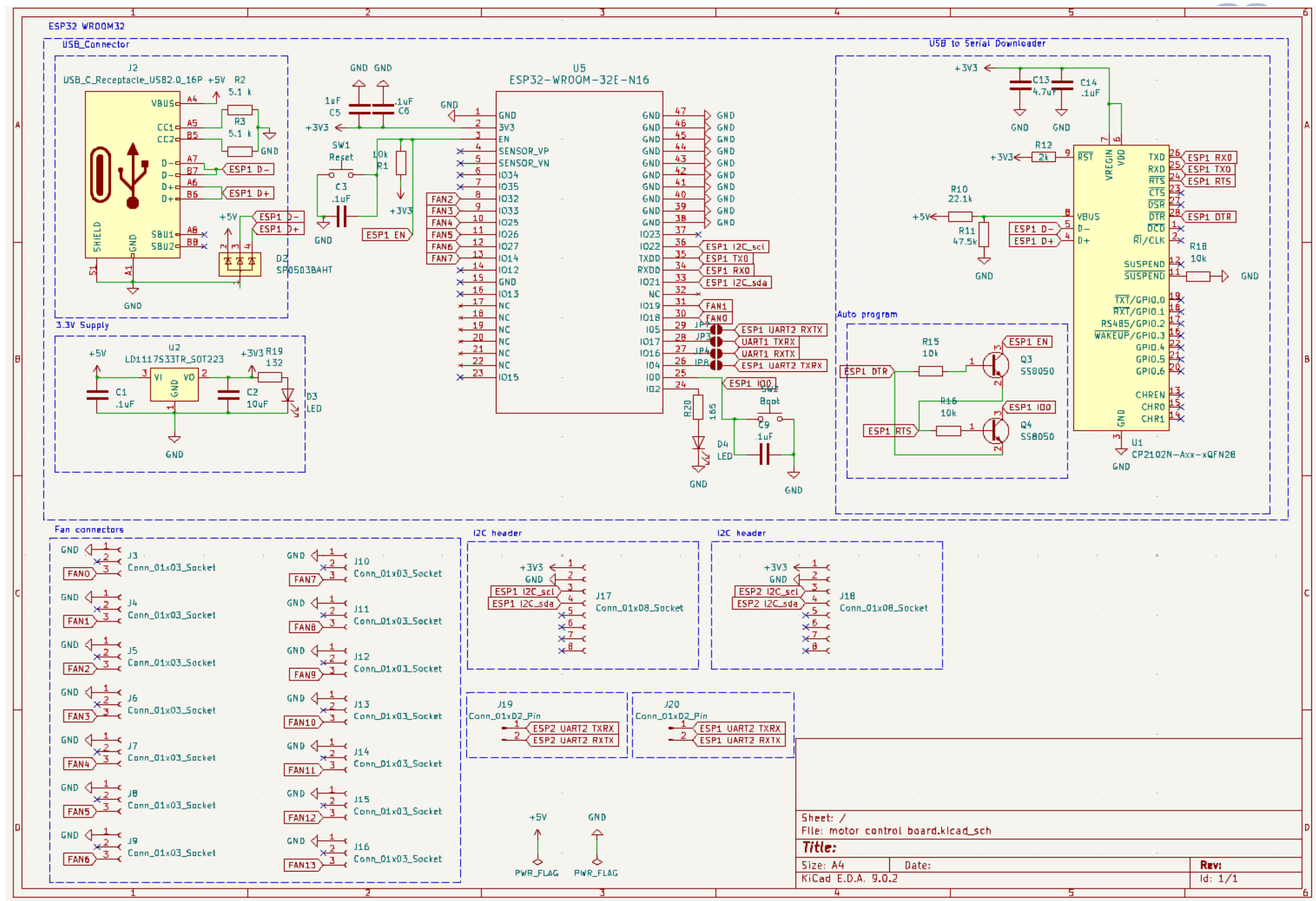


Avionics: a closer look, Right side ESP32 and connectors

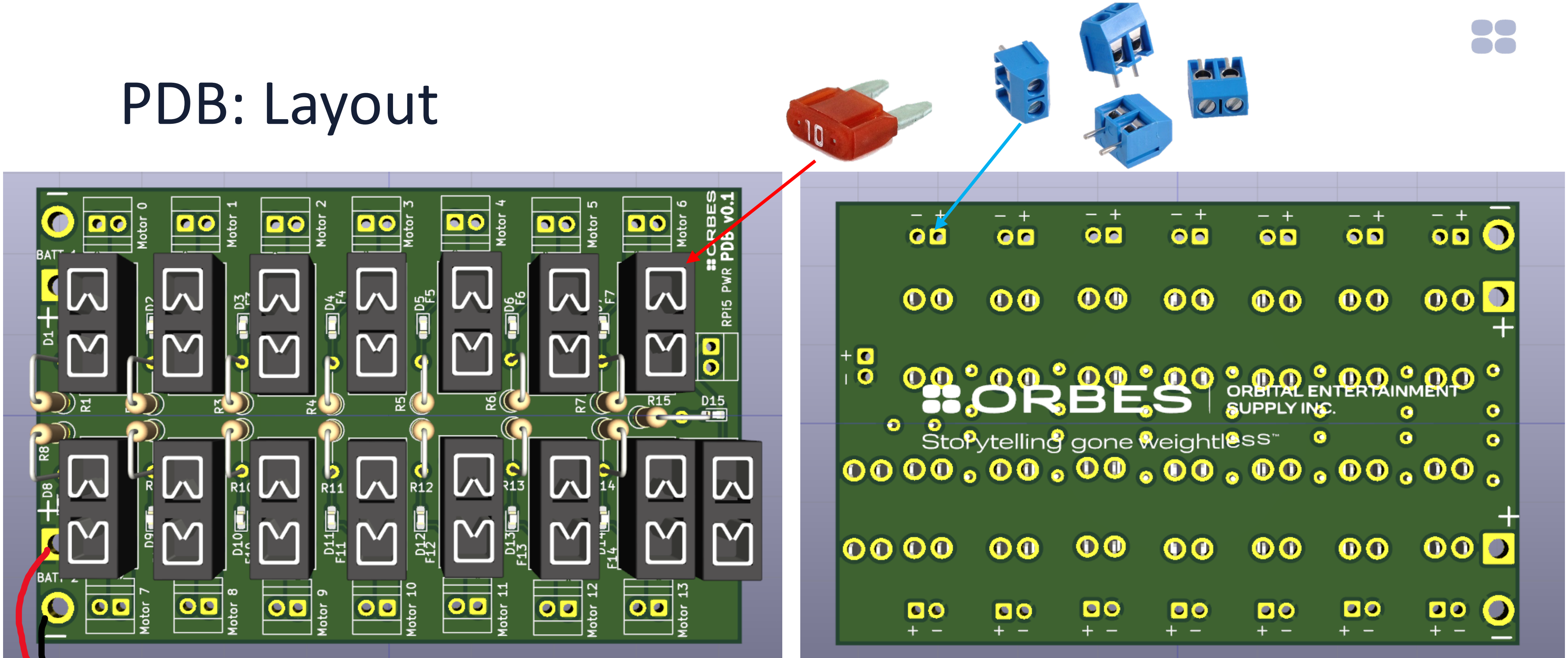
- 3.3 V LDO regulator
- USB to Serial chip
- 8 motors (#0-7)
- 14 jst connectors

One for each ESP32:

- 2 x I2C headers (for optional tests)
- 2 x UART headers (for optional tests)



PDB: Layout



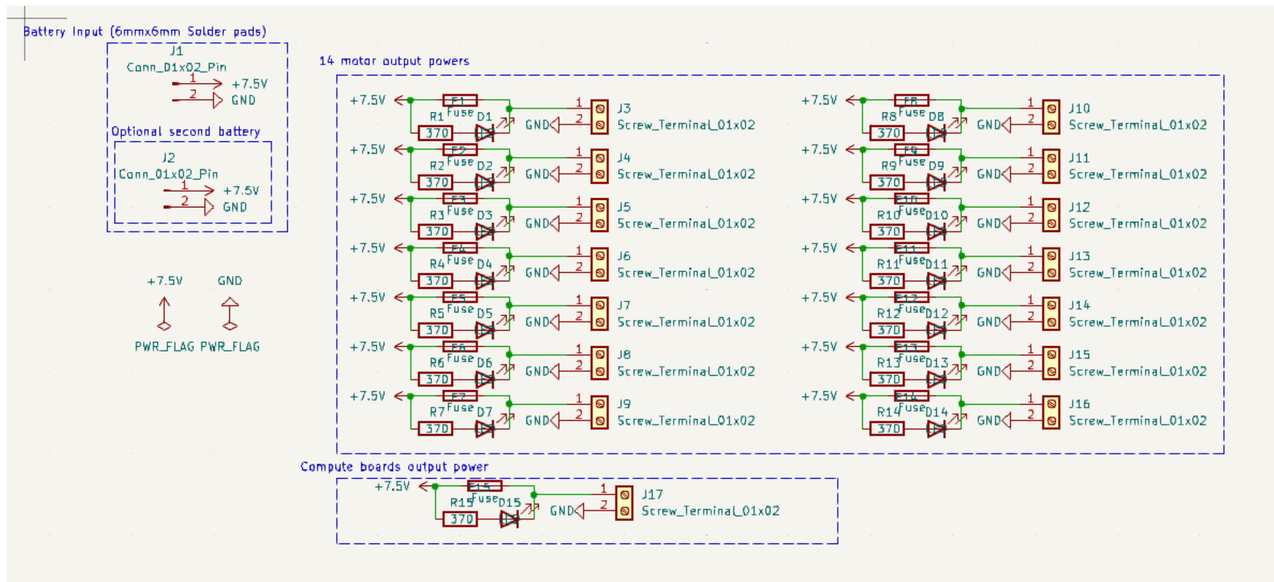
Deans T plug // Direct battery solder

PDB schematics: a closer look

Dual Batteries input

15 power outputs:

- 15 fuses
- 15 fuse fail indicators
  - (LED that lights up if fuse is blown out)



PDB wire thickness

Given:

- constant 3.5A DC current (estimate)
- ΔT=30°C
- Track length = 10mm (roughly)

track width should be =90.45mil, set to 100mil here

Propeller ADC Subsystem

Andrew, Simon, Lukas, Gavin

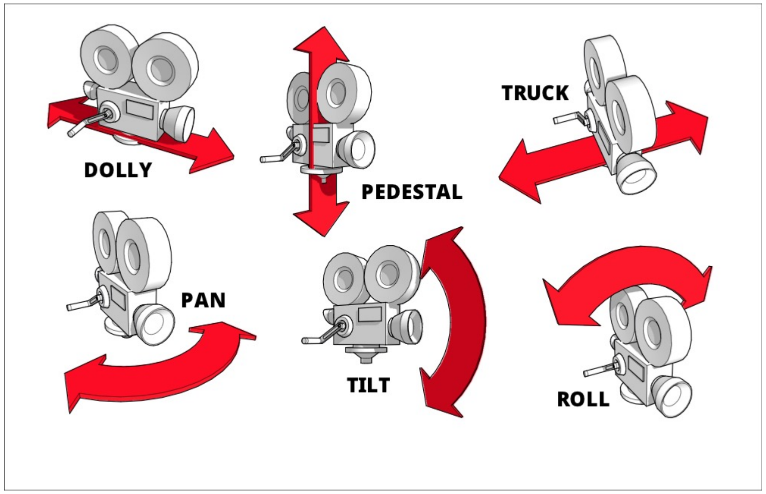
 |

Key Driving Requirements

KDR (Lukas)	Description	Reasoning
1	Enable independent 6DoF maneuverability	Prototype must be able to translate and rotate in a decoupled fashion
2	Provide 350 mN (35.7 gf) of thrust per propeller (40 mm) and 700 mN (71.4 gf) of thrust per propeller (76 mm)	Enable a stopping distance of 1 m from 1 m/s in X and Y directions and 1m from 0.7 m/s in Z direction
3	Provide torques of 78 mNm about z axis, 96 mNm about y axis	Allow compact size of prototype and
4	Securely shield and duct propellers	Ensure propellers do not loosen and threaten safety of engineers/passengers or vehicle in case of failure
5	Capable of stopping from full speed along each of three axes within 1 m	Reasonable stopping distance is essential for operation modes to be executed without damaging the vehicle

Desired Performance

- Achieve 45 degree start/stop maneuver in 1 second (yaw)
    - Enables rapid tracking along most important film making axis of rotation (pan)
    - Tracks walking speed (~ 1.5 m/s) from 2 m away
- Achieve > 1 m/s maximum velocity (forward) requiring 1 meter of stopping distance
    - Simulate dolly operating in film

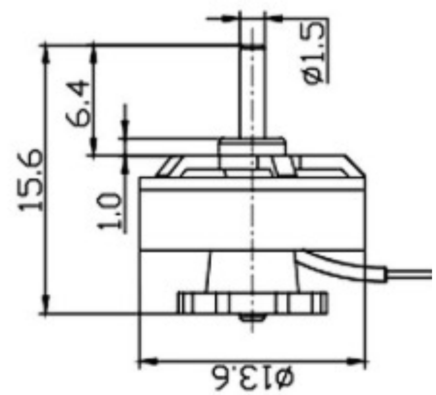


Updated Motor Selection

- Due to poor performance of motors chosen for PDR, especially in ducts, the system required an upgrade

- 100% throttle with 40 mm propeller showed a roughly 250% increase in thrust from about 200 mN to 680 mN (unducted) due to higher RPM

- Improvements came at the expense of a much higher current draw, with 3.3 A nominally at 100% throttle compared to the previous 0.88A



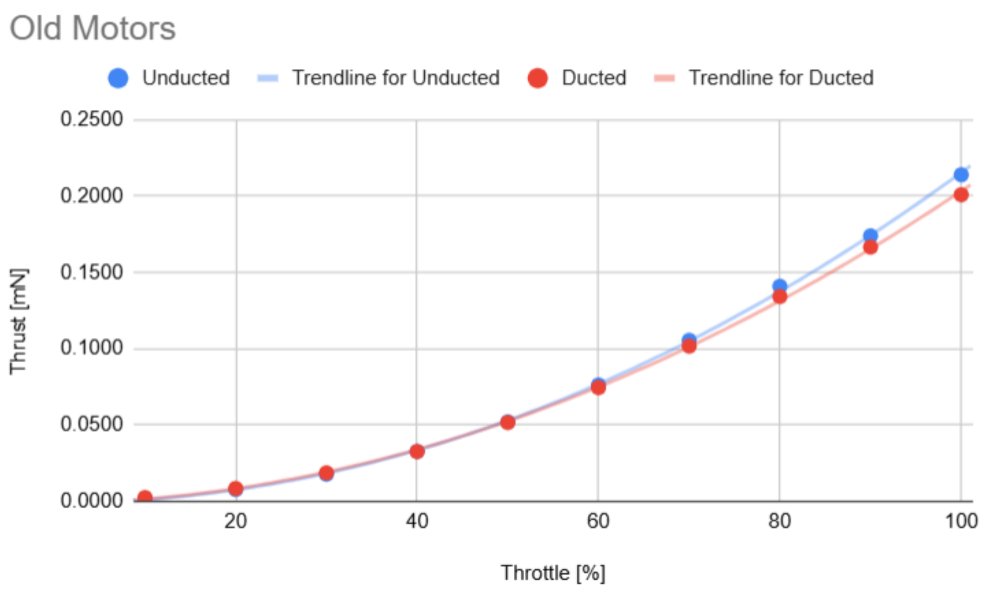
Performance Comparison

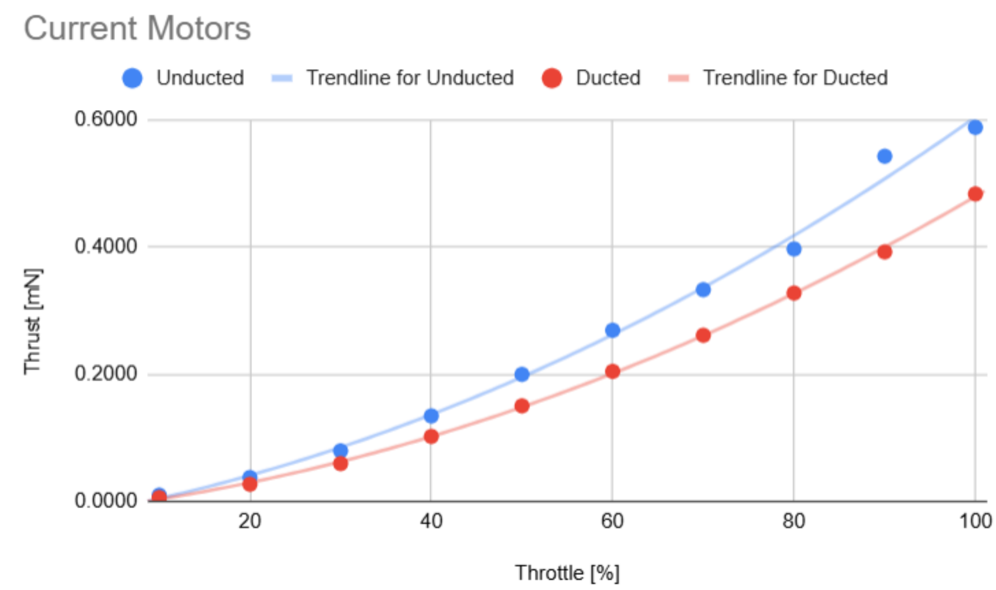
- Clear improvements across full scale of throttle
- Updated motors pass previous full throttle around 50-60%
- Performance between duct to duct is likely very different as they were trimmed by hand; properly toleranced ducts must be tested, in development current
- Testing in the integrated system shows that exit velocities (at the same distance) vary dramatically, suggesting unequal thrust on a motor to motor basis

	Unducted		Ducted	
Throttle [%]	AVG [mN]	STDEV [mN]	AVG [mN]	STDEV [mN]
10	0.0018	0.0002	0.0025	0.0005
20	0.0077	0.0003	0.0084	0.0004
30	1.8133	0.0004	0.0187	0.0002
40	0.0327	0.0009	0.0326	0.0004
50	0.0520	0.0014	0.0516	0.0007
60	0.0762	0.0010	0.0745	0.0011
70	0.1055	0.0018	0.1017	0.0014
80	14.3667	0.0004	0.1342	0.0015
90	0.1739	0.0010	0.1666	0.0035
100	**0.2140**	0.0009	**0.2009**	0.0015

	Unducted		Ducted	
Throttle [%]	AVG [mN]	STDEV [mN]	AVG [mN]	STDEV [mN]
10	0.0098		0.0059	
20	0.0375		0.0269	
30	0.0794		0.0594	
40	0.1343		0.1021	
50	0.1997		0.1501	
60	0.2690		0.2044	
70	0.3328		0.2612	
80	0.3969		0.3275	
90	0.5427		0.3923	
100	**0.5880**		**0.4835**	

Thrust/Throttle Curves

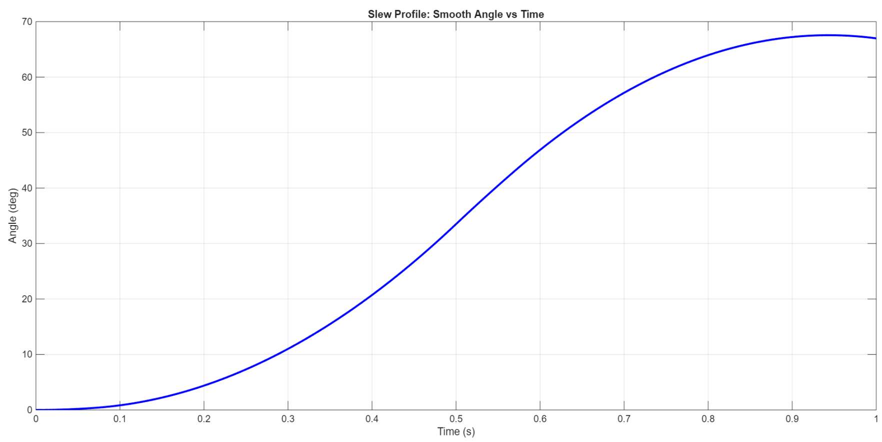


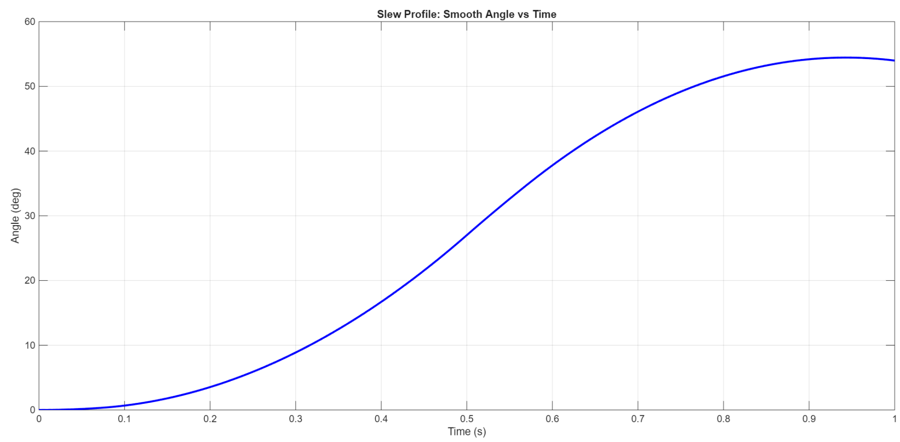


2nd Order fits with $R^2 > 0.99$ for all cases

Simulation Performance *Ducted Data*

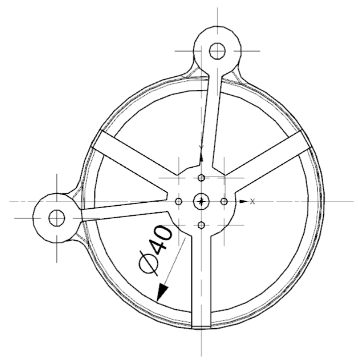
- ~ 70 ms ramp up time (0 – 100% throttle)
- Translation along x up to 1.17 m/s with 1 m stopping distance
- Up to 54 (pitch) or 67 (yaw) degree start/stop in 1 second

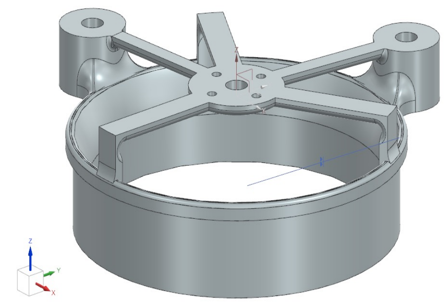


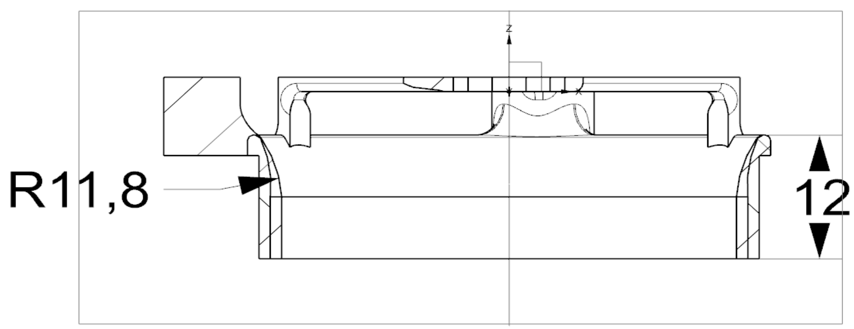


Duct Design & Results

- Design follows proven theory according to
  - Coanda Effect
  - Increasing inlet area
  - Maintaining laminar flow to slight outlet expansion

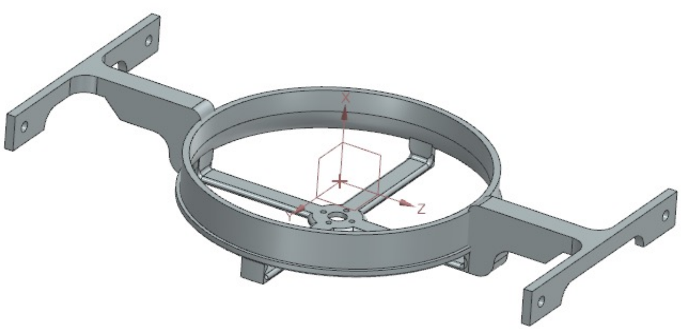


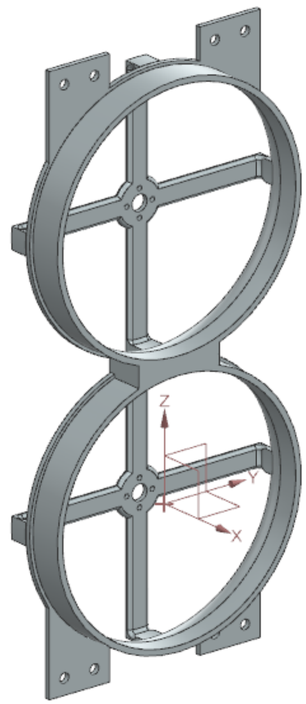


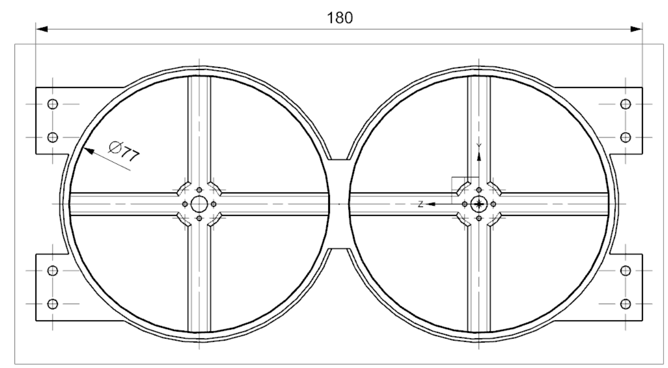


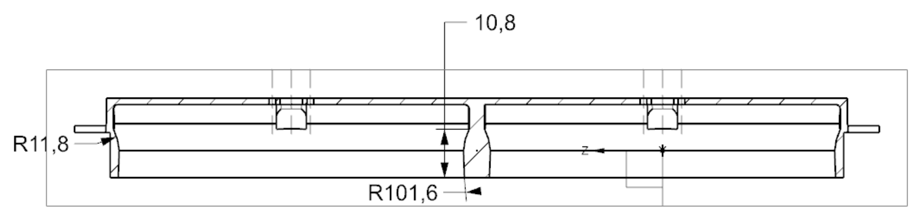
Duct Design & Results

- In reality, imprecise 3D printers have prevented the benefits from being realized
  - High tolerances must be used, otherwise ducts must be machined down
  - Smoothness of curvature is not maintained









Conclusions and Future Plans

- Though ducted performance falls very short of design expectations, performance is still far above where it needs to be -> clear to fly, but anticipating control troubleshooting

Next Steps

- Inconsistency in motor-to-motor performance must be either calibrated or minimized by purchasing more motors and selecting only consistent performers

- CFD will be continued to design more optimal ducting and propellers

- Higher fidelity 3D printing paired with more precise machining will be used to achieve closer to ideal ducting

Power/EPS Subsystem

Andrew, Simon, Walid, Lukas, Justin

Key Driving Requirements

KDR	Description	Reasoning
1 (Lukas)	Have a power capacity of 46 Wh minimum	Supply sufficient electrical power to 2 flight computers + 14 motors for duration of 0G flight with 2 Factor of Safety (14 minutes)
2 (Walid)	Ensure safety of engineers and/or passengers on flight	Electrical components are of concern on aircraft due to overheating and fire risk
3 (Lukas)	Minimize mass, moment of inertia contributions to integrated system	Preserve maneuverability
4 (Andrew)	Provide sufficient discharge rate for 16 power-drawing components	Multiple components operating synchronously draw significant power

Power Budget Overview

- Max Wattage*
  - Theoretical: 197 W
  - Experimental: 105 W

Component	Quantity	Nominal Voltage [V]	V (Min-Max) [V]	Avg Consumption [A]	Peak Amperage [A]	Max Continuous Current [A]	Duty Cycle [%]	Total Allocated Power [W]	Margin [%]	Total Power - Margin [W]
Raspberry Pi 5	1	5.1	4.75-5.25	0.54	1.36	1.36	100	6.936	25	8.67
IMU	1	3.3	1.7-3.6	0.014	0.025	0.025	100	0.0825		0.0825
ESP32	2	5	4.75-5.25	0.15	0.5	1	100	5	25	6.25
RaspPi Camera Module 3	2	5	4.75-5.25	0.2	0.25	0.5	100	2.5		2.5
1104 4300 KV Brushless DC Motor	14	7.4	3.7-11.1	3.3	4	56	40	165.76	25	207.2
Motor driver (ESC)	14	7.4	7.4-11.1	20	25	N/A	40	N/A		0
Raspberry Pi Active Cooling Kit	1	5.00	4.5-5.5	0.12	0.15	0.15	100	0.75		0.75
BlackMagic MicroStudio Camera 4K G2	1	14.4	12-16.8	1.6	2	2	100	28.8		28.8
				Total Max Cont. Current Draw [A]		**61.035**		**Total Power Draw [W]**		**225.453**
				Total -w/o Camera [A]		**59.035**		**Total Power Draw -w/o Camera [W]**		**196.653**

- System Max Continuous Current*
  - Theoretical: 59A
  - Experimental: 18A

System	Supply	Full Charge Lifetime [min]
ORB	Zeee 100C 8000 mAh 2S LiPo	18.1
BlackMagic Camera	Batmax 2650 mAh 7.4 V Lithium Ion	40.9

18.1 minutes based on theoretical wattage

*w/o BlackMagic Camera

Chosen Battery

- ZEEE 8000 mAh 2S (7.4V) 100C Lithium-Polymer Battery Pack

- 59.2 Wh
  - 8.0 Ah * 7.4 V
  - Supports full system for 34 minutes at nominal operation (40% duty cycle for motors, 100% for the rest) according to experimental results

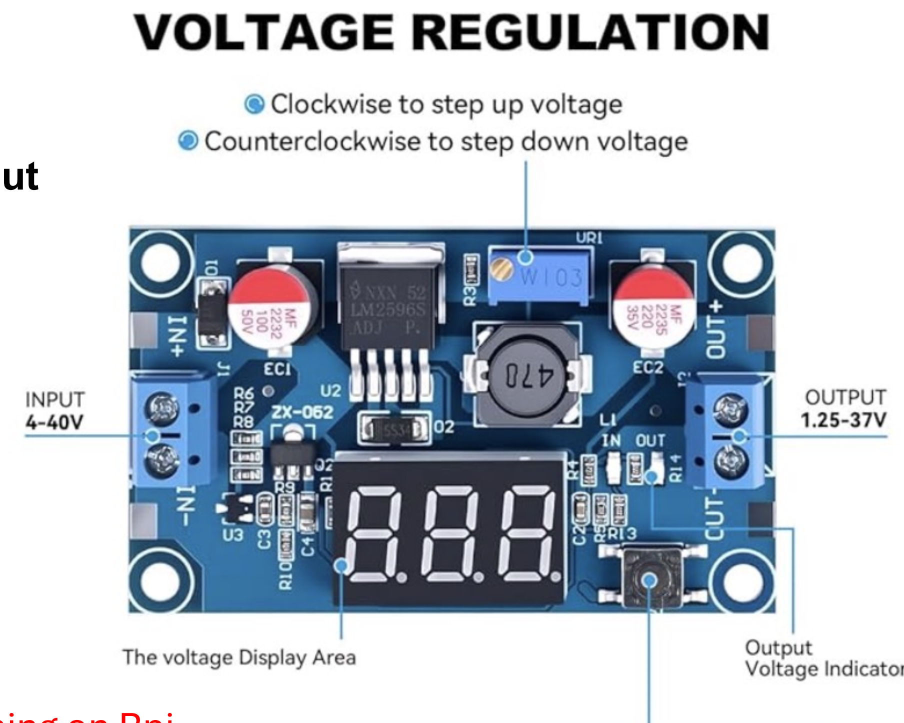
- Capable of supplying 800 A continuously (~35-40 FoS based on specs, likely closer to 10-20 FoS range realistically)
  - 100 C * 8.0 Ah

- Motors require up to 7.4 V, boards require up to 5.25 V

- Non-permissible power supply on space flight, sufficient for 0G flight

Buck Converter

- Steps down **7.4 V battery input** to a stable **5 V output**
- Powers the ESP32 boards and Raspberry Pi
- High-efficiency DC-DC conversion for reduced heat and power loss



*Outputting 5.3V into the Rpi: Seeing low voltage warning on Rpi

Testing (Power)

- Current motors shows no signs of thermal concern
    - Not warm to the touch at any point during tesing

- 100C 8000 mAh, the battery can safely handle a continuous supply of up to 800 A*

- Individual motors draw 3.3 A nominally at 100% throttle; testing 14 motors in parallel showed a measured nominal draw of 9.92A at 90% throttle

- Boards and subcomponents (cooling unit, IMU) expected to draw 2-3 A nominally

- Power distribution rated for up to 30 A

- Demands on the power system appear to be easily met -> No concerns of thermal stress due to operation

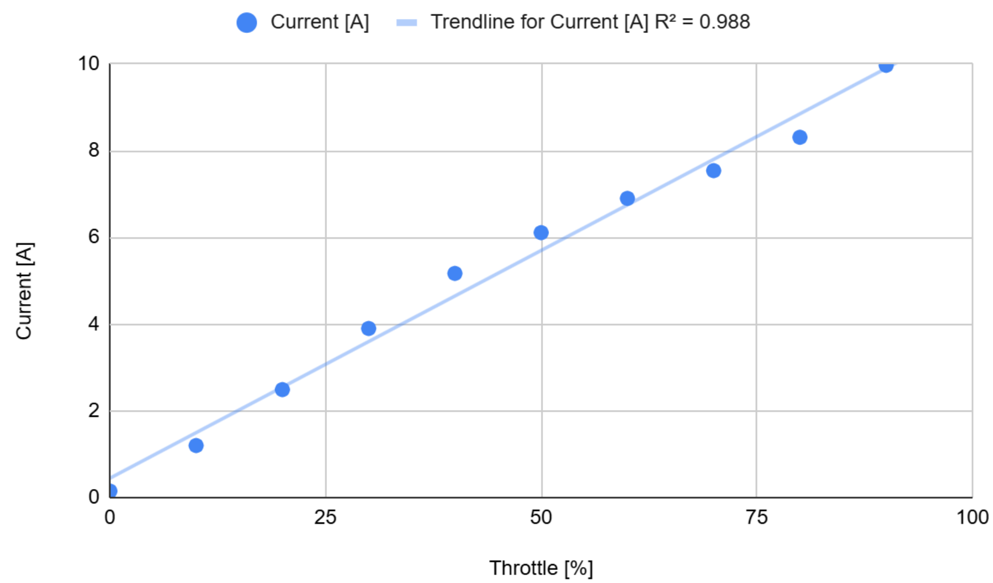
*This number is according to theory; in reality it is suggested that the true limit is about 50% or less

Interesting Experimental Results

- **Previous** motors (0.88A nominally @ 100% throttle) all fired simultaneously @ 100% only drew 4.4A continuously, peaking ~7.4A during 0-100% ramp up
    - Via power supply (capped at 10A), suggestions of performance drop off

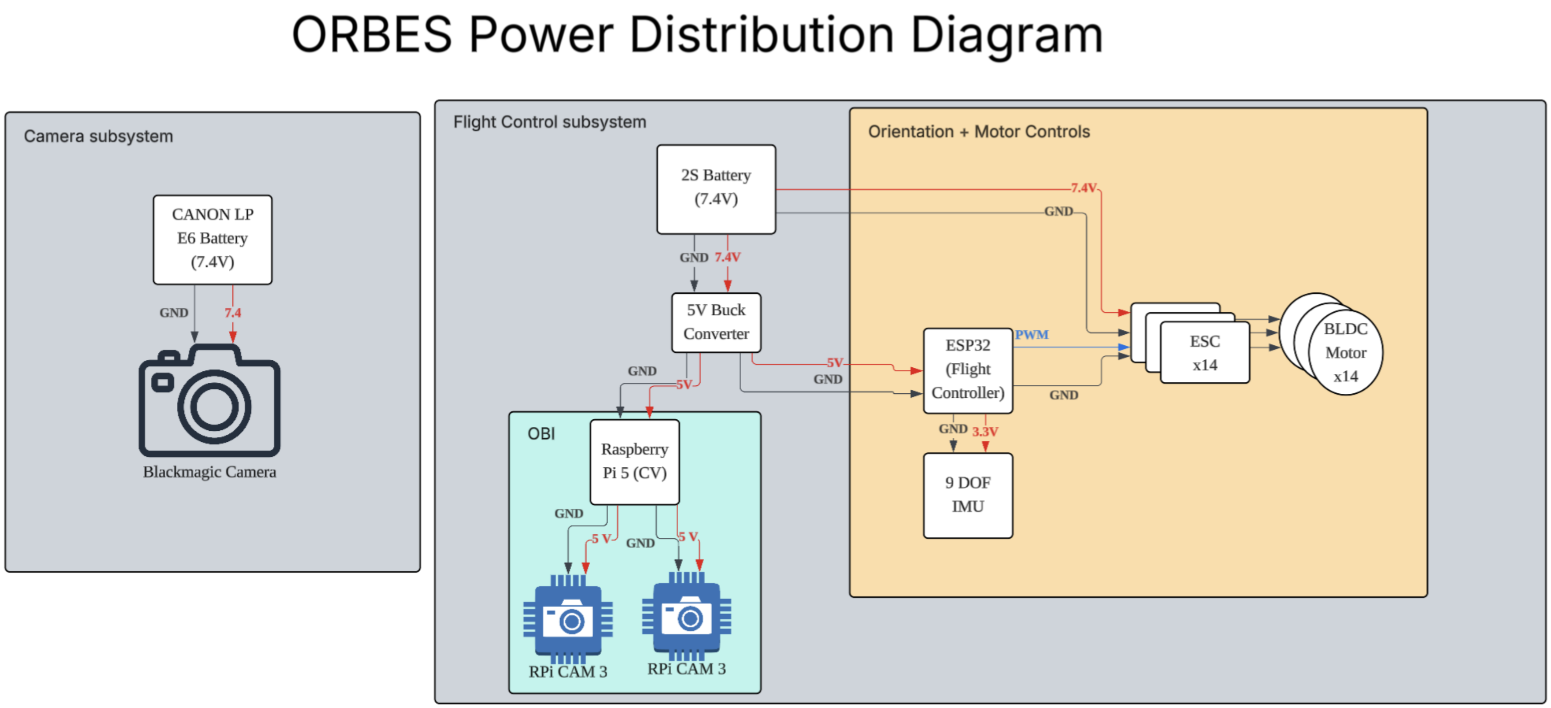
- **Current** motors (3.3A nominally @ 100% throttle) all fired simultaneously @ 90%* drew 9.92A continuously
    - Via battery (capped at 200-400A), no suggestion of performance drop off
    - Extrapolating expects 100% draw to be in the 11-12A region

*90% Since our DMM has a max current reading of 10A



System current draw vs motor throttle plot using current motors, up to 90%.

Avionics Power Distribution diagram



[Editable link]

GNC "OBI" & Flight Software Subsystem

Andrew, Gavin, Walid, Nicholas

Key Driving Requirements

KDR	Description	Reasoning
1 (Gavin)	Plant All Axis Response shall have 5° Steady State Error with 1° Drift Rate	Provide acceptable pointing accuracy for film-making
2 (Walid)	Plant All Axis Response shall have within 1 m translation stop distance within 0.05 m/s drift	Operational Safety Boundary
3 (Gavin)	Plant Yaw Response shall have 30° Start-Stop rotation within 1 sec (0.5° drift rate)	Pan Maneuvering Film-making
4 (Gavin)	Plant X-axis Response shall have 0.2m station keeping within 0.05 m/s drifting	Film-making demonstration and react to disturbance

Software Architecture

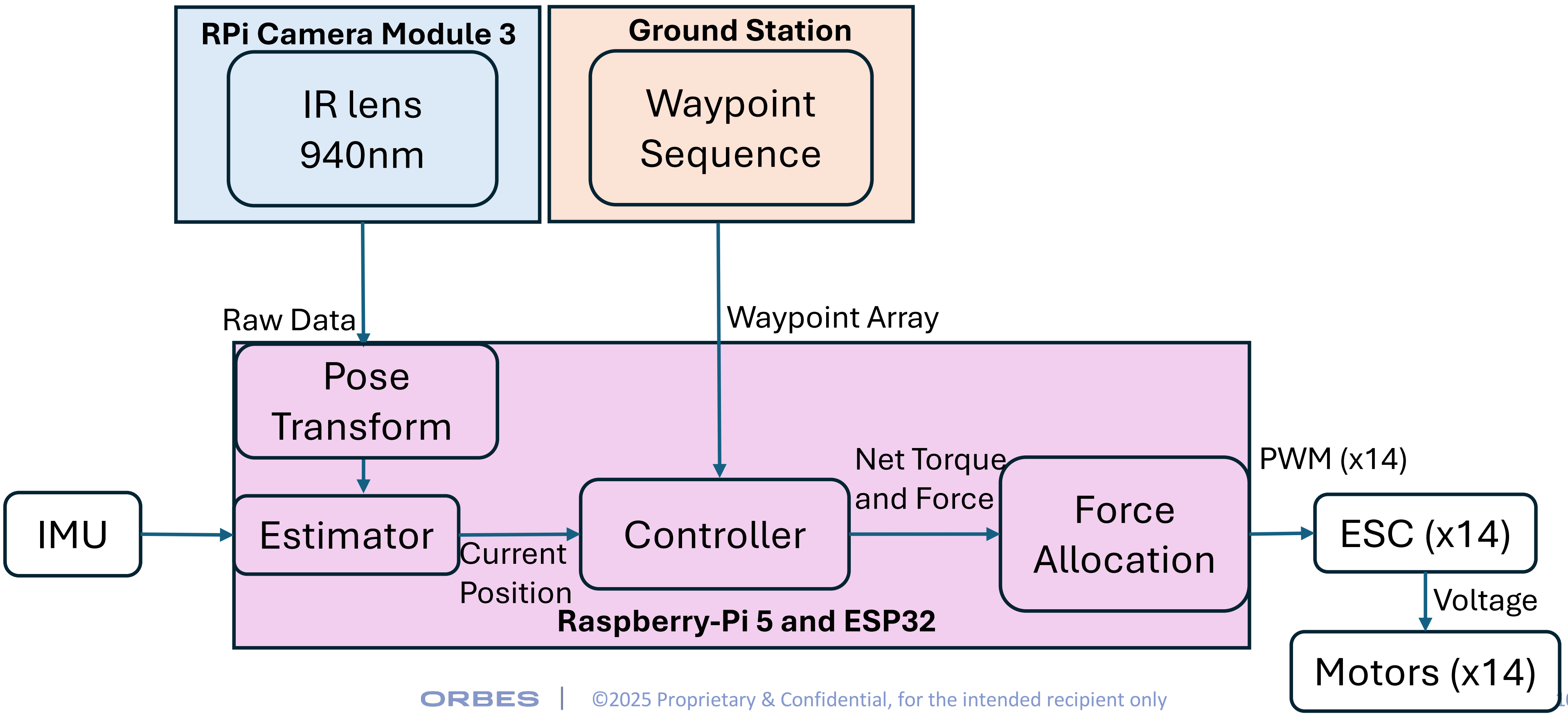
RPi:

Running Raspbian 64 Bit
2 onboard RPi cameras
IMU
OpenCV
AprilTag
LQR + Kalman filter
Cooling fan control

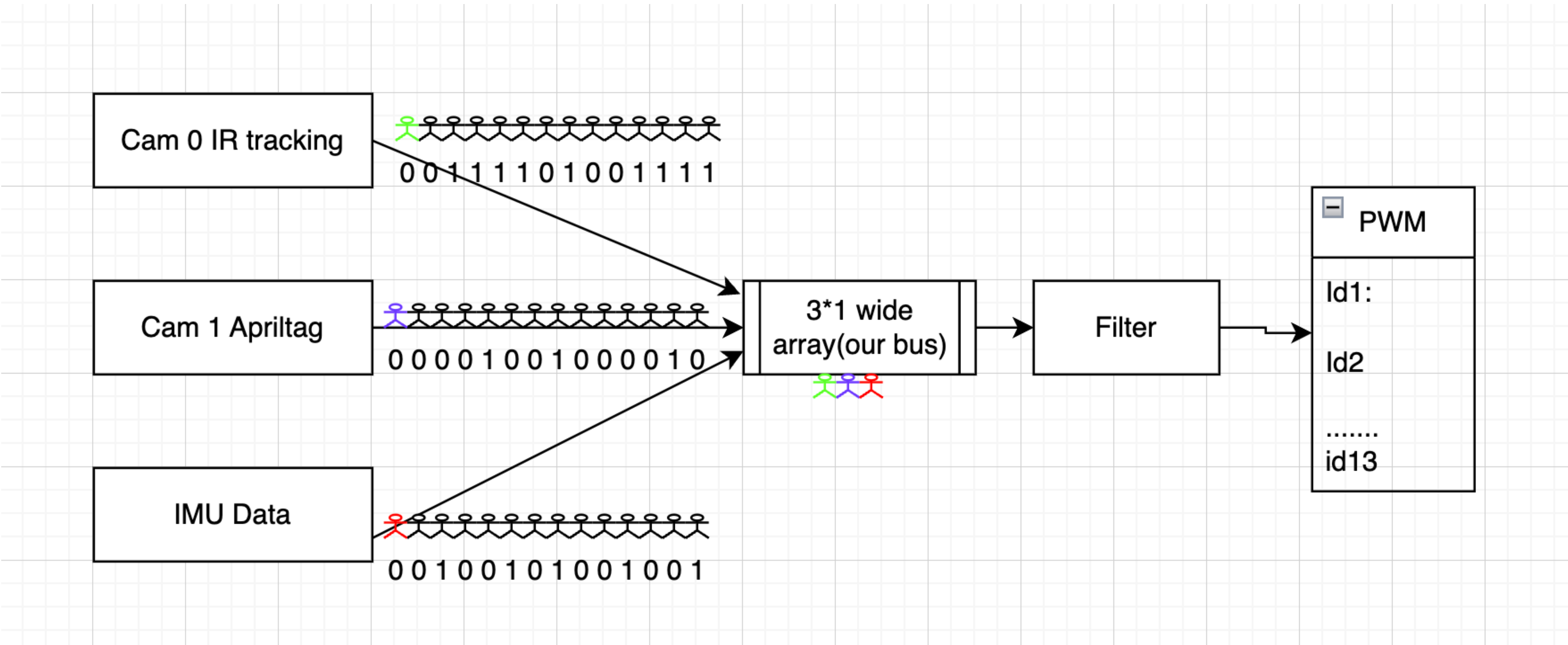
ESP32:

Running FreeRTOS OS
Attitude control
Controlling 14 Fans

GNC Overview



Bus system for stale/fresh information



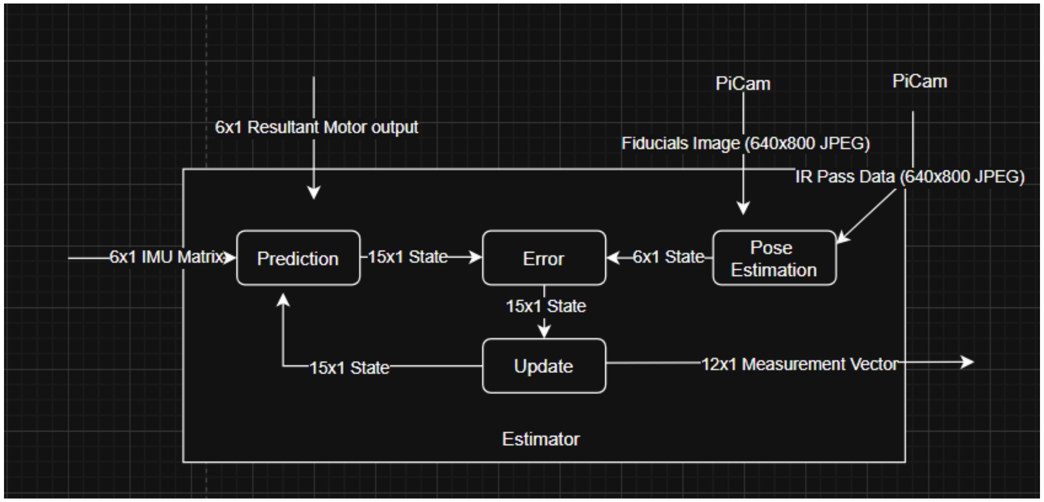
Format


```
'imu': {
    'accel': [2, 10, 9],
    'gyro': [1, 2, 3],
    'orientation': {'yaw': 50, 'pitch': 90, 'roll': 30}
  },
  'apriltag': {
    'x': 20, 'y': 40, 'z': 60,
    'distance': 1.8, '50': None,
    'corners': [{'x': 200, 'y': 200}, {'x': 200, 'y': 200},{'x': 200, 'y': 200},{'x': 200, 'y':200}
  },
  'ir': {'x': 200, 'y': 300}
```


Estimator Overview

- Extended Kalman Filter
  - Sensor Fusion using IMU prediction
  - Camera Pose Update
  - 15x1 State for system full observability
  - Noise model using gaussian distribution

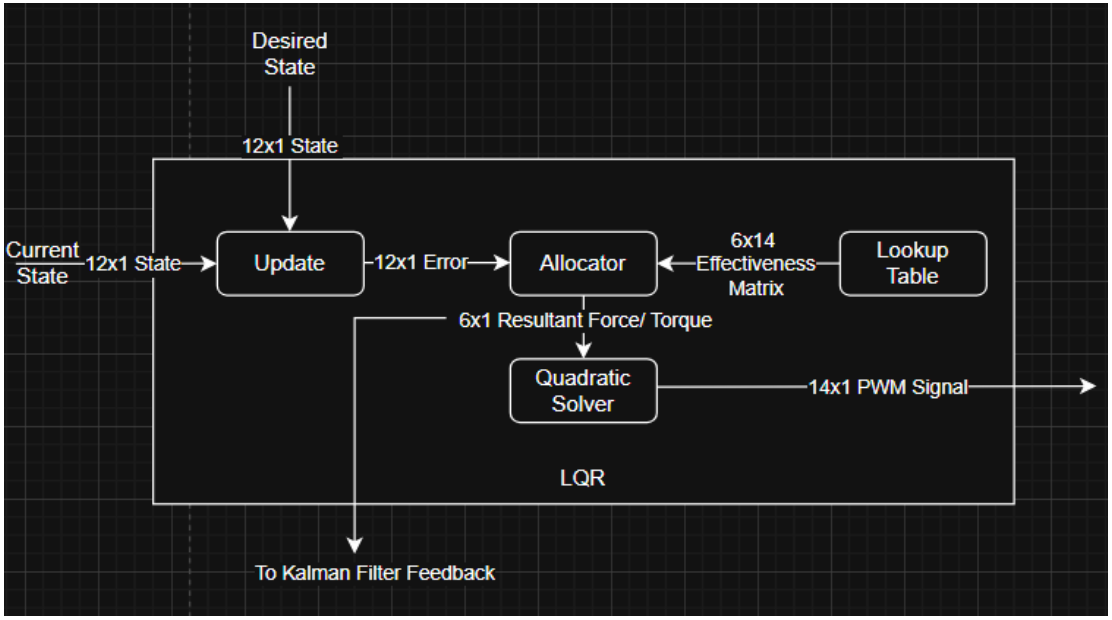
*16x1 state



LQR Overview

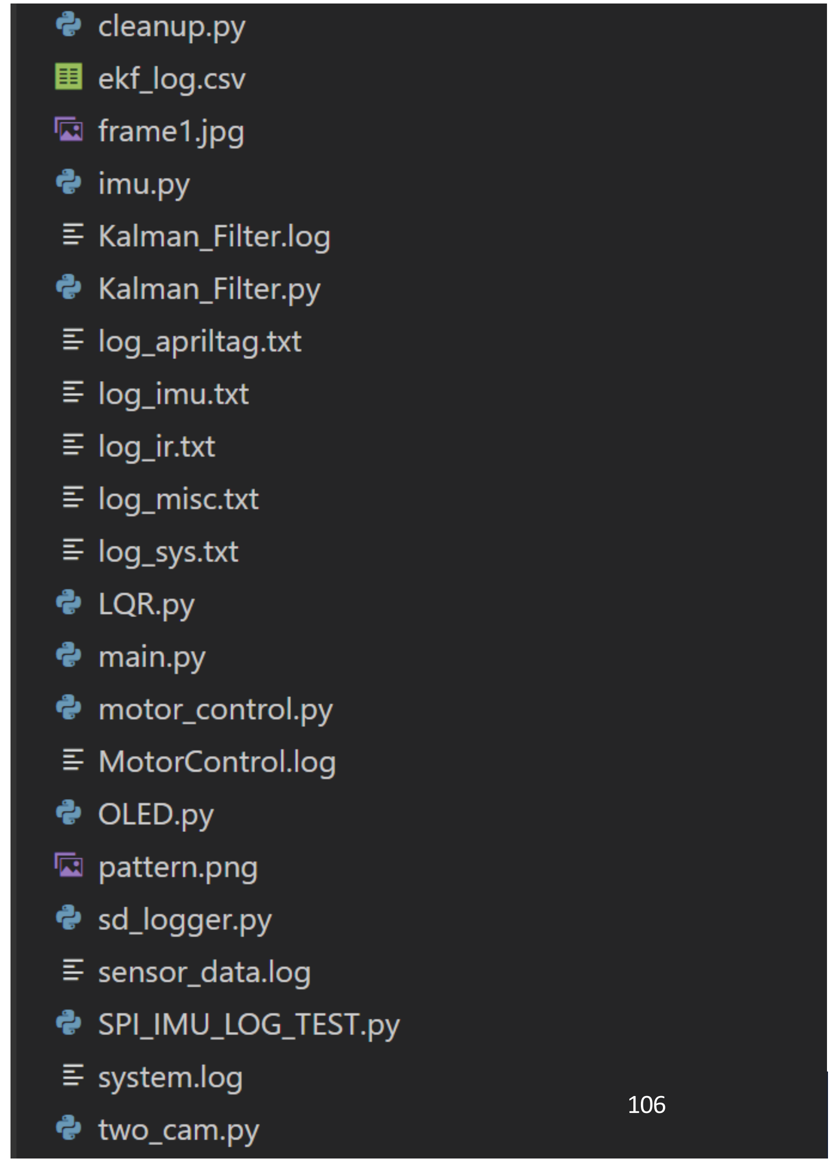
- LQR optimization for control
  - Compares Current and Desired State
  - Effectiveness mapped by motor location and thrust curve
  - Using open sourced CVXPY library solver
  - Objective function using least squares
  - Cost function ranking: Velocity > Position > Pose > Spin rate
  - Updates every 100ms
  - Couple ~4 ms to go through this calculation
  - 14 unknowns (PWM)

*Allocator is the LQR



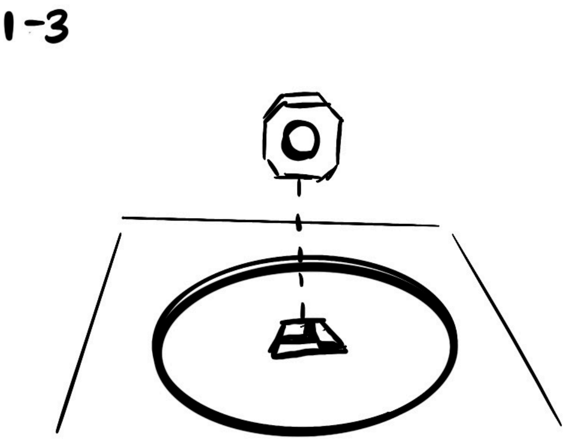
Logging and saving files

- We have a robust and long logging system on board tracking and saving all data being recorded and even passed between our different files, these data are
- Kalman Filter log
- AprilTag log
- IMU log
- IR log
- System Log (one file that has all the different logs in it)
- Motor Control log
- Miscellaneous log (any errors that we have are saved here)



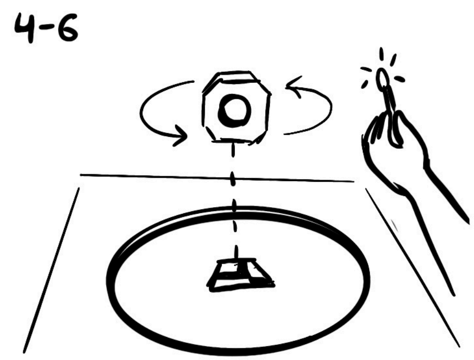
GNC Mode 1

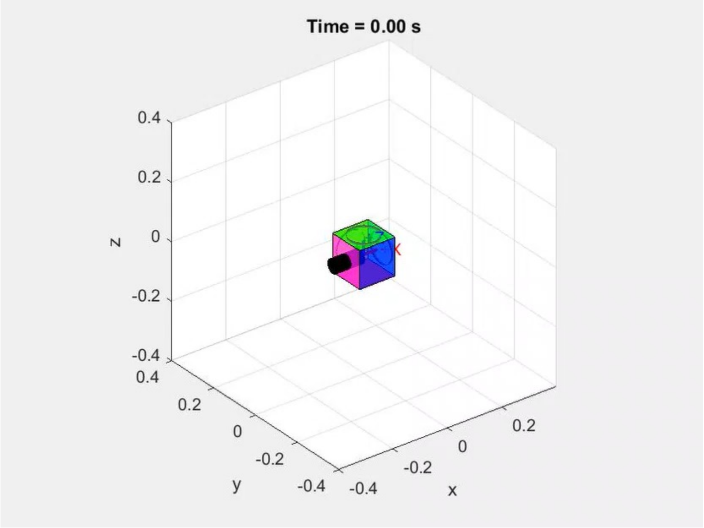
- Full Translational and Rotational Control
  - 8-fans Active on X-axis
  - 4-fans Active on Y-Axis
  - 2-fans Active on Z-Axis
- Maintain within +/-5° All Axis Alignment and 1° Drift Rate with AprilTags for 20s
- Maintain within 0.2m X-Y alignment with 0.05m/s drift AprilTags for 20s
- Maintain within 1m with 0.05m/s drift Z-distance with AprilTags



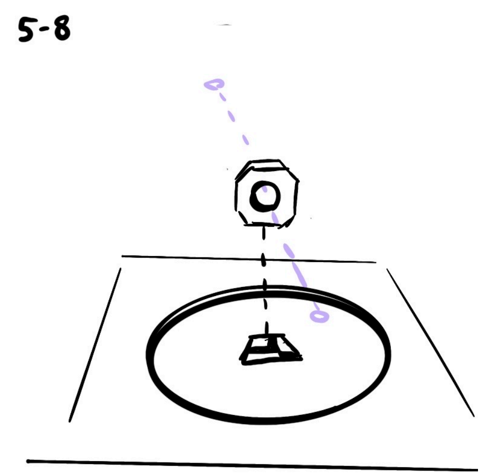
GNC Mode 2

- Full Translational and Rotational Control
  - 8-fans Active on X-axis
  - 4-fans Active on Y-Axis
  - 2-fans Active on Z-Axis
- Maintain within 5° Yaw and 1° Drift Rate Alignment with IR Target
- Maintain within 0.2m and 0.05m/s drift X-Y alignment with AprilTags for 20s
- Maintain within 1m and 0.05m/s drift Z-axis with AprilTags

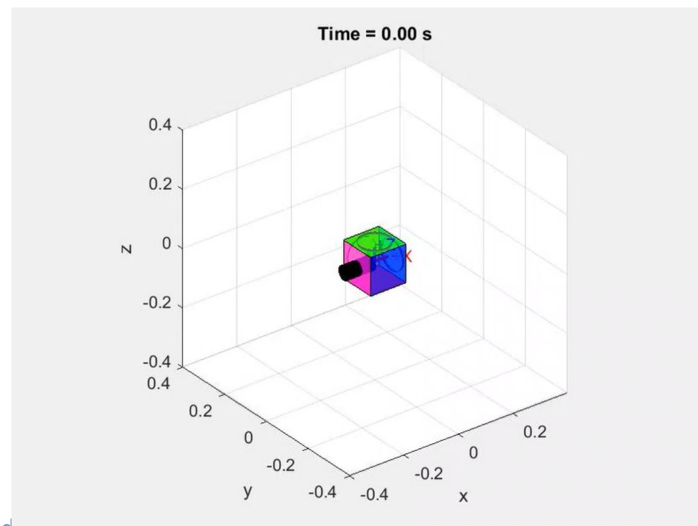




GNC Mode 3

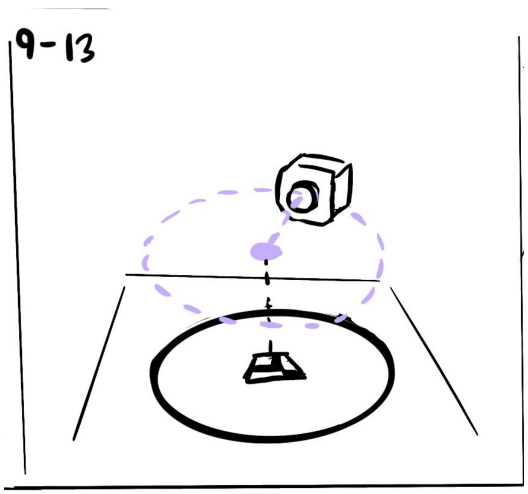


- Full Translational and Rotational Control
  - 8-fans Active on X-axis
  - 4-fans Active on Y-Axis
  - 2-fans Active on Z-Axis
- Maintain within 5° and 1°/s drift alignment
- Maintain within 0.2m and 0.05m/s drift X-Y alignment with AprilTags for 20s
- Maintain within 1m and 0.05m/s drift
- Move along X-axis +2m, hold 3s and move back to X = 0 m
- Hold at X = 0m within 0.05m/s drift for 2s
- Move along X-axis –2m, hold 3s and return x= 0 m



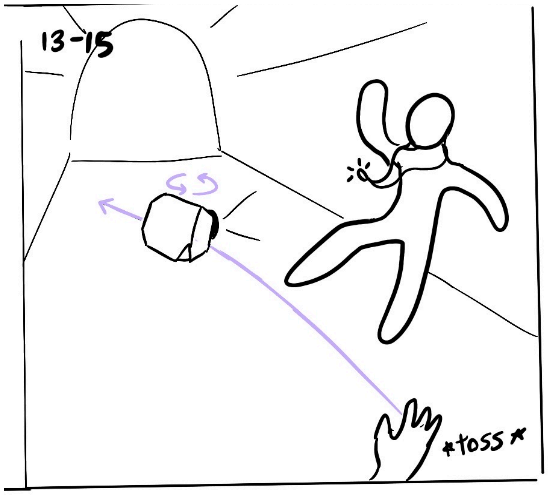
GNC Mode 4

- Full Translational and Rotational Control
  - 8-fans Active on X-axis
  - 4-fans Active on Y-Axis
  - 2-fans Active on Z-Axis
- Maintain within 1m X-Y radial alignment and 0.05m/s drift with April Tags for 20s
- Maintain within 1m and 0.05m/s drift Z-distance with AprilTags
- Maintain Pitch & Roll Accuracy within 5° and 1°/s drift
- Maintain 0.2 m/s X-Y translation speed within 0.2 m drift corridor



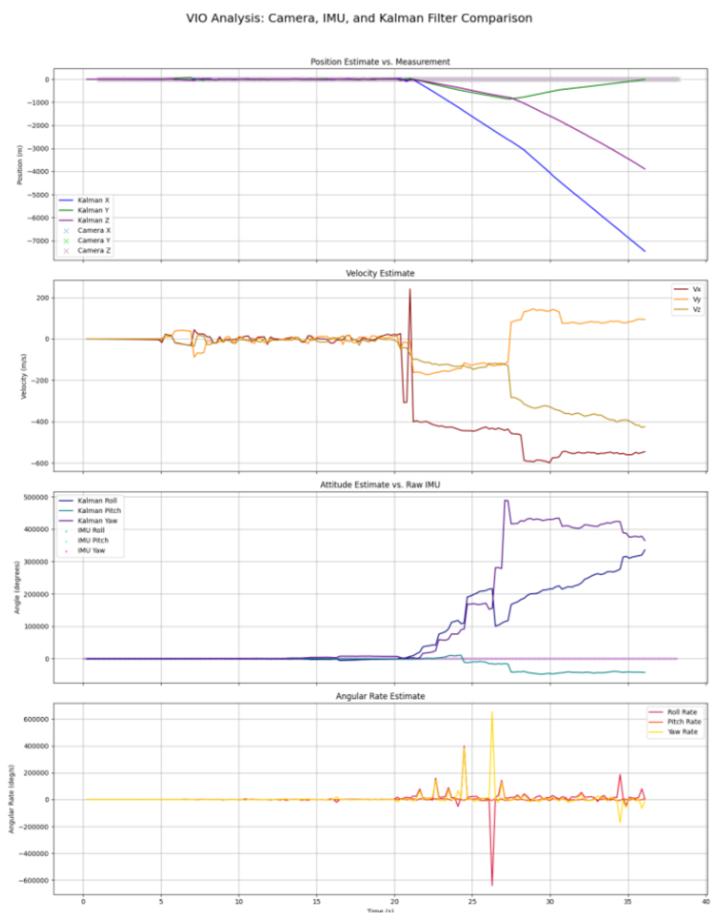
GNC Mode 5 (Bonus)

- Full Rotational Control, no Translation Control
  - 8-fans Active on X-axis
  - 4-fans Active on Y-Axis
  - 0-fans Active on Z-Axis
- Maintain within 5° Yaw + Pitch Pointing Alignment with IR target within 1°/s drift
- Maintain within 5° Roll Aligned with Cabin Horizon within 1°/s drift
- Plant is given a +0.5 m/s X-axis initial velocity
- Timer set for 20s



GNC Testing Objective

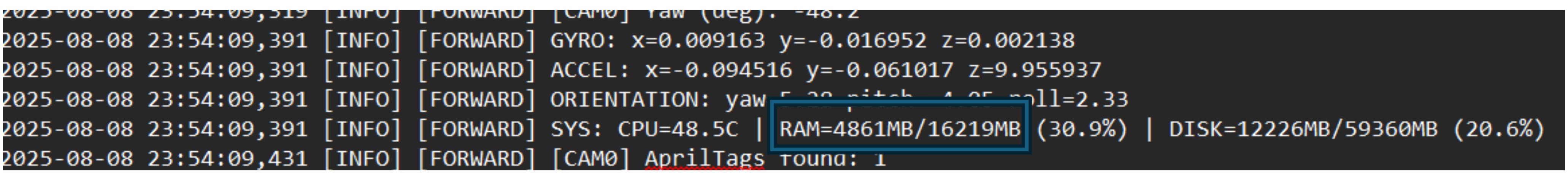
- Validate Kalman Filter Performance
  - Steady State Error
- Validate LQR Performance
  - Response Time
  - Tuning Factor
- 3 DOF test (X,Y, Yaw)
  - 30° rotational in 1 sec start-stop
  - 1 m translation
  - Station-keeping inside the 0.2m circle



RAM Data Budget Summary

- RAM capacity will have 16GB available for data handling

Item	Data Rates	Notes	Channel
PiCam	210 MB	JPEGS	UART
IMU	0.6 MB	State Variables	I2C (ESP) UART (Pi)
Pi5 Command	2MB	Data Logging	UART
OpenCV/April tags	2GB	VGA Files	UART
Raspbian OS	2GB		



Storage Data Budget Summary

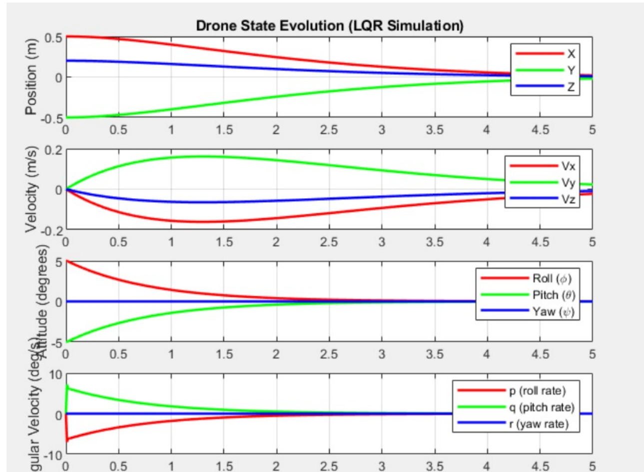
- Storage capacity will have 100GB available for data logging and 1TB for Media

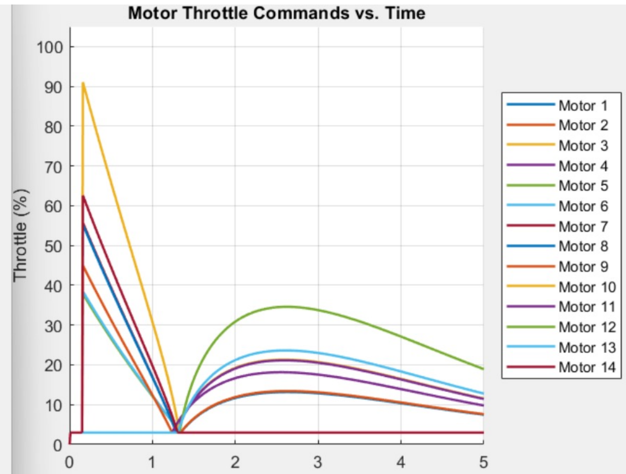
Item	Data Rates	Notes	Runtime Available
PiCam	0.2 MB/s	State Variables	
IMU	0.1 MB/s	State Variables	
Pi5 Command	1.4 MB/s	Data Logging	
OpenCV/April tags	0.2 MB/s	State Variables	
Total	1.9 MB/s		877 min

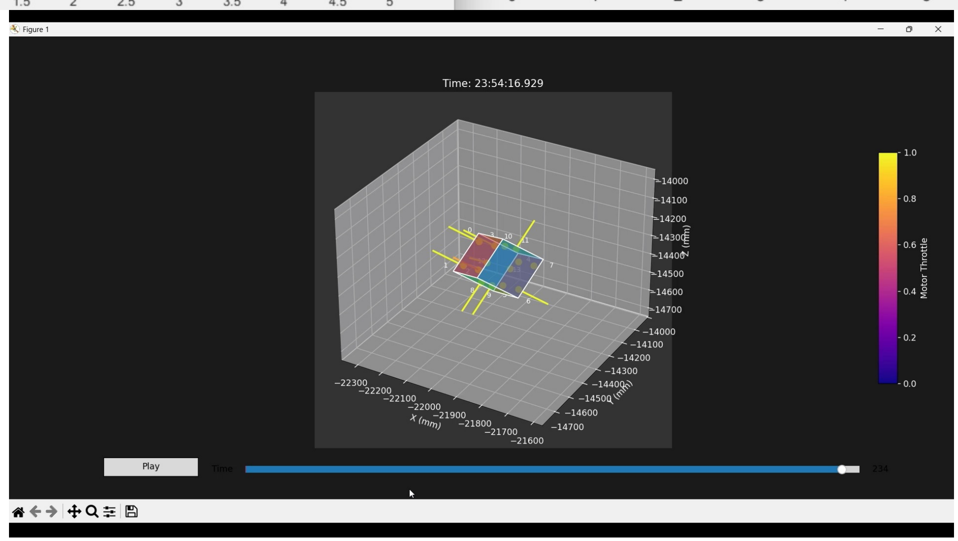
Item	Data Rates	Runtime Available
Black Magic	10 Mb/s	1666 mins

Simulation with Kalman Filter

- MATLAB LQR Simulation using quadprog() function
  - Isolating LQR into camera feedback as temporary solution
- FlatSAT Kalman Filter feedback using CVXPY library
  - Kalman Filter Feedback running to Gimbal Lock
  - 16 State EKF will be using Quaternion
  - Include IMU Biases to achieve full observability
  - Visualized using pyplot







Progress (Simulations)

- Control methodology successfully implemented in a Python simulation.
  - LQR for 12x1 state vector (world position, world veloctiy, euler angles, and body rates)
  - Quadratic Programing for force allocation (subject to all thruster values positive and less than maximum thrust, minimizing net thrust).
  - Tuning Factor optimized using CAD properties